USEReady

A PRACTITIONER'S GUIDE TO

ALTERYX

Alteryx 2018.1

A Practitioner's Guide to Alteryx®

A Practitioner's Guide to Alteryx®

Published by
USEReady
805 Third Avenue, 11th floor, New York, NY 10022
www.useready.com

ISBN: 978-0-692-44795-6
Manufactured in the United States of America

About USEReady

USEReady's mission is to help organizations succeed by fast tracking their business performance. We relentlessly strive towards value-driven innovation and digital transformation using advanced business intelligence, data management, and infrastructure management services. Our global delivery orientation is enriched with winning strategies, rich industrial and technology expertise and customer fanaticism. Our proven track record of success in versatile domains such as capital markets, insurance, healthcare, pharma, retail, media etc. is a testimony to our performance.

Visit www.useready.com or e-mail info@useready.com

About Alteryx, Inc.

Alteryx, Inc. is the leader in data blending and advanced analytics software. Alteryx Analytics provides analysts with an intuitive workflow for data blending and advanced analytics that leads to deeper insights in hours, not weeks, which is typical of traditional approaches. Analysts love the Alteryx Analytics Platform because they can deliver deeper insights by seamlessly blending internal, third party and cloud data; and then analyze it using spatial and predictive drag-and-drop tools. This is all done in a single workflow, with no programming required. More than 1,000 customers and thousands of data analysts worldwide rely on Alteryx daily.

Visit www.alteryx.com or call 1-888-836-4274.

Acknowledgements

When we started writing this edition of the book, we decided to go with a team of authors instead of a single author. The team went through a pretty challenging process of reading, updating, and reviews to get the book where it is today.

I would like to express my gratitude to the team of authors and other colleagues who helped in making this book a reality. Without them, the book truly would not be what it is today.

Uday Hegde: For being the driving force and a constant motivator.

Anand Vadul: For sharing the experiences and challenges faced in the last edition and being the technical guiding force.

Prashant Singh: For helping with content collation, editing, marketing, and an excellent cover design.

Honey Shahi

Letter from the CEO

As we are publishing this third edition, Alteryx is a public company. Alteryx has gained a wide spread adoption across industries and geographies. We are glad that we partnered with this successful product early on. We see an increased adoption through the orders that we receive for this book. This edition incorporates many feedback and suggestions received from our readers.

As suggested by many of you, we have incorporated additional hands-on exercises. Many newly certified Alteryx practitioners at USEReady have contributed to this book. Their hard work and dedication has resulted in this book that is now reaching 600 pages in content. A total of 11 practitioners have revised this book with upgraded product features, exercises and new chapters.

The team has gone great heights to ensure the content is refined to the most recent version of Alteryx and exercises are useful to a fellow practitioner.

We are grateful to our readers of the previous editions and their constructive feedback has helped us improve this edition. We hope our efforts are well worth it and you are going to find this book useful.

Uday Hegde
Chief Executive Officer
USEReady

Foreword

As Vice President of product management at Alteryx, Inc. working closely with our product team, our customers and our partners, a training manual from USEReady is testament to the growing demand for easy-to-use data blending and advanced analytics solutions. USEReady knows the analytics industry and has created this book as both an independent guide and as a classroom aid to help its customers and others not only quickly learn Alteryx products, but more importantly grow in their ability to help gather deeper insights from their data.

The book has created a great launching point for beginners who want a manual, in addition to the real-world use cases, so you can easily learn how best to use Alteryx Designer. By the time, you finish working through this manual, someone who has never opened the Alteryx Designer before, will be able to create workflows, design reports, develop applications, and write macros to solve any of their data needs.

These are exciting times as the Alteryx community continues to grow globally, and we continue to witness an unprecedented demand for data analytics with actionable information. The Alteryx Designer and materials like this manual help the community of self-service data analysts make the most of their data.

We appreciate the partnership with companies like USEReady that know how to help these self-service analysts and reduce the time to insight with Alteryx.

Laura Sellers
Vice President, Product Management of Alteryx

Preface

Every day we are faced with options, questions, and choices. These decisions, as we all know, are much easier to make when we are well informed. Let's say that we want to eat. We literally have an entire world of possibilities, given the proper resources. However, practically, there are real limitations. Are we at home without transportation? Are we backpacking in the mountains? Are we in the middle of Times Square? Do we have food restrictions for health reasons? Do we have $5 or $5000? What are we in the mood for?

This task that we all solve day in and day out depends on a considerable amount of information that we know about our world, and often take for granted. This information is all based on data about our world.

What Is Data?

Data is stored information. It comes in various forms ranging from the number and types of items on our desk to the total mass of the universe to the contents of this book to the information in digital files and systems, which will be our focus.

What Does Data Do?

Data does nothing. It simply exists. It is what we do with data that is important. When we look at data we interpret it to create meaningful information, which gives us the ability to make better-informed decisions.

How Do We Consume Data?

Data can be consumed in many forms. We can look at all of the raw data and read every piece individually. We can use aggregation methods to create summary data so that we can easily see high-level trends. We can visualize the data because a picture truly is worth a thousand words. Since we often do not want to look directly at the original data source and read each individual piece of data, we need to perform data preparation.

What Is Data Preparation?

Data Preparation is the process by which raw data is converted into a clean, usable source for later consumption.

The three core components of data preparation are data retrieval, data manipulation, and data export. In more traditional analytic terms, data preparation refers to the extract-transform-load process referred to as ETL. However, in order to ease communication, we are going to avoid these technical terms and discuss the aspects of the processes as follows:

Data retrieval refers to the process of going to a data source, asking for data, and returning the desired data.

Data manipulation refers to anything we decide to do to the data between the time we retrieve it and the time we export it.

Data export refers to what we do with data after we have extracted and manipulated it even if we haven't finished transforming it.

What is Data Manipulation?

Data retrieval and Data export are fairly straightforward; respectively, they can be likened to drawing water from a well and putting an ice cube in someone's drink. However, data manipulation is that tricky process of running the water through the pipes into our house, then filling the ice cube tray, then putting the tray in the freezer, and letting the water have enough time to freeze so that we have ice to consume. Going forward, we will be using an allegory to a river to explain the entire process of the data preparation and specifically the data manipulation portion.

Data manipulation can come in many forms which typically fall into three buckets as follows:

Combination

One of the most common problems with data is that it comes from multiple sources. It is generally possible to perform the analysis separately, or through a significant amount of manual effort, but these methods often leave something to be desired or are too slow for effective use. In order to solve this problem, we will be designing data streams that come together.

If we think about data streams as actual rivers, original data locations can be thought of as glacial streams, smaller rivers, or lakes. Bringing data together is like the tributaries that bring these different water sources together to form a river. Along the course of this river, way we can perform calculations.

Calculation

If the data is to be used, it is generally advisable to have as much data pre-calculated as possible. One reason for this is that it allows an organization to create a standardized formula for everyone's use. Another is that when we can run calculations before data is provided to a front-end user or system, the consumer will experience a much faster process.

If the data is to be used in a report, then the calculations are often fundamental aspects of that report.

Returning to the river analogy, we can think of calculations as hydroelectric dams along a river, we are using the resources that already exist in order to generate something new.

We may change the landscape because we are changing the flow of water, and we are also slowing down the river (introducing calculations will slow down the data preparation process).

We also have the ability to transform the data stream into a more useable format.

Transformation

It is often the case that data is not in the format that we need. We may have been given access to a database that has data stored in a very machine-readable format, and we need to pivot the table to make it human readable, or we may have been working with an Excel file which has data extremely normalized that makes it hard to use in a front-end system. Either way, we need to transform the structure of the data so that it can be effectively consumed. In thinking about the river, we can imagine this as the process of cutting a channel into the riverbed so that the river is deep enough to move barges up and down. In doing this, we are fundamentally changing the structure of the river in order to make it more useable.

To the reader

In the following chapters, we are going to cover many topics, but the format of the chapters will all be the same.

You will assume the role of a new consultant at a company that works with Alteryx. We introduce a business scenario, discuss the tools that we will use to solve the problem(s), walk through the initial problem(s), and then provide you a self-guided exercise. We conclude this book with a capstone assignment in *NYC*.

The exercises will use data that can be download from http://www.useready.com/resources/a-practitioners-guide-to-alteryx by following the instructions on the website to unpackage the file.

Additional data will also be needed to install the *US 2010 Census SF1* and *USGS North America Map* packages from http://downloads.alteryx.com/data.html which we will start using in *Cultural Musings*. We will also be using the Solocast Datasets in the *Statistics in Alteryx* section.

Let us know what you think by emailing us at AlteryxBook@USEReady.com and we will try to incorporate reader requests going forward.

If interested in Alteryx training sessions or Alteryx consulting, visit http://www.useready.com/.

Best of Luck,

USEReady

Table of Contents

About USEReady

About Alteryx, Inc.

Acknowledgements

Letter from the CEO

Foreword

Preface

To the reader

CHAPTER 1 .... **1**

An Introduction to Alteryx **3**

1.1 What Is Alteryx Analytics Platform? *3*
1.2 Alteryx Gallery *4*
1.3 The Alteryx Interface *4*
1.4 The Interface *8*
1.5 File Menu *9*
1.6 Edit Menu *10*
1.7 View Menu *11*
1.8 Options Menu *12*
1.9 Help Menu *14*
1.10 Samples *15*
1.11 User Settings *16*
1.12 Toolbar *18*
1.13 Tool Palette *19*
1.14 Overview *22*
1.15 Results Window *23*
1.16 Properties Window *24*
1.17 Interface Designer *32*
1.18 Canvas *40*
1.19 Using Tools *42*
1.20 Insert Tool Menu *44*

CHAPTER 2 .... **46**

The Games **46**

2.1 Tools and Concepts *48*

2.1.1 Input Data ... 48
2.1.2 Browse ... 48
2.1.3 Output Data ... 49
2.1.4 Comment ... 49
2.1.5 Filter ... 50
2.1.6 Formula ... 50
2.1.7 Join ... 51
2.1.8 Running Total ... 51
2.1.9 Sample ... 52
2.1.10 Select ... 52
2.1.11 Sort ... 53
2.1.12 Summarize ... 53
2.1.13 Tool Container ... 54
2.1.14 Transpose ... 54
2.1.15 Union ... 55
2.1.16 Cross Tab ... 55
2.2 Freestyle ... 58
2.3 Let's Tidy Things Up ... 72
2.4 Modern History ... 81
2.5 Brains vs Brawns ... 91
2.6 How are we doing? ... 111

CHAPTER 3 ... 113

Unisex Baby Names ... 113
3.1 Tools & Concepts ... 114
3.1.1 Imputation ... 114
3.1.2 Multi-Field Formula ... 116
3.1.3 Multi-Row Formula ... 117
3.1.4 Text Input ... 117
3.1.5 Data Cleansing ... 117
3.2 Gender Swapped ... 118
3.3 What about Me? ... 136
3.4 What's In A Name? ... 142

CHAPTER 4 ... 143

The Direct Approach ... 143
4.1 Tools & Concepts ... 146
4.1.1 Append Fields ... 146

4.1.2 Auto Field 146
4.1.3 Date Time Now 146
4.1.4 Date Time 147
4.1.5 Directory 147
4.2 What's The Policy On That? 148
4.3 Where, Oh Where Have the Three Files Gone? 163

CHAPTER 5 165

Cultural Musing 165
5.1 Tools & Concepts 168
5.1.1 Allocate Input 168
5.1.2 Find and Replace 168
5.1.3 Join Multiple 169
5.1.4 Text To Columns 169
5.1.5 Random % Sample 170
5.1.6 XML Parse 170
5.2 Culturally Rich 171
5.3 Culturally Divided 193

CHAPTER 6 195

The Sport Report 195
6.1 Tools & Concepts 199
6.1.1 Charting 199
6.1.2 Create Points 199
6.1.3 Layout 200
6.1.4 Render 200
6.1.5 Report Footer 200
6.1.6 Report Map 201
6.1.7 Report Header 201
6.1.8 Table 202
6.1.9 Email 202
6.1.10 Image 202
6.1.11 Report Text 203
6.2 Major Sports Teams 204
6.3 Regional Sales Monthly Update 231
6.4 Football! 235

CHAPTER 7 237

Expensive Beauty Product **240**
7.1 Tools & Concepts *240*
7.1.1 Fuzzy Match 240
7.1.2 Record ID 240
7.1.3 RegEx 241
7.1.4 Unique 241
7.2 Expensive Beauty Products *242*
7.3 More Flags *266*

CHAPTER 8 .... **267**

Applications Wanted **269**
8.1 Tools & Concepts *270*
8.1.1 Action 270
8.1.2 Condition 270
8.1.3 Control Parameter 271
8.1.4 Date 271
8.1.5 Drop Down 272
8.1.6 Error Message 272
8.1.7 File Browse 273
8.1.8 Folder Browse 273
8.1.9 List Box 274
8.1.10 Radio Button 274
8.1.11 Text Box 275
8.1.12 Tree 275
8.2 Let's Build an App For That! *276*
8.3 To Summarize or Not to Summarize: That is the Question *301*

CHAPTER 9 .... **303**

Where's the Joe **303**
9.1 Tools & Concepts *305*
9.1.1 Macro Input 305
9.1.2 Macro Output 305
9.1.3 Map 306
9.1.4 Map Input 306
9.1.5 Numeric Up Down 307
9.1.6 Spatial Match 307
9.1.7 Trade Area 307
9.1.8 Create Point 308

9.1.9 Distance 308
9.1.10 Find Nearest 309
9.1.11 Spatial Info 309
9.1.12 CASS 310
9.1.13 Parse Address 310
9.1.14 Street Geocoder 311
9.1.15 US ZIP9 Coder 311
9.2 Where's the Joe? *312*
9.3 Find the locations *336*

CHAPTER 10 351

Meta-Morphosis 351
10.1 Tools & Concepts *354*
10.1.1 Check Box 354
10.1.2 Detour 354
10.1.3 Detour End 354
10.1.4 Dynamic Rename 355
10.1.5 JSON Parse 355
10.1.6 Message 356
10.2 All the Tools Along the Way *357*
10.3 The Analytical App *374*
10.4 Narrowing Down the Search *379*

CHAPTER 11 381

Let's Do It (In-DB) 381
11.1 Tools & Concepts *384*
11.1.1 Browse In-DB 384
11.1.2 Connect In-DB 384
11.1.3 Data Stream In 385
11.1.4 Data Stream Out 385
11.1.5 Filter In-DB 386
11.1.6 Formula In-DB 386
11.1.7 Join In-DB 386
11.1.8 Macro Input In-DB 387
11.1.9 Macro Output In-DB 387
11.1.10 Select In-DB 388
11.1.11 Summarize In-DB 388
11.1.12 Union In-DB 389

11.1.13 Write Data In-DB 389
11.1.14 Sample In-DB 390
11.2 Superstore Database Report 391
11.3 Superstore Database Report with Macro 399

CHAPTER 12 407

GREEN ON THE GO 407
12.1 Tools & Concepts 409
12.1.1 Download 409
12.1.2 Twitter Search 409
12.2 Green on the go 410
12.3 What is trending for Tesla? 427

CHAPTER 13 441

Meet the programmer in You 441
13.1 Tools & Concepts 444
13.1.1 Block Until Done 444
13.1.2 Dynamic Input 444
13.1.3 Dynamic Replace 445
13.1.4 Dynamic Select 445
13.1.5 Field Info 446
13.1.6 Run Command 446
13.1.7 Test 447
13.1.8 Throttle 447
13.1.9 JSON Build 448
13.2 Jeeves! What was the weather like at Chicago last year? 449
13.3 Fieldnames to JSON tree 462
13.4 How's the oil market doing? 467

CHAPTER 14 469

Statistics in Alteryx 469
14.1 Tools & Concepts 471
14.1.1 Basic Data Profile 471
14.1.2 Field Summary 471
14.1.3 Frequency Table 472
14.1.4 Pearson Correlation 472
14.1.5 Spearman Correlation 473
14.1.6 Decision Tree 473

14.1.7 Forest Model ... 474
14.1.8 Score ... 474
14.1.9 Support Vector Machine ... 475
14.1.10 ARIMA ... 476
14.1.11 ETS ... 476
14.1.12 TS Compare ... 477
14.1.13 TS Forecast ... 477
14.1.14 Linear Regression ... 478
14.1.15 Logistic Regression ... 478
14.1.16 Naïve Bayes Classifier ... 478
14.1.17 Stepwise ... 479
14.1.18 Append Cluster ... 479
14.1.19 K-Centroids Cluster Analysis ... 480
14.1.20 K-Centroids Cluster Diagnostics ... 481
14.2 Which car has the best performance? ... 482
14.3 Blood Donation ... 499
14.4 Boutique Investment Bank ... 509
14.5 Super Store Sales ... 514
14.6 Property Price Prediction ... 514

CHAPTER 15 ... 515

Self-Guided Solutions ... 517
15.1 The Games: How are we doing? ... 519
15.2 Unisex Baby Names: What's In a Name? ... 520
15.3 The Direct Approach: Where, Oh Where Have My Three Files Gone? ... 520
15.4 Cultural Musing: Culturally Divided ... 521
15.5 The Sport Report: FOOTBALL! ... 522
15.6 Expensive Beauty Products: More Flags ... 523
15.7 Applications Wanted: To Summarize or Not to Summarize ... 526
15.8 Where's the Joe? ... 528
15.9 Metamorphosis: Narrowing Down the Search ... 529
15.10 How's the oil market doing? ... 530
15.11 Super Store Sales ... 531
15.12 Property Price Prediction ... 532

Appendices ... 533

Appendix A – File Types ... 535

Appendix B – Hot Keys **536**
Appendix C – Downloads/Content **537**
Appendix D – Field Types **538**
Appendix E – Properties Window **540**
Appendix F – Boolean Expressions **541**
Appendix G – Data Components **542**
Appendix H – Date/Time Units **543**
Appendix I – RegEx Cheat Sheet **544**
Appendix J – Action Tool Sets **545**
Appendix K – Directory Tool Data **550**
Credited Original Data Sources **551**

CHAPTER 1

An Introduction to Alteryx

1.1 What Is Alteryx Analytics Platform?

The Alteryx Analytics Platform from Alteryx Inc. consists of two products: Alteryx Designer and Alteryx Server.

Alteryx Designer allows us to build a data stream (or module) and run it locally. In addition to a base package, Alteryx offers datasets and tool packages to expand the product to enhance computing capabilities. These packages permit additional features like drive-time analysis when TomTom data is available.

Alteryx Server has two main functions. The first allows users who have Alteryx Designer to publish applications for consumption. The second allows scheduled workflows to run without external intervention.

The *Alteryx Analytics Platform* has solved two major issues affecting the data preparation process: time to run the process and high specialization of labor. Typically, the preparation process is highly time-intensive because it is largely manual. While automation solves the time required to run, it still requires personnel with skills in computer programming. This means we need additional resources allocated to every project, eventually raising the project costs and often preventing projects from being taken on. Alteryx, Inc. has elegantly addressed both of these problems. It has designed an easy-to-use drag-and-drop interface where all we need to do is give each tool the appropriate settings. This simple concept has allowed business users to develop workflows that meet their exact needs much faster than has ever been possible because it provides business people with the tools to design or modify a fully or semi-automated data preparation.

1.2 Alteryx Gallery

The *Alteryx Gallery* is a public version of *Alteryx Server* where the community can upload interesting data streams, share ideas, and get inspired. The *Alteryx Gallery* can be accessed at gallery.alteryx.com.

1.3 The Alteryx Interface

By the end of the chapter, the reader should be able to understand all aspects of the Alteryx Designer Interface and how to work with the product.

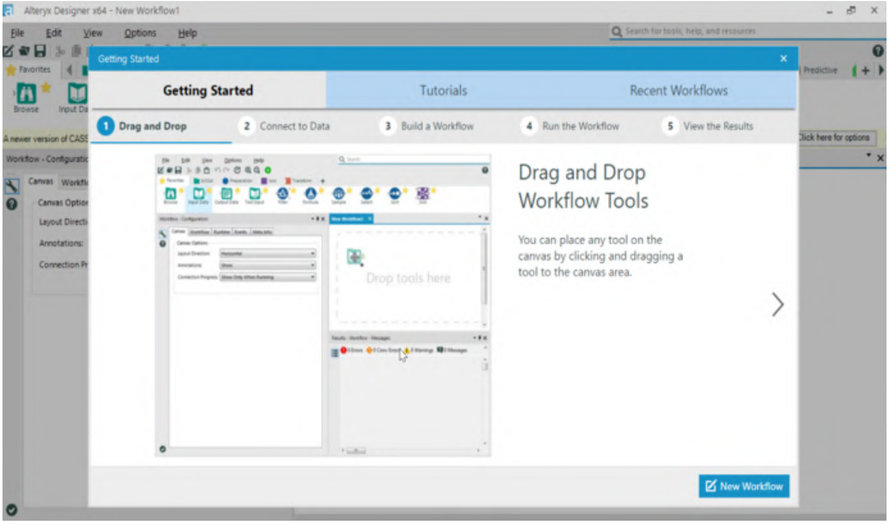


Figure-1-1-Alteryx Interface

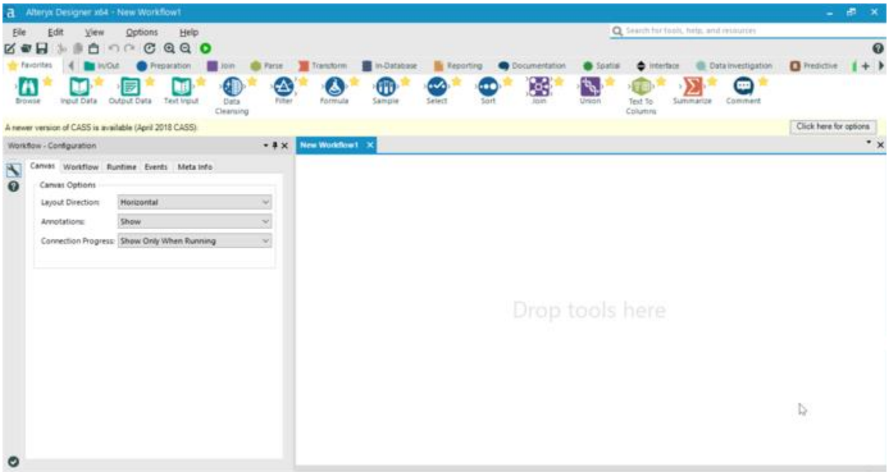


Figure-1-2-Workflow Configuration

Above are the two things we may see when we open Alteryx Designer. The first image is the default for when the program is first opened. Alteryx Designer asks us what we want to do with the session. These options, fairly clearly, are to open up the basic tutorials, open an existing module, or open a new module. The second image is what comes up when we have previously selected the option *Don't show this again and always open a blank canvas*. When using this option, Alteryx Designer always opens a new module. (For those who see the first image, please select "New Workflow."

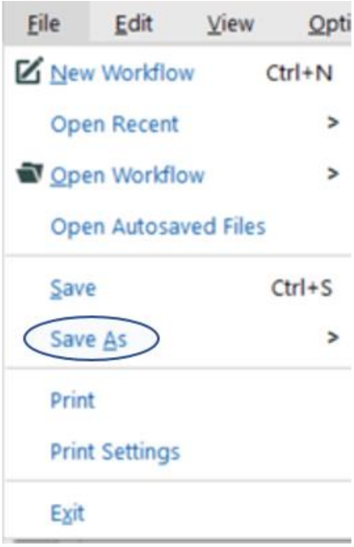


Figure 1-3 - File Dropdown

The first thing we will do is save our module. Open the file drop-down at the top left of the screen and select the *Save As* option as indicated above.

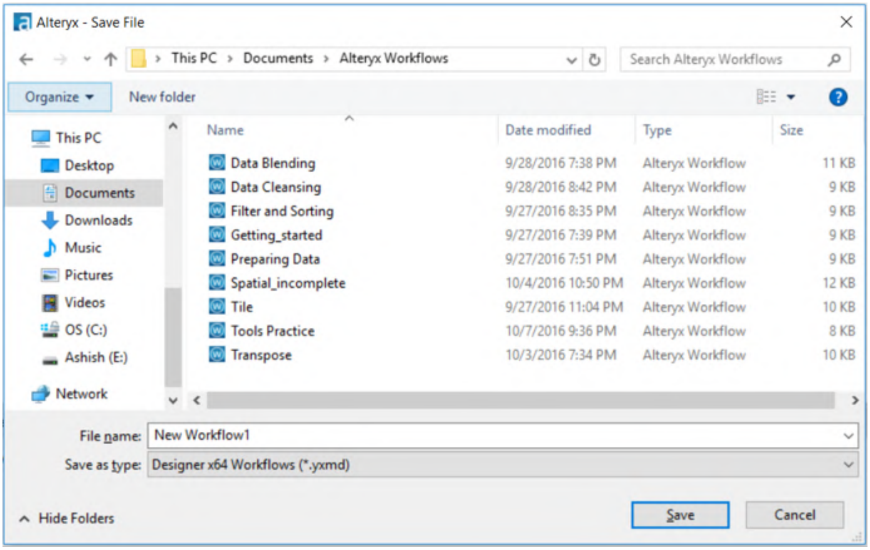


Figure 1-4 - Save As

We can save our data stream as one of three file types. At a high level, we have the ability to save our data streams as:

- .yxmd

  Files saved in this format are data streams in a standard *Alteryx Workflow*.

- .yxwz

  Files saved in this format are data streams that have been designed as *Analytic Applications*, which means they have user inputs and can be run on *Alteryx Server*.

- .yxmc

  Files saved in this format are data streams that have been designed as *Macros* that allow the reuse of the data stream.

Please save this workflow as *The Basics.yxmd*

1.4 The Interface

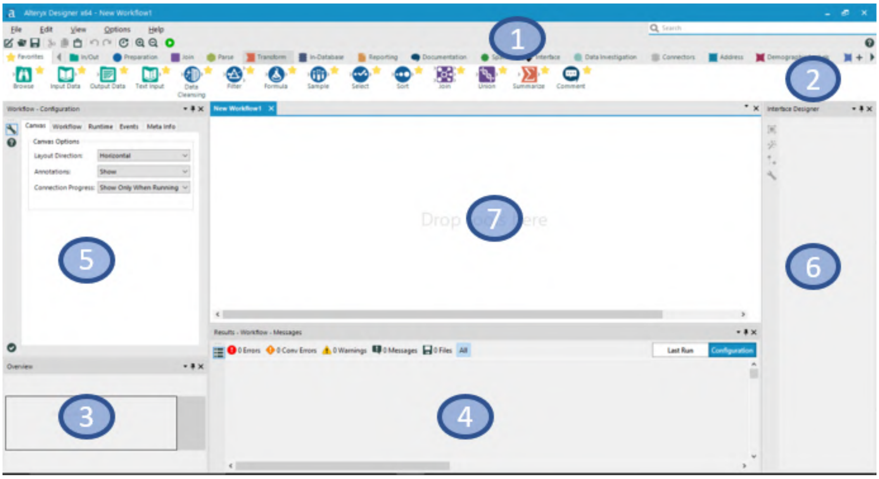


Figure 1-5 – Alteryx Designer Interface

This is one potential view of the *Alteryx Designer Interface* if every *View* option is on. The numbers represent each of the seven primary aspects of the interface. One through six can be disabled and will be discussed under the corresponding sections in *View*. The seventh is the canvas, where we build workflows.

1.5 File Menu

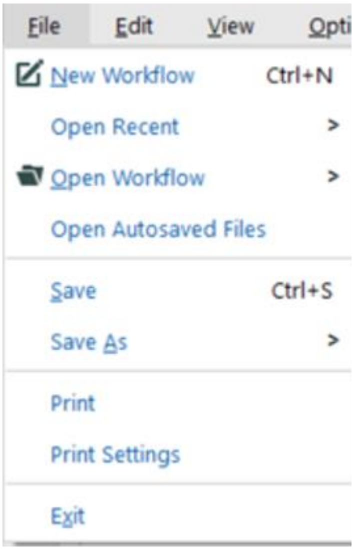


Figure 1-6 – File Menu

The *File Menu* functions in the same way as most programs, with a few specialized options:

New Workflow: Opens a blank canvas to work in. (*Ctrl+N* means that there is a *Hot Key* that allows us to use this action without opening the drop-down. Pressing the *Ctrl* and *N* buttons at the same time does the same thing as selecting *New Workflow*.) (Hot Key: Ctrl+N)

Open Recent: Opens a list of the most recently opened workflows.

Open Workflow: Opens a file browser that allows us to open an existing *Workflow*, *Analytic App*, or *Module* in a tab of the canvas. (Hot Key: Ctrl+O)

Open Autosaved Files: Unknown to many, by default Alteryx saves your open workflows every 10 minutes, keeps the last three iterations of them, and keeps them around for 30 days since the last save. To edit the Autosave settings, go to 'Advanced' tab under Options > User Settings > Edit User Settings.

Save <File Name>: Saves the workflow that is in the active canvas. (We will see the name of the current workflow instead of *<File Name>)* (Hot Key: Ctrl+S)

Print: Prints a copy of the workflow.

Print Setup: Opens a window that allows us to configure how the workflow will print if we choose to print out a copy.

Exit: Closes Alteryx Designer entirely. This differs from *Close* because *Close* will close the active canvas but *Exit* will close every open data stream.

1.6 Edit Menu

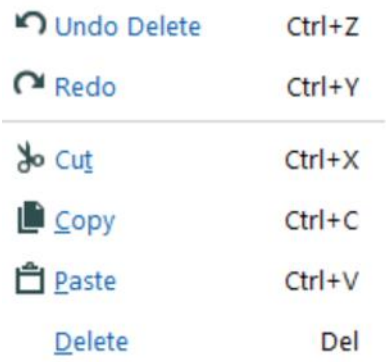


Figure-1-7-Edit Menu

Undo: Allows us to undo the steps just performed, in reverse order. (Hot Key: Ctrl+Z)

Redo: Allows us to restore a step from what was undone, as long as we have not changed anything else. (Hot Key: Ctrl+Y)

Cut: Allows us to remove the selected items on the canvas while keeping a copy on the clipboard. (Hot Key: Ctrl+X)

Copy: Allows us to copy the selected items on the canvas to the clipboard. (Hot Key: Ctrl+C)

Paste: Allows us to take the most recent thing copied or cut into the clipboard and put it on the canvas. (Hot Key: Ctrl+V)

Delete: Removes the selected items on the canvas. (Hot Key: Del or Delete)

1.7 View Menu

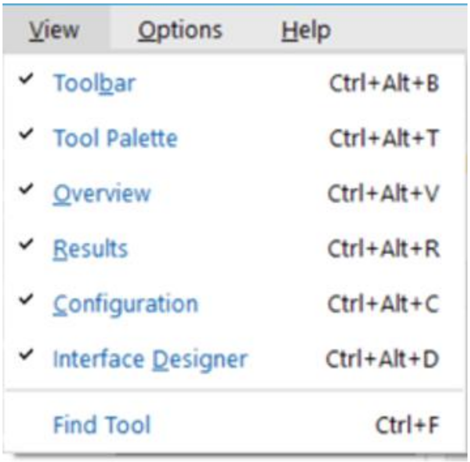


Figure 1-8 – View Menu

Toolbar: Turns the toolbar on and off. When it is checked, the buttons on the toolbar are visible (1 in Figure 1-5). (Hot Key: Ctrl+Alt+B)

Tool Palette: Turns the tool palette on and off. When checked, we will see the two rows of drag-and-drop tools (2 in Figure 1-5). (Hot Key: Ctrl+Alt+T)

Overview: Turns the overview window on and off. When checked, we can see a high-level overview of our canvas (3 in Figure 1-5). (Hot Key: Ctrl+Alt+V)

Results: Displays the Results window docked in its last visible position. (Hot Key: Ctrl+Alt+R)

Configuration: Displays the Configuration window docked in its last visible position. (Hot Key: Ctrl+Alt+C)

Interface Designer: Turns the interface designer window on and off. When it is checked, we will see a window that allows us to modify the user interface (6 in Figure 1-5). (Hot Key: Ctrl+Alt+D)

Find Tool: Opens a pop-up window that allows us to quickly find particular tools on the canvas. (Hot Key: Ctrl+F)

1.8 Options Menu

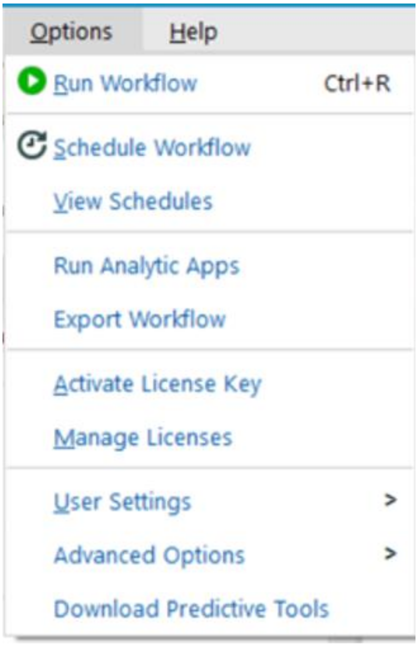


Figure 1-9 – Options Menu

Run Workflow: Runs the workflow. Ctrl+R will also run the workflow. The icon changes to Cancel Workflow while the workflow is running.

Schedule Workflow: Allows us to set up a schedule to run our workflow as long as we have Alteryx Server.

View Schedules: Allows us to look at all of the scheduled jobs that we have access to on Alteryx Server.

Run Analytic Apps: Allows us to run Alteryx Analytical Apps that we have created.

Export Workflow: Allows us to export the workflow along with the packages that have been created.

Activate License Key: Opens a window that asks for an activation key so we can use Alteryx Desktop.

Manage Licenses: Opens a window that allows us to look at historical license keys and see what we currently have available.

User Settings: Opens a window that allows us to customize the way the canvas looks and the way some of the default settings behave.

Advanced Options: Allows us to manage aliases, workflow dependencies, and encrypt workflows.

Download Predictive Tools: Allows us to download tools for the purpose of predictive analytics from the Alteryx repository.

1.9 Help Menu

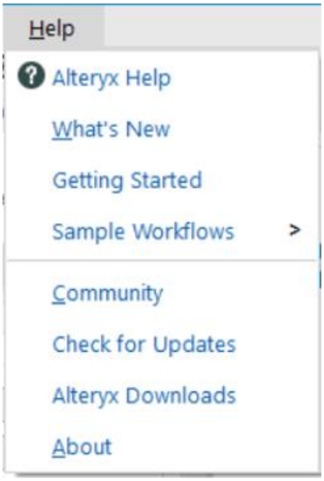


Figure 1-10 – Help Menu

Alteryx Help: Opens the Alteryx Help home page.

What's New: Opens the Alteryx Help release notes for the version of Alteryx that we are currently using.

Getting Started: Opens the Alteryx Getting Started page that is loaded in the beginning.

Sample Workflows: Allows us to access the Sample Workflows that are shipped with Alteryx.

Community: Opens the Alteryx Community web page.

Check for Updates: Checks our current version of Alteryx Designer and verifies if it is the most recent version.

Alteryx Downloads: Opens the Alteryx Download page where we can find additionally downloadable content. One example is the U.S. Census data.

About: Opens a window that tells us version, publisher, and serial number information about Alteryx Designer

1.10 Samples

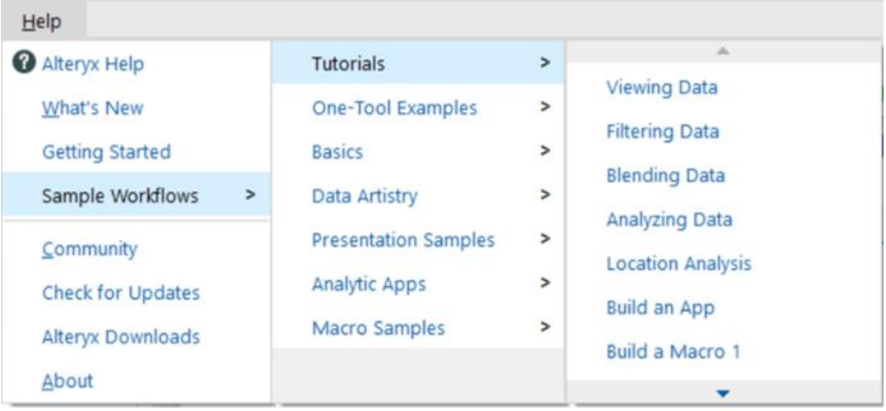


Figure 1-11 – Samples

Under the *Help Menu,* there is the option to open Sample Workflows. Depending on what we have downloaded from Alteryx, we will see different samples.

These samples are good starting points to get used to Alteryx and to learn how to use tools that we do not know.

Walking through the samples is beyond the scope of this book, as all the samples are well documented and show how to build the workflow step by step. However, below is a picture of what the *Viewing Data* sample looks like before you begin to work with it.

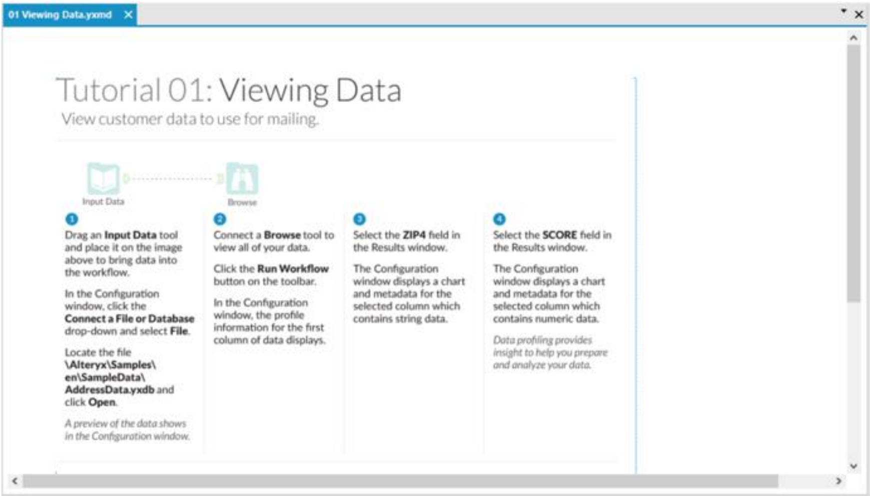


Figure 1-12 – Sample Tutorial

1.11 User Settings

Under the *Options menu,* there is an option called *User Settings*. This sub-menu has four options.

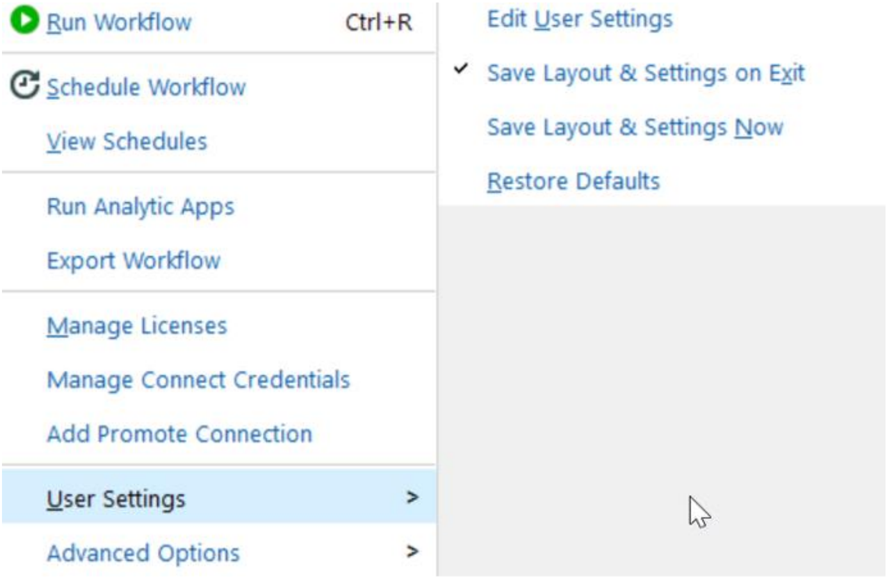


Figure-1-13-User Settings

Edit User Settings: Opens a pop-up window for the user to edit the settings affecting the way Alteryx Designer looks and behaves.

Save Layout & Settings on Exit: Allows Alteryx to copy the settings we had when Alteryx was last closed as the default when we next open the program. Unchecking the option can turn off this functionality.

Save Layout & Settings Now: Allows us to tell Alteryx to use the settings we currently have as the default behavior for the program. This is most effectively used when we are not using *Save Layout & Settings on Exit*.

Restore Defaults: Allows us to revert Alteryx Designer to the default settings.

1.12 Toolbar

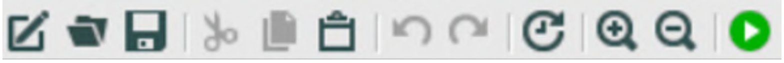


Figure 1-14 – Toolbar

The *Toolbar* has buttons we can use to do the following tasks. It is the same as if we were to navigate to the parallel options in the menus described above. The buttons simply make it easier to perform the most common actions.

 Allows us to create a new workflow. (Same as File > New Workflow) (Hot Key: Ctrl+N)

 Allows us to open a workflow. (Same as File > Open Workflow) (Hot Key: Ctrl+O)

 Allows us to save our current workflow. (Same as File > Save <File Name>) (Hot Key: Ctrl+S)

 Allows us to cut our current selection. It is grayed out and inaccessible when we have nothing selected and dark when we do. (Same as Edit > Cut) (Hot Key: Ctrl+X)

 Allows us to copy our current selection. It is grayed out and inaccessible when we have nothing selected and dark when we do. (Same as Edit > Copy) (Hot Key: Ctrl+C)

 Allows us to paste what we have on the clipboard. If we do not have something on the clipboard that can be pasted into Alteryx, we will get a pop-up that says *Nothing to paste.* (Same as Edit > Paste) (Hot Key: Ctrl+V)

 Allows us to undo what we just did. It is grayed out when we have nothing to undo. (Same as Edit > Undo) (Hot Key: Ctrl+Z)

 Allows us to redo what we just undid. It is grayed out when we have nothing that we can redo. (Same as Edit > Redo) (Hot Key: Ctrl+Y)

 Allows us to schedule the workflow on Alteryx Server. (Same as Tools > Schedule Workflow)

 Allows us to zoom in on our workflow. (Same as View > Zoom In)

 Allows us to zoom out on our workflow. (Same as View > Zoom Out)

 Allows us to run the workflow. (Hot Key: Ctrl+R)

 Allows us to cancel the workflows while running. (Hot Key: Ctrl+R)

1.13 Tool Palette

Figure 1-15 – Tool Palette

The tool palette is one of the most important aspects of the Alteryx Designer interface. It is the easiest way to bring the tools onto our canvas. In the later section below called *Canvas*, we will discuss the alternative method.

The three main elements of the tool palette are:

- *Search*: Located in the top-right corner of the tool palette, this element allows us to type something into the bar, and every tool associated with that word is displayed in the Tool Selection.
- *Add/Remove Tools* : Located to the far right, below *Search* and next to the *Tool Categories*, this element allows us to access/add/remove all the tools grouped into categories that Alteryx has defined. We can add to this list by developing macros, which we will discuss toward the end of the book.
- *Tool Selection*: This comes as a drop-down when we type in the search box and shows all tools meeting either the criteria of the *Search* or *Tool Category* that we have selected.

There is one special category in the *Tool Categories* called *Favorites*. It is special because it does not have tools of its own. Instead, we can add and remove tools from this list by clicking the star at the top-right corner of each tool. When it is yellow, it means that it will show up in the *Favorites Category*. When we cannot see the star unless we point our mouse at the tool, and it shows up gray, that means that it is not in the *Favorites Category*.

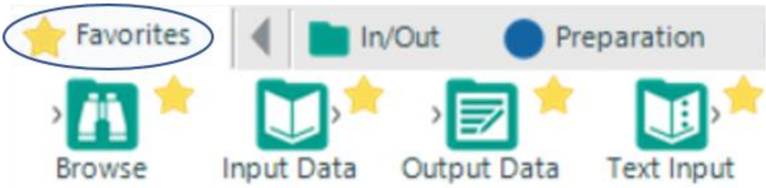


Figure 1-16 – Favorites

The arrows on either side of the *Tool Categories* offer the ability to scroll across to see each of the categories. We see that, by default, *Favorites* doesn't follow this behavior. That is because it is pinned outside the scroll arrows. We can do this with any of the categories by right-clicking on them and selecting *Pin <Category Name>*. See the example below.

Right click on In/Out:

Figure 1-17 – Pin In/Out

Select Pin In/Out:

Figure 1-18 – Selecting Pin In/Out

In/Out is pinned:

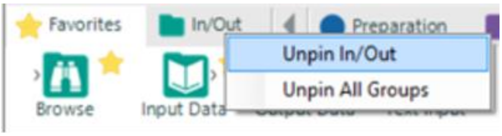


Figure 1-19 – Unpin In/Out

Select Unpin In/Out:

Figure 1-20 – Tool Categories

We can see that certain icons show up on the *Tool Category* names. That is because each of the categories has a *Default Tool*. If we were to click on the *Tool Category* and drag it to the canvas, the tool in the icon is what we bring onto the canvas.

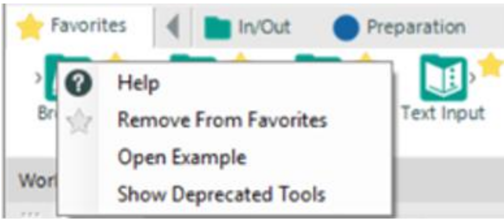


Figure 1-21 – Tool Categories

We can see in Figure 1-21 an option called *Show Deprecated Tools*. Selecting this option shows tools that belong to outdated versions of Alteryx. If we need a former functionality, we can access it here.

Figure 1-22 – Additional Tool Categories

All of these categories may not be available to all users because Alteryx has many additional packages that we can download in addition to the basic package. One example is if we had not chosen to download the Predictive package when we downloaded Alteryx Designer, we would not see the *Predictive Category.*

We will only cover the basic package in this book, plus some downloaded content.

1.14 Overview

Below is a picture of the canvas and overview window with a simple data stream that is too wide to see what is happening in the view of the canvas. The right side of the image is the *Overview* window, and it shows the view of the canvas in relation to the entire workflow.

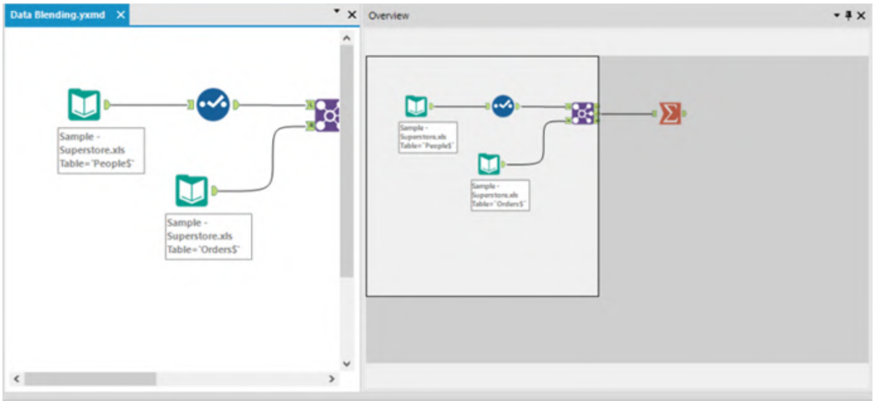


Figure 1-23 – Overview Window

In addition to showing where we are, the overview window makes it easy to find the part of our program that we are interested in because we can move the view of the canvas by clicking and dragging the white box in the overview window.

1.15 Results Window

Every time we run a workflow, we will get a series of returns that lets us know things about the run. One example of this is below.

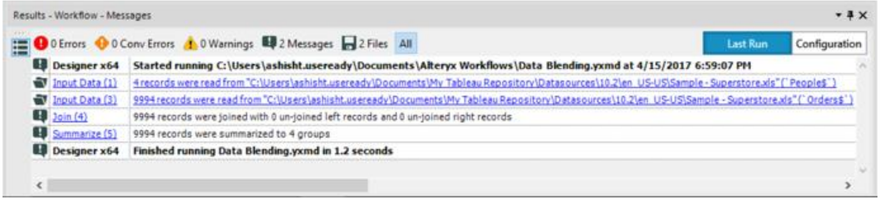


Figure 1-24 – Results Window

In this run, we can see that we had 0 Errors, 0 Conversion Errors, 0 Warnings, 2 Messages, and 2 Files. The counts here reflect the eight notes we see about the run in the text portion of the Output window. We have six notes instead of the four we would expect from the counts of the messages. This is because the first note is that the workflow started to run, and the last note is that the module had finished the run. Even if we have no issues or messages during the run, we will still have these two notes because they are always written. The last message changes depending on whether or not we have any errors during our execution. The output window is important in diagnosing issues with the data stream.

- *Errors*: These are critical failures that will terminate the data stream they are in.
- *Conversion (Conv) Errors*: These are potentially problematic issues that stem from a data mismatch when converted from one format to another.
- *Warnings*: These are things we should be aware of but could be expected, so the results are printed in the Output for us to know about.
- *Messages*: These are text in the program that are designed to print when certain stages of the data stream have been finished or certain conditions were met.
- *Files*: These are links that will allow us to open any file created during the running of the workflow

1.16 Properties Window

The properties window is where we will spend the vast majority of our time. Every tool has a different properties window, and we will discuss each of them as we discuss the individual tools. Here, we will discuss the properties window of the workflow and connections.

There is a pane on the left side of every properties window that has a universal set of symbols. We will discuss each of these as they first appear in this book. A list of all of these symbols is in Appendix E.

Workflow configuration

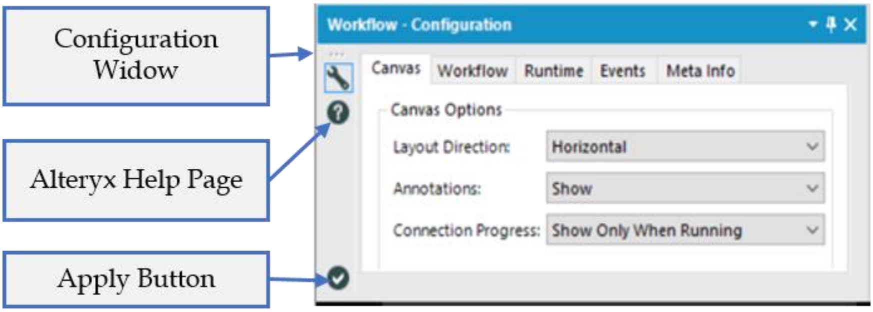


Figure 1-25 – Workflow Configuration – Canvas Tab

Notice the button that looks like a wrench at the top-left of each of these properties windows. Selecting the wrench shows the *configuration window* so that we can modify the settings of the tool. As we discuss other tools, the importance here will become apparent.

A click on  symbol opens the Workflow Alteryx Help page.

A  symbol at the bottom-left corner is an apply button. We can click this to apply the changes we have made. We do not need to click this button; we can just click somewhere on the canvas.

The first tab of the *Workflow configuration* window is Canvas. We can change three things in this window.

- *Layout Direction* is a drop-down list that allows us to rotate our data stream from a horizontal layout, which is the default, to a vertical layout, which is the layout for older versions of Alteryx. We must set either *Horizontal* or *Vertical* and cannot use a combination of the two.

- *Annotations* is a drop-down list that allows us to change what the text boxes under Tools say. When this option is set to *Hide,* the text does not show. When this option is set to *Show*, the default text is shown for each tool. When this option is set to *Show w/ Tool Names,* the annotations tell us what the tool names are as well as the default text.
- *Connection Progress* is a drop-down list that allows us to change when we see the progress labels on each of the tools. By default, this is set to *Show Only When Running*, but we can also set it to *Hide* and *Show*, which will never or always show the last run's progress respectively.

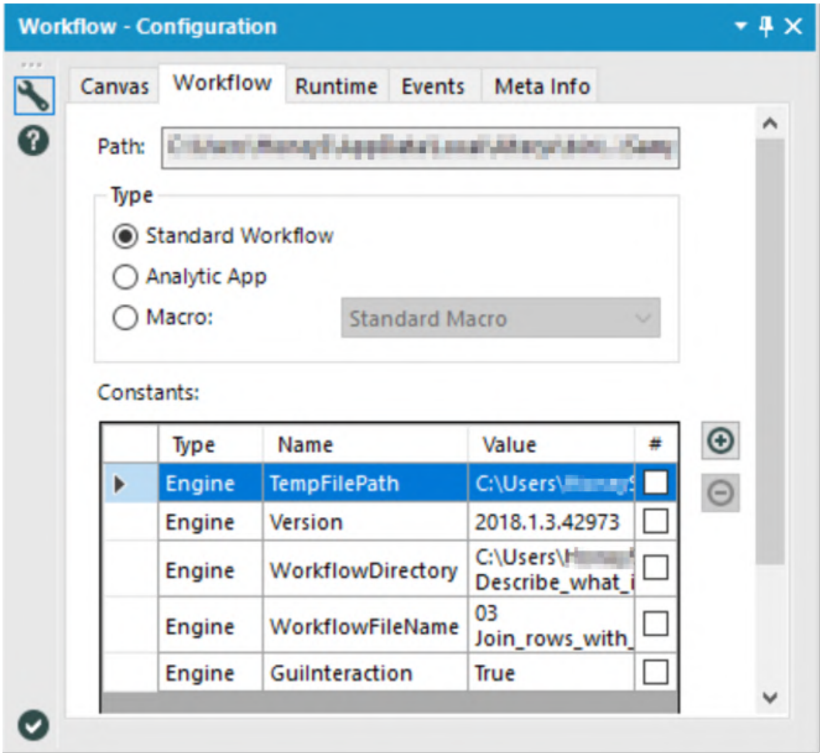


Figure-1-26-Workflow Configuration-Workflow Tab

The next tab is *Workflow*.

- The *Path* (file location) shows up in this box when the file has been saved.

- The *Type* allows us to choose which format the file should be. This is also where we select the type of *Macro* we want to create. (This will be discussed in *The Mermaid Coffee Company* chapter.)
- The *Constants* section allows us to define variables that we can use throughout our Alteryx Module.

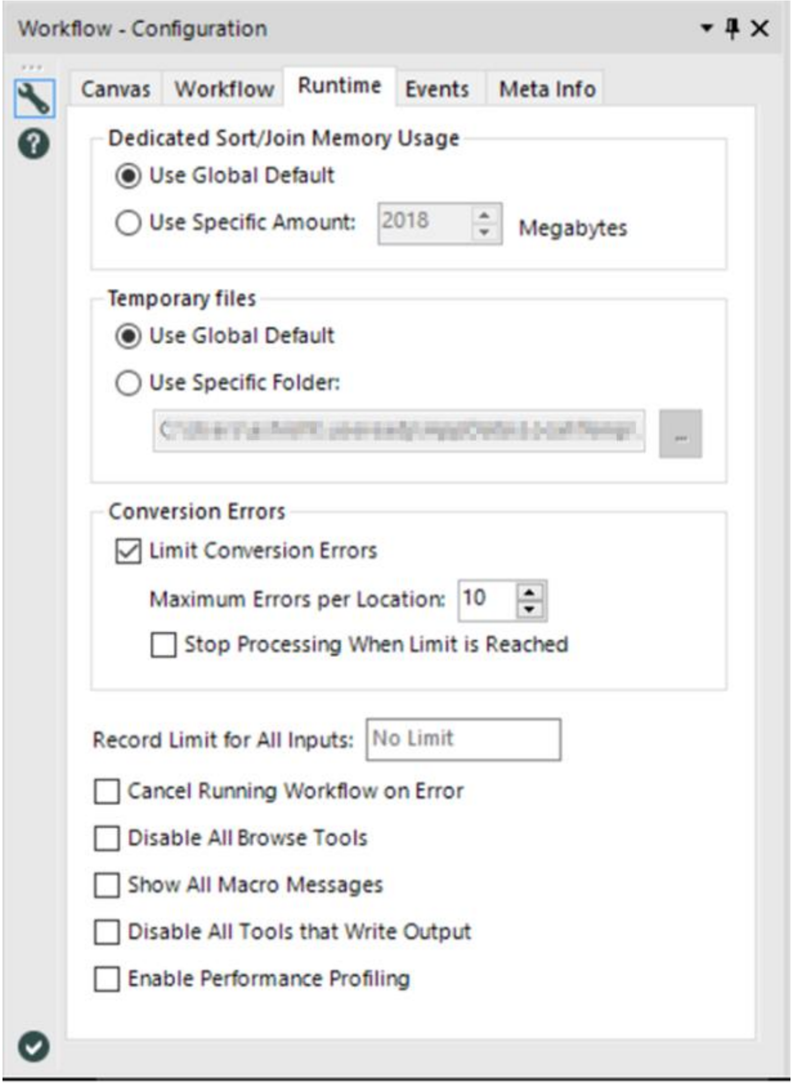


Figure 1-27 – Workflow Configuration – Runtime Tab

The *Runtime* tab allows us to configure settings that impact the way that Alteryx runs.

- The *Dedicated Sort/Join Memory Usage* allows us to set the maximum amount of memory that Alteryx is allowed to use for the *Sort* and *Join* functions, which can be highly memory-intensive.
- The *Temporary Files* setting allows us to define the folder that our temporary files are written to.
- The *Conversion Errors* setting allows us to change the way the workflow behaves if it experiences conversion errors, including limiting them from showing up for each tool and stopping the workflow if they occur.
- The *Record for All Inputs* setting allows us to set a maximum number of records read from any of our input files.
- The *Cancel Running Workflow on Error* is useful if we are trying to diagnose a specific issue in a workflow.
- The *Disable All Browse Tools* should be set after we finish testing and the process is in production. We want to do this because browse tools are slow to create and unnecessary unless testing. It is better to use this than to delete all of the tools because it will make modifying the workflow easier later.
- *Show All Macro Messages* allows us to see the messages we are getting in the macro tools so that when we are running a workflow, we can see everything that may be an issue.
- *Disable All Tools that Write Output* is useful for testing because we can turn off only the outputs so we are not rewriting the data every time we run the workflow.

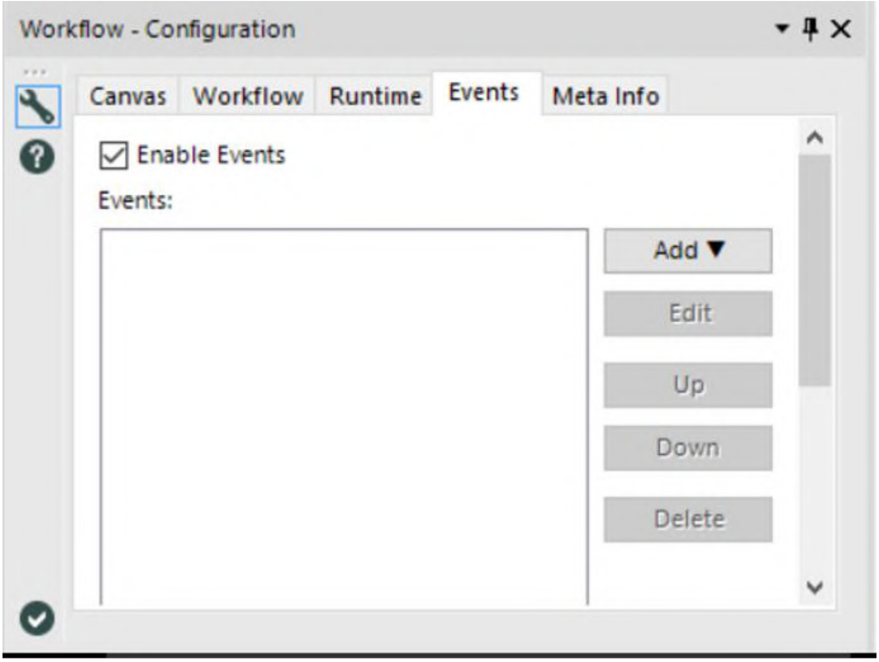


Figure 1-28 – Workflow Configuration – Events Tab

The *Events* tab allows us to define events that will allow secondary actions to take place.

When we add an event, we can choose to define a command line code or send an email when a condition is met. This is useful for security or maintenance because we can set up our workflows to let us know anytime the workflow ran or anytime it ran with issues.

The *Meta info* tab allows us to define meta information about our workflow. This is important when we are publishing a workflow to Alteryx Server or the public Alteryx Gallery.

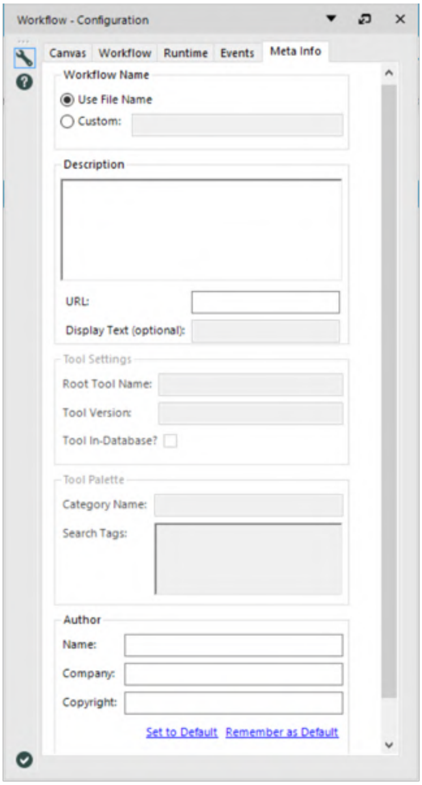


Figure 1-29 – Workflow Configuration – Meta Info Tab

- *Workflow Name* allows us to customize what we want the workflow to be called (independent of the file name) so that versioned workflows can have the same name, as far as the end user is concerned.
- The *Description* is a place for us to describe what the workflow does.
- The *URL and Display Test* fields allow us to identify where the module is published.
- The *Author* section allows us to publish information about us and our company along with the workbook.

Connection configuration

When using the default settings, connections are the curved black lines that connect one tool to the next.

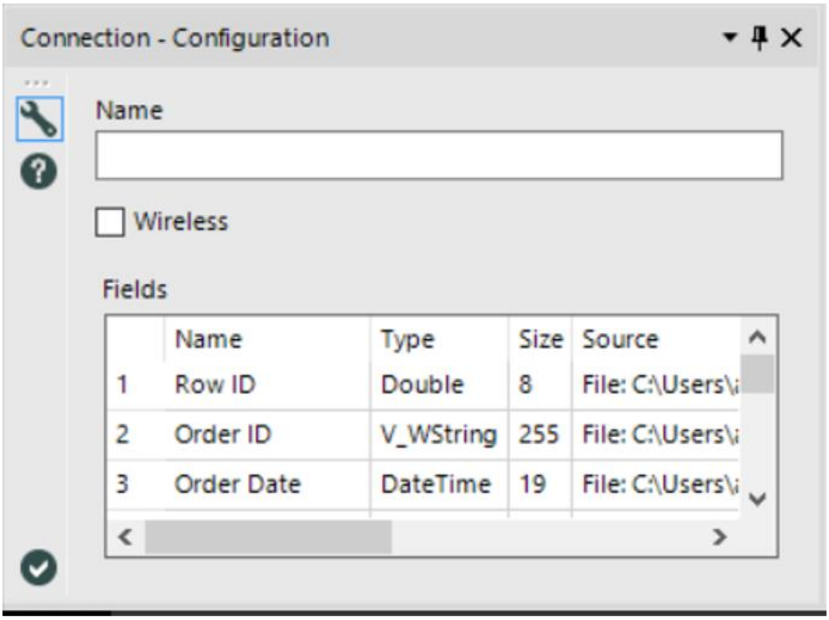


Figure 1-30 – Connection Configuration

The *Connection configuration window* has three components.

- *The name* allows us to rename our data connection. This is most useful when we have multiple data connections going into the same tool connector.
- The *Wireless* checkbox allows us to make the connection wireless (invisible) unless one of the tools it connects is selected.
- The *Fields* box allows us to see some of the metadata about each field in the data stream in this connection.

1.17 Interface Designer

The Interface Designer is a window used in developing apps and macros. This is where we get to design the user interface the end user will see.

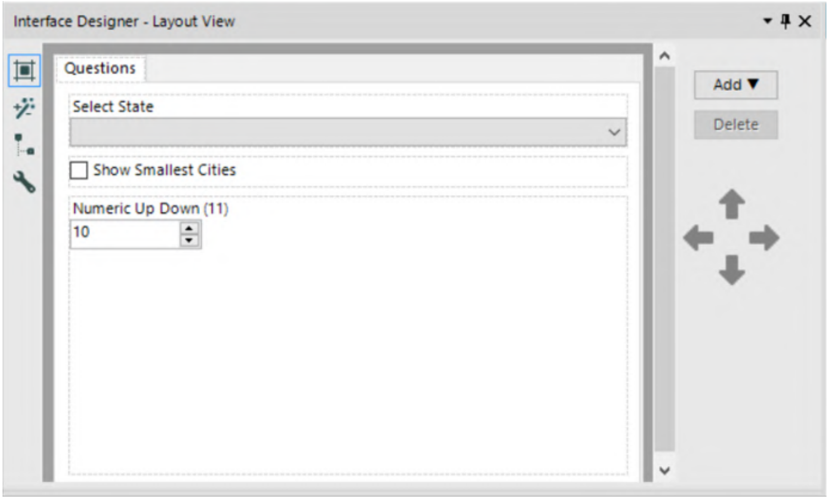


Figure 1-31 – Interface Designer – Layout View

The default tab view of the Interface Designer window is of the Layout View, which allows us to design the user interface in a graphical format.

Figure 1-32 – Interface Designer Settings

The Interface Designer, like the Properties window, has a sidebar that tells you what you are looking at. From top to bottom, the icons are:

- *Layout View*, where we design the interface graphically.
- *Test View*, where we see what the end user would see.
- *Tree View*, where we look at the structure of our questions in a tree structure
- *Properties*, where we set properties associated with the interface (this is independent of the Properties window itself).

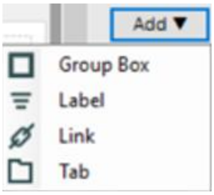


Figure 1-33 – Interface Designer Dropdown

In the *Layout View,* we design the interface and thus have a significant amount of flexibility in what we can do.

The first and most intuitive thing to do is rearrange questions in the order they need to be asked from the end user. We can do this by clicking the up and down arrows on the right-hand side. We may notice that while we are moving questions, they sometimes move within a box that frames the question we were trying to pass. This is because we can make one question determine if the other is also asked. By leaving a question inside another, we make that question dependent.

The next thing we can do is select the Add drop-down menu and see that we can add any of four things. Those things are:

- *Group Box*: a text element that we can put questions in.
- *Label***:** a text element that we cannot put questions in.
- *Link*: a hyperlink in the interface.
- *Tab*: a tab in the interface.

When we are working with the *Interface Designer*, we will be opening the Properties windows associated with whichever tool we have selected at the time. The following are the properties windows of the four objects described above.

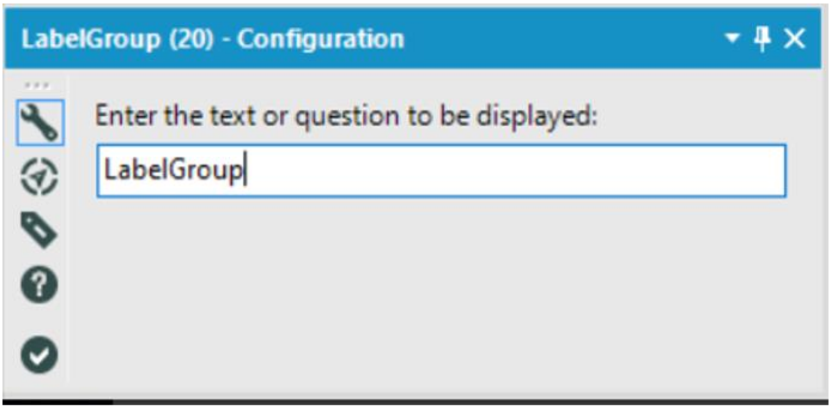


Figure 1-34 – Label Group Configuration

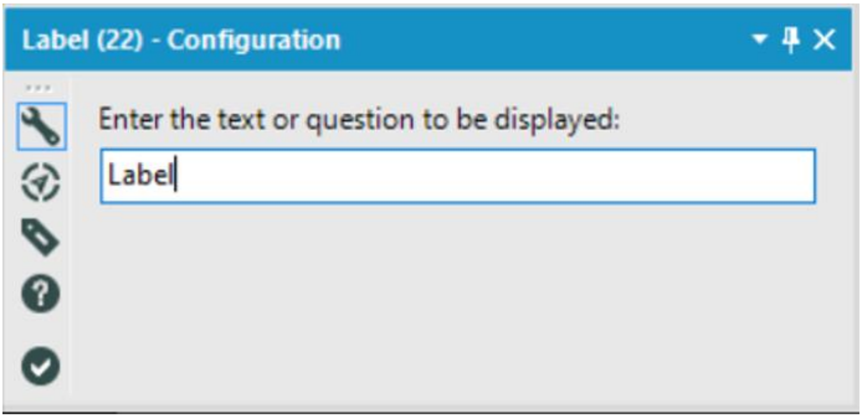


Figure 1-35 – Label Configuration

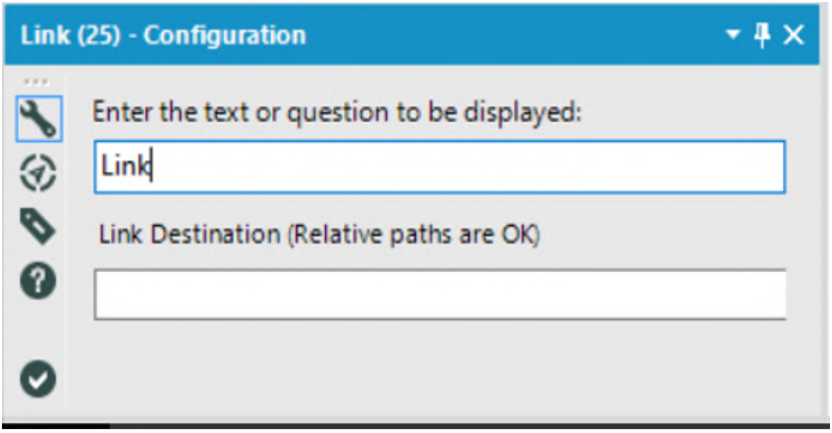


Figure 1-36 – Link Configuration

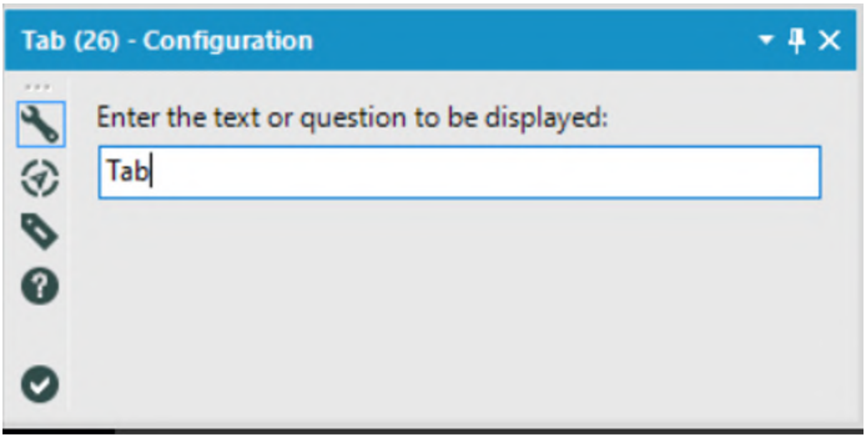


Figure 1-37 – Tab Configuration

As we can see, the four windows are similar. All four ask what we want the element in the interface to be called, and *Link* also asks us what link address we want to add to the interface. In addition to these questions, we see for the first time that we have three new icons in the sidebar of the Properties window.

Figure 1-38 – Sidebar Icons

- *Navigation*: Allows us to move directly to the incoming or outgoing tools.
- *Annotation*: Allows us to change how this tool is named and how its annotation behaves.

Navigation and *annotation* are part of every tool properties window, and they always behave the same.

Below are examples of a *Check Box* Navigation and Annotation window when connected to a tool.

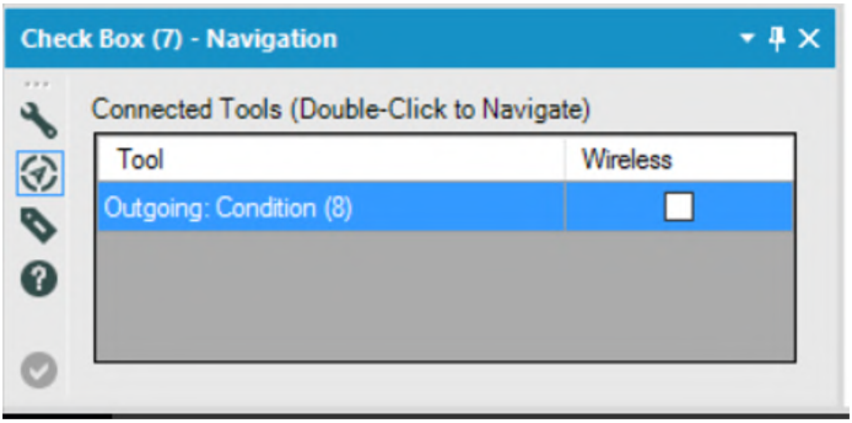


Figure 1-39 – Checkbox Navigation

We can see that *Navigation* lists the tool that this *Check Box* is connected to, Action (20), and tells us that it is an outgoing connection. That means that the Action is downstream of the Check Box. If the Wireless box were checked, the wire would be invisible. We will discuss both tools, as well as wireless connections, in chapters *Applications Wanted* and *Meta-morphosis*.

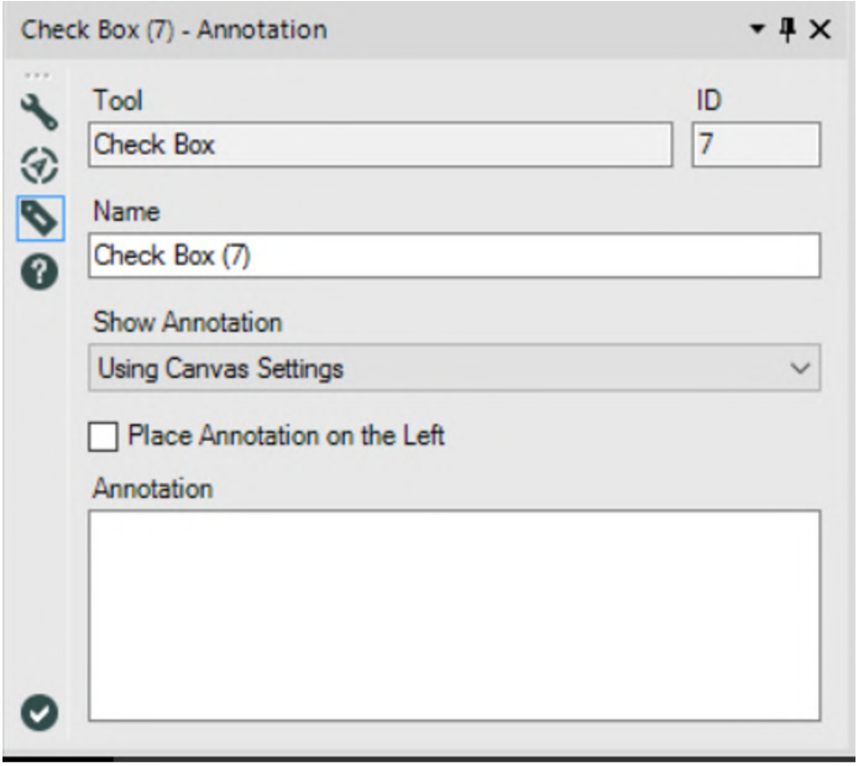


Figure 1-40 – Checkbox Annotation

The Annotation window tells us that the tool is a Check Box and that it has the ID number 7 (which means it will try executing this tool third if there are not outside influences). We then can change the name of the tool. We can change the annotation setting specific to this tool as well as provide a custom annotation (this will show up under the tool in the data stream).

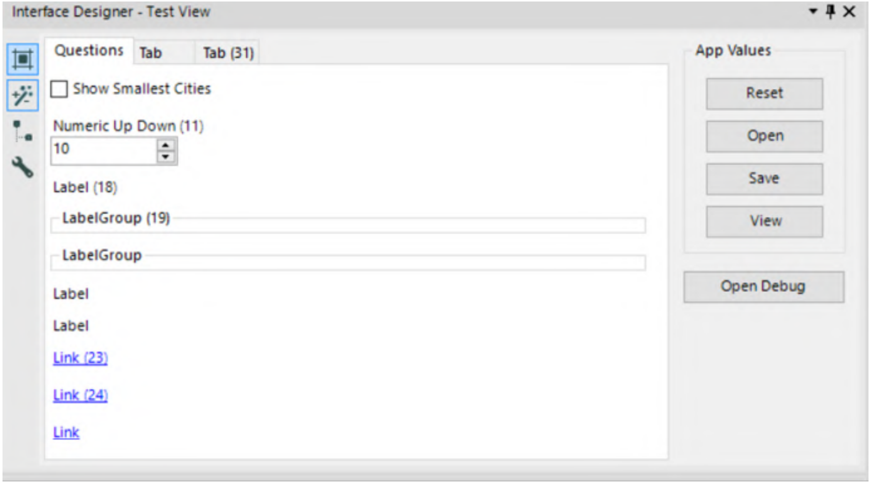


Figure 1-41 – Test View

Looking back at the Interface Designer, *Test View* allows us to answer the questions in the same way that the end user would so we can test the app.

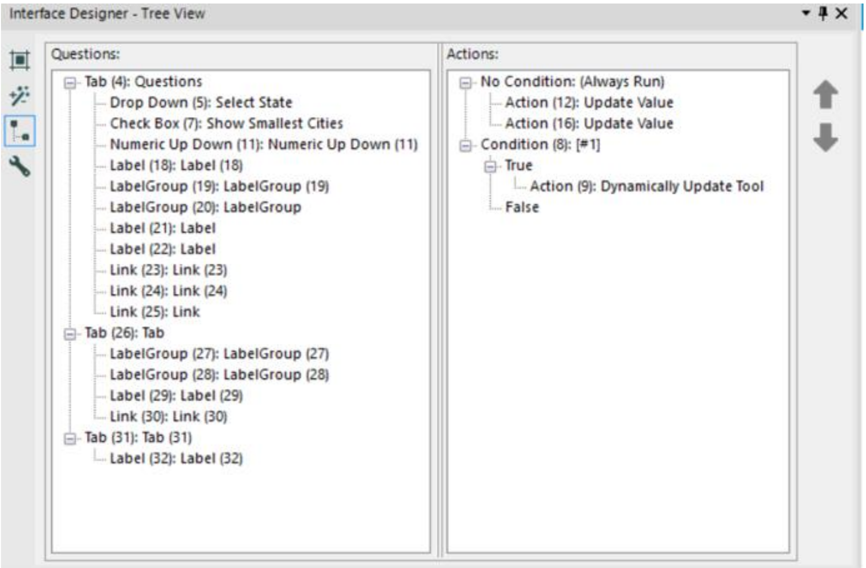


Figure 1-42 – Tree View

TreeView allows us to look at our user interface as a tree structure so it's easier to make sure our logic is properly grouped.

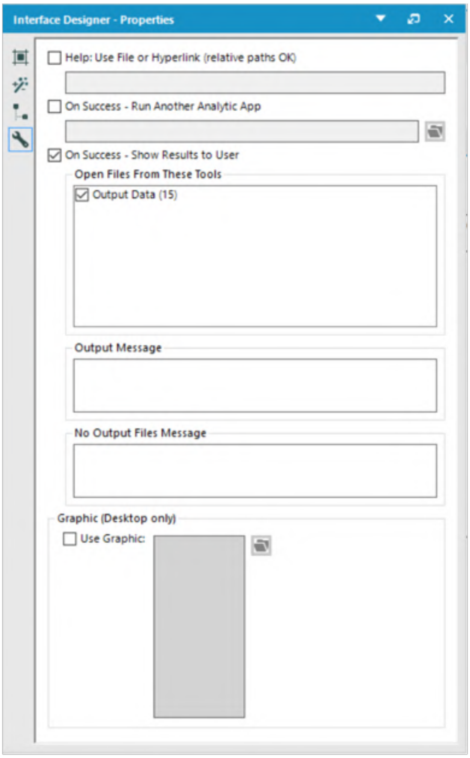


Figure 1-43 – Interface Designer Properties

Interface Designer Properties allows us to customize many aspects of our app or macro.

1.18 Canvas

This is what the default canvas looks like before we have brought any tools onto it.

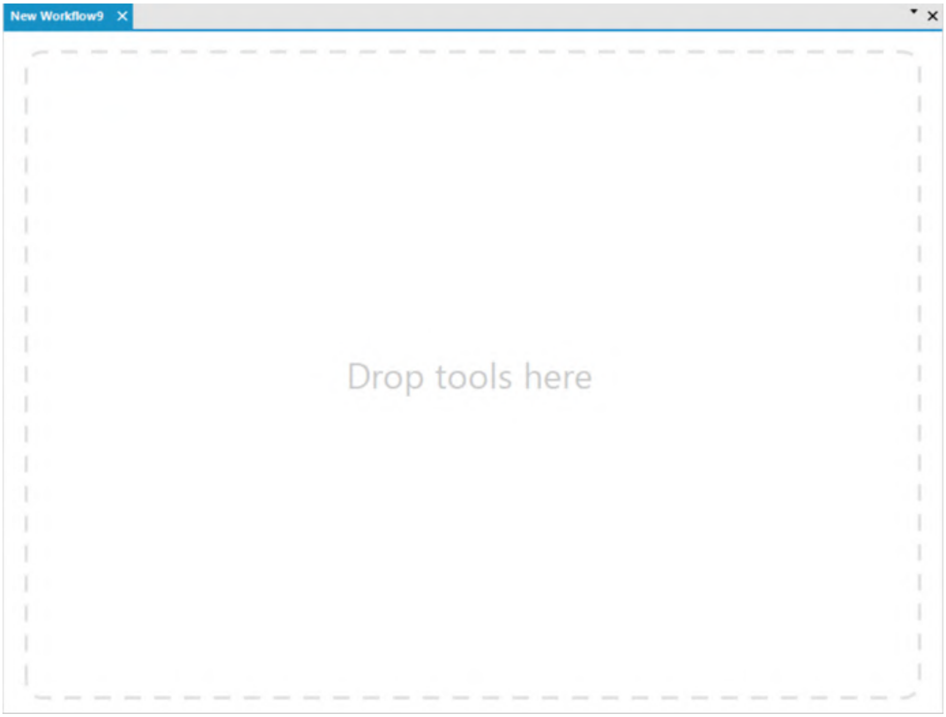


Figure 1-44 – Default Canvas

We can see three elements: the tab name, a drop-down list of all the tabs, and a white area that reads "Drop tools here."

Let's see what happens if we follow the following steps.

1. Open three new windows.
2. Click on the third tab.
3. Click on Window > New Horizontal Tab Group.
4. Click and drag the fourth tab (now third on the top) from the top group to the bottom group.
5. Click on Window > New Vertical Tab Group.

We should see that our canvas has split into three sections and looks like the image below.

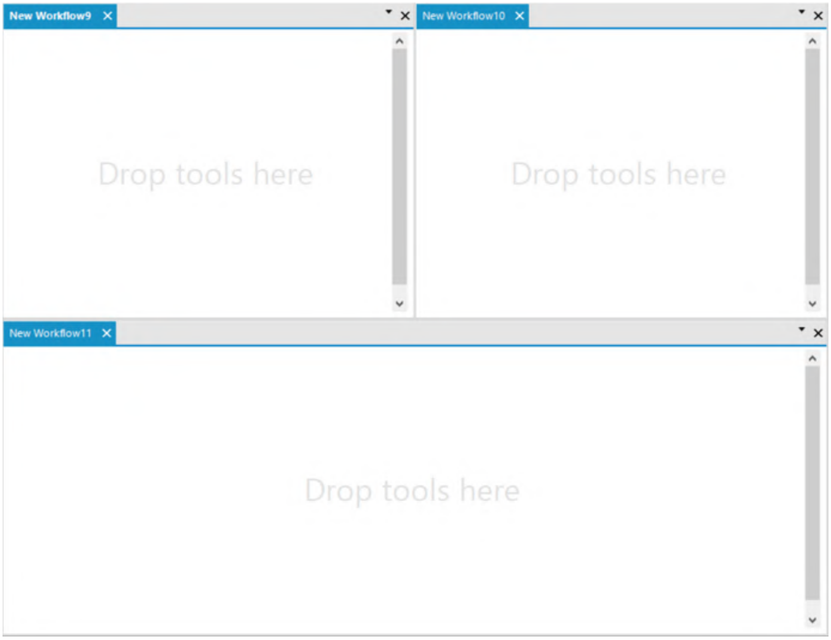


Figure 1-45 – Canvas with Elements

The ability to display workflows next to one another makes it much easier to test parts of a program. We can copy and paste portions between canvases to test, compare potential changes, or just work with multiple workflows at the same time.

Moving the locations of window elements

The overview, output, interface designer, and properties windows allow us to move them around the screen by clicking and dragging the rough part of the title bar (or). When we do this, we can snap them into the window by dropping them on one of the arrows or let it float in front of the canvas or other windows by letting go of it while not over an arrow.

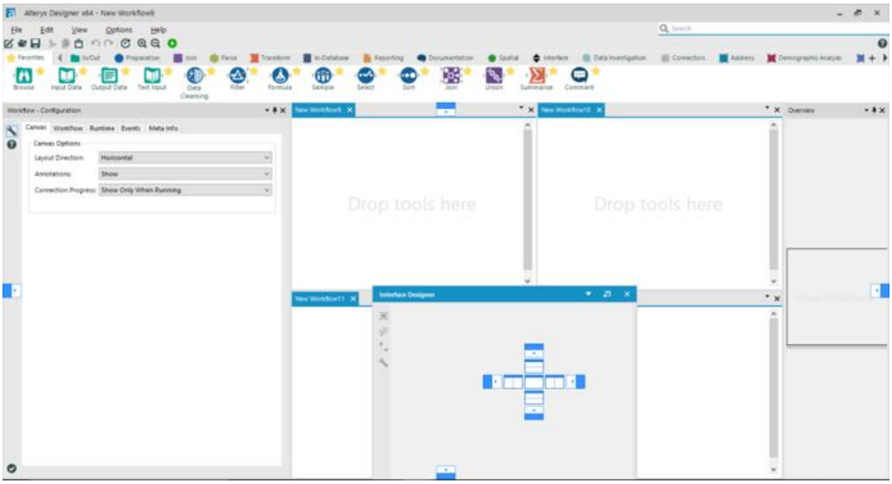


Figure 1-46 – Movement of Canvas Elements

1.19 Using Tools

Data streams start with some sort of data source, and in most cases, this will be the input tool. From there, we may do a series of operations before bringing the data back together to look at.

We will cover each of these methods in detail in the next chapter, but for now, we will look at the structure of a data stream.

This Alteryx Workflow takes input data from a file called Test.txt, splits the data stream (doubling the data), adds a new field to one side, and brings the two streams together to be viewed.

Notice that there are two types of arrows on the incoming (left) side of the tools. A green arrow indicates that a single input can be connected there, while a gray double arrow indicates that multiple connections can be made. This is important because it allows us to know how to connect items to each of these tools.

We can see that there is no parallel tool on the outgoing (right) side of the tools, despite the data stream having multiple outgoing connections. This is because every tool allows us to branch the data stream off into many directions. We also see the lightning bolt and question anchors, which we will discuss when we first use them in *Applications Wanted*.

There are two ways to bring tools onto the canvas: We can both drag the tool from the Tool Palette or right-click on the canvas and navigate the menu on the next page.

1.20 Insert Tool Menu

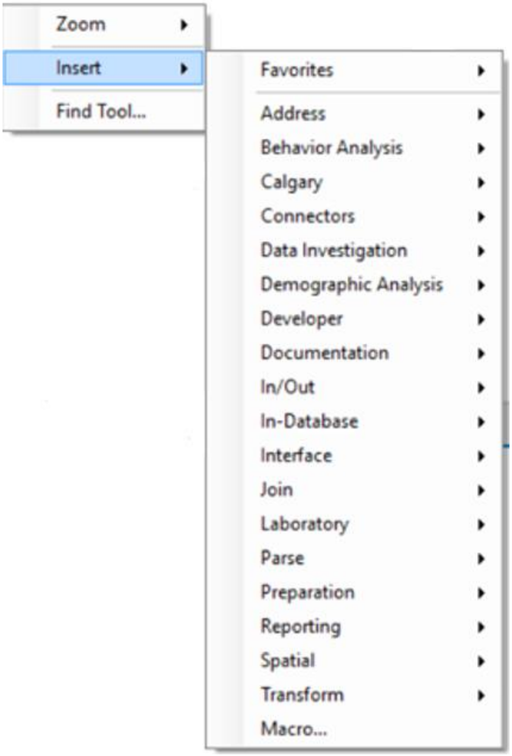


Figure 1-47 – Insert Tool Menu

As an alternative to using the tool palette, you can right-click on the canvas to add tools using the Insert Tool Menu. Simply navigate this menu to find the tool you are looking for.

CHAPTER 2

The Games

We all love playing make believe, don't we? As we discussed in the last chapter, assume that you, dear reader, are the new consultant at a company that works with the Alteryx Analytical Platform.

This book is your mentor and here is our first problem to solve!

To...

Cc...

Subject: Welcome - Let's Get Started

Hey,

Welcome Aboard!

We try to get all of our new hires a basic understanding of Alteryx as quickly as possible.

We will center the basic training around the most important sporting event in the world, which should need no introduction.

I will be asking you a few questions and walking you through examples until I feel like you are ready to handle it yourself.

The first question we are going to explore: Which country has produced the best Freestyle Skiing results overall in the 2002 and 2006 Winter Games?

Assume each Gold is worth 3 points, Silver is worth 1.5 points and Bronze is worth 1 point.

Something important to recognize is that I am asking you for the answer to a very specific question. Once you have some of the basics down, we will talk about making a generalized tool for you or your end user to ask related questions. For now, just understand that when you are asked about a specific answer, they are going to only want the result.

I'll show you how this works.

Thanks,

2.1 Tools and Concepts

2.1.1 Input Data

Concept- Input Data

<table>
<tr><td rowspan="2">
Figure 2-1-Input Data</td><td>**Tool Palette: In/Out**</td></tr>
<tr><td>Imports data from different data sources.
For more details, use the link below.
bit.ly/2JNVIIl</td></tr>
</table>

2.1.2 Browse

Concept- Viewing Data

<table>
<tr><td rowspan="2">
Figure-2-2-Browse</td><td>**Tool Palette: In/Out**</td></tr>
<tr><td>Gives a tabular view of the data in a data stream at the point it is connected.
For more details, use the link below.
bit.ly/2JQTszZ</td></tr>
</table>

2.1.3 Output Data

Concept – Outputting Data

	Tool Palette: In/Out
 Figure 2-3-Output Data	Writes the data stream out to a file or database. For more details, use the link below. bit.ly/2JRM9Id

2.1.4 Comment

Concept – Adding Notes

	Tool Palette: Documentation
 Figure-2-4-Comment	Gives us the ability to write notes on our workflows to add additional information on the data stream. For more details, use the link below. bit.ly/2HpssZY

2.1.5 Filter

Concept – Splitting Data

<table>
<tr><td rowspan="2">
Figure-2-5-Filter</td><td>**Tool Palette: Preparation**</td></tr>
<tr><td>Gives the ability to create a function that will split the data row by row into either the true or false outputs.
For more details, use the link below.
bit.ly/2JRUqMk</td></tr>
</table>

2.1.6 Formula

Concept – Creating Calculations

<table>
<tr><td rowspan="2">
Figure-2-6-Formula</td><td>**Tool Palette: Preparation**</td></tr>
<tr><td>Gives the ability to create a function that will update an existing column or will be written to a new column in our data.
For more details, use the link below.
bit.ly/2H7pdXF</td></tr>
</table>

2.1.7 Join

Concept – Combining Data

 **Figure-2-7-Join**	**Tool Palette: Join**
	Combines two data streams by lining up records based on matching fields. For more details, use the link below. bit.ly/2H3utLT

2.1.8 Running Total

Concept – Cumulative sum

 **Figure-2-8-Running Total**	**Tool Palette: Transform**
	Create a running sum for a numeric field in the incoming data stream. For more details, use the link below. bit.ly/2qIsCkW

2.1.9 Sample

Concept – Creating Data Subsets

 Figure-2-9-Sample	**Tool Palette: Preparation** Allows us to create and work with a subset of data. For more details, use the link below. bit.ly/2HuPChn

2.1.10 Select

Concept – Identifying Desired Results

 Figure-2-10-Select	**Tool Palette: Preparation** Helps to include and exclude fields in the data stream and also modify the metadata associated with the data stream, including the order of fields. For more details, use the link below. bit.ly/2Hal01h

2.1.11 Sort

Concept – Organizing Data

	Tool Palette: Preparation
 **Figure-2-11-Sort**	Arranges the records of a table in alphabetical or numerical order of the specified fields. For more details, use the link below. bit.ly/2JUh1HZ

2.1.12 Summarize

Concept – Summarizing Data

	Tool Palette: Transform
 **Figure-2-12-Summarize**	Helps to perform group operations on the data. For more details, use the link below. bit.ly/2qDplUr

2.1.13 Tool Container

Concept – Grouping Tools

<table>
<tr><td rowspan="2">
Figure-2-13-Tool Container</td><td>Tool Palette: Documentation</td></tr>
<tr><td>Groups tools together for clarity and allows the tools to be disabled when not required. For more details, use the link below.
bit.ly/2vmRlAO</td></tr>
</table>

2.1.14 Transpose

Concept – Denormalizing Data

<table>
<tr><td rowspan="2">
Figure-2-14-Transpose</td><td>Tool Palette: Transform</td></tr>
<tr><td>Helps to denormalize the data. For more details, use the link below.
bit.ly/2H7Y86S</td></tr>
</table>

2.1.15 Union

Concept – Appending records

 Figure-2-15-Union	**Tool Palette: Join**
	Appends records together one after another from multiple data sources. For more details, use the link below. bit.ly/2HGI8FX

2.1.16 Cross Tab

Concept – Normalizing Data Sets

 Figure-2-16-Cross Tab	**Tool Palette: Transform**
	Creates a normalized (more human-readable) dataset by creating columns out of the rows of data. For more details, use the link below. bit.ly/2vjlUHD

Improved Features

Alteryx has changed the naming standards for the platform to reflect when the release takes place (thus Alteryx 2018.1). This will help the users and organizations to get a better understanding of which version they are working on based on the time of year.

Following are some of the new features of Alteryx 2018.1:

Collaborative Insights

- **Alteryx Connect Loaders** - New directly access metadata loaders for data stored in SAP HANA, DB2 and Hadoop Distributed File System (HDFS).
- **Evaluate and display Analytic Assets** – Users can now quickly get a glance at the most popular assets to ensure they are using the right assets. The collaborative environment in Alteryx Connect lets users make assets more searchable.
- **Discover and Access** – Enhanced integration between Alteryx Connect and Alteryx Designer where the Alteryx Designer users can simply search for data using the global search bar in the upper right-hand corner or Designer. Once they find the data set from Connect, it is as simple as dragging it onto the canvas and entering your Alteryx Connect credentials, which ensures that you are an approved user of the data. This is data collaboration and data governance all in one platform.
- **Expanded Data Lineage** - Establish Alteryx Connect lineage from Designer workflows that use In-Database processes/tools

Analytic Flexibility

- **Expanded data Connections** – New connectors for Amazon Web Services (AWS) Athena, Redshift Spectrum and enhanced integration with Excel.
- **Tableau Support** – Alteryx already supported Tableau output by outputting directly to. tde (Tableau Data Extract) and

integrations with Tableau Server. Now output directly from an Alteryx workflow into Tableau Hyper is also supported.

- **Harness the power of Spark** - New Code Tool for Apache Spark within the In-Database tools in Alteryx Designer to execute code from R, Python or Scala directly against the Spark cluster, leveraging the power of Spark but removing the complexity through easy to use drag-and-drop environment.
- **Enhanced Formula and Filter tool** – Enhanced formula tool to write fast and accurate expressions. Added the same functionality to quickly use automated suggestions, error notifications and auto- completion of custom expressions to speed up time to insights and data results.

Operationalize Models

- **Deploy Predictive Models** - Alteryx Promote makes it easy for data scientists and data analysts alike to deploy, manage and monitor production-based models and real-time applications through a REST API. With Alteryx Promote users can take predictive/machine learning models that were built in Alteryx, R, or Python and quickly deploy them to the Alteryx Promote server.
- **Manage Models from development to production** - Once the model is in the Promote management system it can be tested and staged before it is put into production to ensure it will deliver the best impact on the business.
- **Monitor Model Performance and Health** - Organizations can then monitor the effectiveness and health of the models to understand whether or not the model needs to be retrained or removed.

2.2 Freestyle

To...

Cc...

Subject: Welcome - Let's Get Started

Hey,

Welcome Aboard!

We try to get all of our new hires a basic understanding of Alteryx as quickly as possible.

We will center the basic training around the most important sporting event in the world, which should need no introduction.

I will be asking you a few questions and walking you through examples until I feel like you are ready to handle it yourself.

The first question we are going to explore: Which country has produced the best Freestyle Skiing results overall in the 2002 and 2006 Winter Games?

Assume each Gold is worth 3 points, Silver is worth 1.5 points and Bronze is worth 1 point.

Something important to recognize is that I am asking you for the answer to a very specific question. Once you have some of the basics down, we will talk about making a generalized tool for you or your end user to ask related questions. For now, just understand that when you are asked about a specific answer, they are going to only want the result.

I'll show you how this works.

Thanks,

Let's start building a workflow that will answer our question. We are going to start with a blank canvas and save it as Freestyle Skiing. Next, bring an Input Data tool so that we can connect to data.

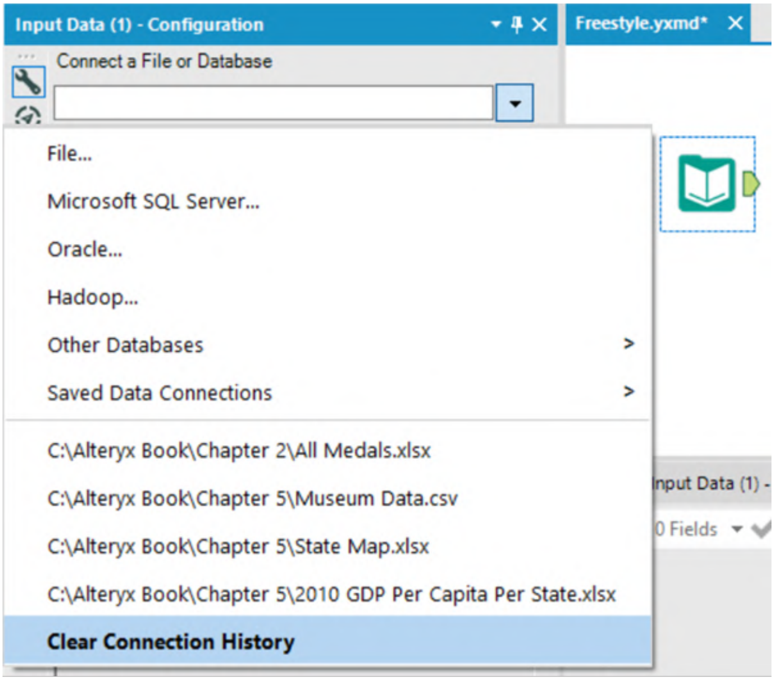


Figure-2-17-Freestyle Skiing Input Data Configuration

Now navigate to where data files are unpackaged and connect to the file in *Chapter 2 – All Medals.xlsx*. For downloading the data associated with this book, please refer to the letter to the reader on this chapter's first page.

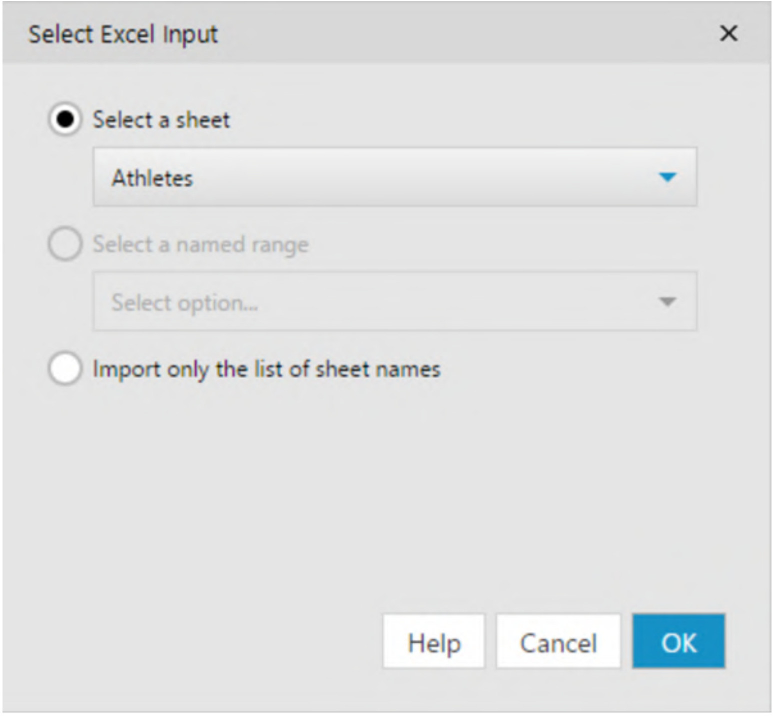


Figure-2-18-Freestyle Skiing – Data Input

When connected, we see this window pop up. Select *Athletes* and then click OK to connect to the *Athletes* sheet in the *All Medals* Excel file. This is shown in the figure above.

Best practices are to put a select and a browse after every input.

- Browse helps us check the data at the time of import. This ensures that the data we are getting is correct.
- Select allows us to make sure that the fields are in the right format from the beginning.

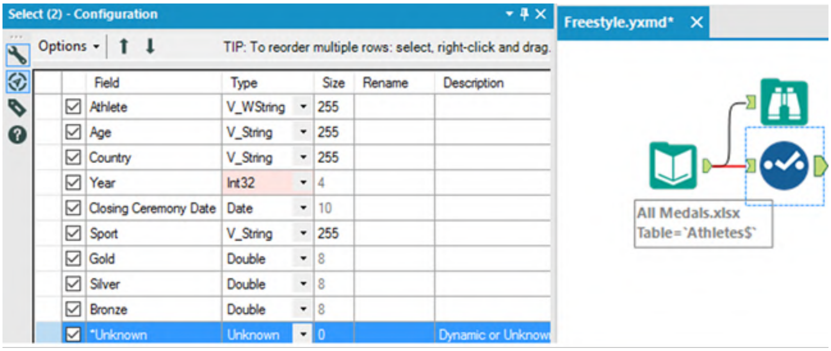


Figure-2-19-Freestyle Skiing
Select Configuration

If we click on Select, we should see that our fields are of different types than the above image. Change them to match what is shown.

Now that we have the data and the fields are the right type, the first thing we should do is filter the data. We always want to *limit the data as soon as possible*, as this will speed up our data stream and prevent memory errors by limiting the information.

Best practice is to remove data as soon as it is no longer needed.

It makes sense that the first step in filtering would be to bring in the Filter tool; however, if we are not familiar with the data set and we have not run it, we may not have enough information to filter properly. In this case, we want to run the module so that there's data in the Browse tool for us to work with.

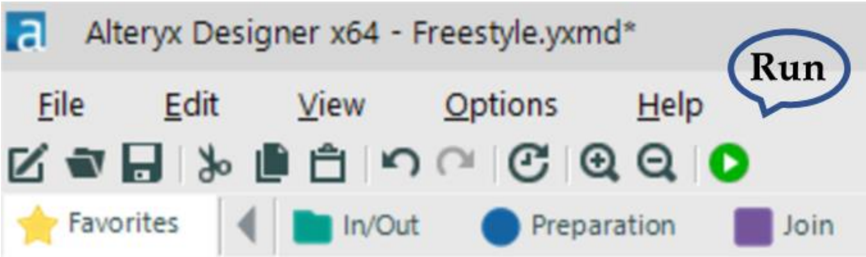


Figure-2-20 – Freestyle Skiing
Click Run to populate Browse

We can see that when the module finishes running, we get a pop-up window that lets us know how long it took to run and if there were any errors.

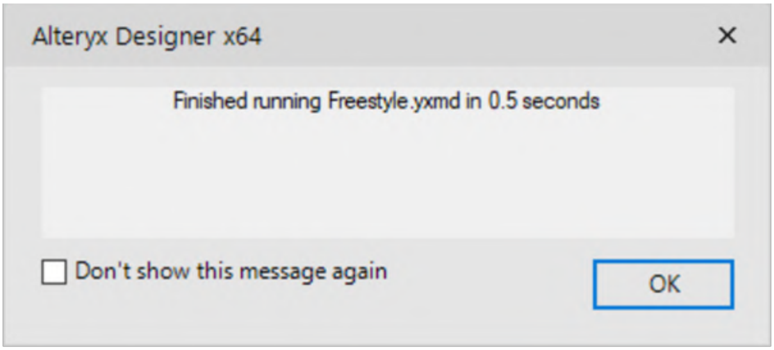


Figure-2-21-Freestyle Skiing – Message after Running Workflow

Feel free to click on the *Don't show this message again* check box before closing if the pop-up window is distracting.

Now, we can start thinking about the filter. We know that we are only interested in freestyle skiing results for the 2002 and 2006 games. So, the first thing we are going to filter is the sport of "freestyle skiing." If we look at the browse tool, we first see freestyle skiing at row 5818, identified by the string *Freestyle Skiing*.

Record #	Athlete	Age	Country	Year	Closing Ceremony Date	Sport	Gold	Silver	Bronze
5815	Irina Slutskaya	23	Russia	2002	2002-02-24	Figure Skating	0	1	0
5816	Aleksey Yagudin	21	Russia	2002	2002-02-24	Figure Skating	1	0	0
5817	Zhao Hongbo	28	China	2002	2002-02-24	Figure Skating	0	0	1
5818	Shannon Bahrke	29	United States	2010	2010-02-28	Freestyle Skiing	0	0	1
5819	Dale Begg-Smith	25	Australia	2010	2010-02-28	Freestyle Skiing	0	1	0
5820	Hedda Berntsen	33	Norway	2010	2010-02-28	Freestyle Skiing	0	1	0
5821	Alexandre Bilode...	22	Canada	2010	2010-02-28	Freestyle Skiing	1	0	0
5822	Aleksey Grishin	30	Belarus	2010	2010-02-28	Freestyle Skiing	1	0	0
5823	Audun Grønvold	33	Norway	2010	2010-02-28	Freestyle Skiing	0	0	1

Figure-2-22-Freestyle Skiing – Browse

This is a crucial piece of information we didn't have before. Now that we know exactly what we need to look for in our data, we can create the filter.

Drag a Filter tool after the Select, and make sure there is a connection between the Select output and the Filter input. This time, we will use the Basic Filter builder. Set the field drop-down to *Sport*, and type *Freestyle Skiing* into the text box like we see below.

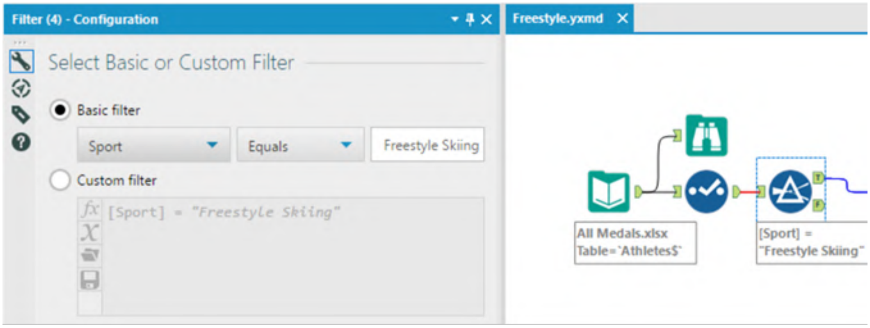


Figure-2-23-Freestyle Skiing – First Filter Configuration

Notice that the Expression says [Sport] = "Freestyle Skiing." This is because field names are in square brackets and string values are in quotes. What is happening here is that for each record, we test to see if the value in Sport in exactly *Freestyle Skiing*. If it is, then True; if it isn't, then False.

The next thing we want to do is create a filter to keep the records with years 2002 and 2006. If we look at the Select tool on the previous page, we will see that the *Year* field is a string. This is fine; we just need to remember this while we are writing the filter formula. Drag a new Filter tool onto the canvas and make sure that the first Filter's true (*T*) output is connected to the new Filter's input.

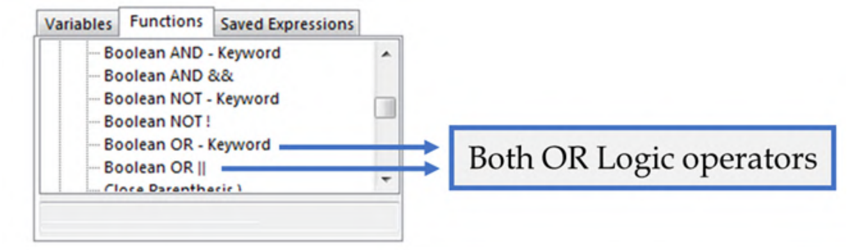


Figure-2-24-Freestyle Skiing Filter Configuration - Functions

This time, we are going to create the filter logic on our own using the Custom Filter option. From the previous filter, we know that **[<Field Name>] = "<Value>"** is the syntax for filtering a string field, thus creating the first half of this filter is not that difficult – it is **[Year] = "2002"** – but what we need to do now is make sure 2006 is also retained.

One way that we could do this is by using logical operators. *Logical operators* are terms that allow us to combine two or more Boolean (true or false) values to create a single Boolean output from the two. The three Boolean operators that we will be discussing are *AND*, *OR* and *NOT*.

AND: if **both** the value to the **left and** the value on the **right are true, then true**.

OR: if **either** the value to the **left or** the values to the **right or both** are **true, then true**.

NOT: **if** a value is **true** then **false**; **if** the value is **false,** then **true**.

We have the option to use the *Boolean OR - Keyword* or the *Boolean OR* | |. There is no computational difference between using the keyword or the double vertical bar symbol. Both options are available for our convenience. For those who are not used to programming, the keyword *OR* is much easier to remember and use, but for those who program, double vertical bars (| |) is a common standard they may be used to.

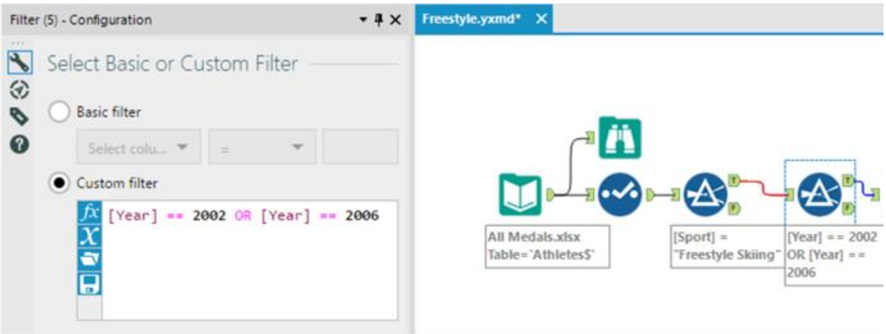


Figure-2-25-Freestyle Skiing Custom Filter Configuration

See Appendix F for examples of Boolean logic.

Now that we know about logical operators, we can finally finish configuring the filter. We can use the formula **[Year] = "2002" OR [Year] = "2006"** in order to filter out this data.

Note: We could have combined both of these filters together by using the following: **[Sport] = "Freestyle Skiing" AND ([Year] = "2002" OR [Year] = "2006")**

The parentheses allow us to change the order of operations so that this formula reads "Freestyle Skiing in the years 2002 or 2006" instead of "Freestyle Skiing in 2002 or anything in 2006."

We have now removed the data which we do not require in order to solve this question. But still the problem remains - that the data is too granular. We know who the athlete was and in which year they won their medal(s). We should bring in a Summarize Tool in order to aggregate the data up to the country level. If we add Country using Group By, and Gold, Silver, and Bronze using Sum, we will get a list of countries and their total medal count for Freestyle skiing for 2002 and 2006. Place a Browse Tool and run the module to see what we have so far.

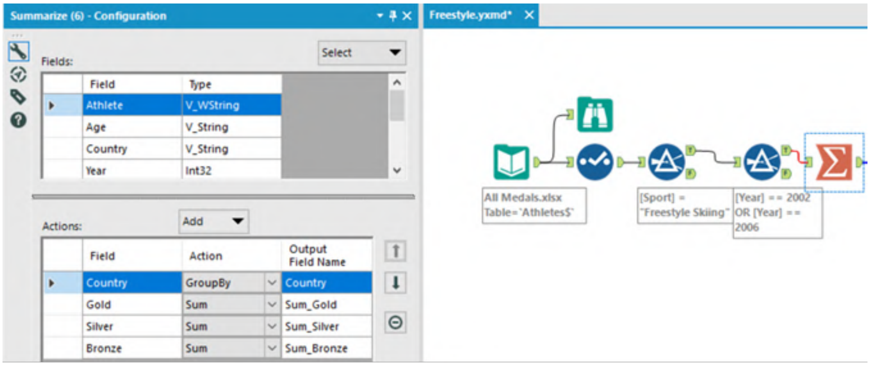
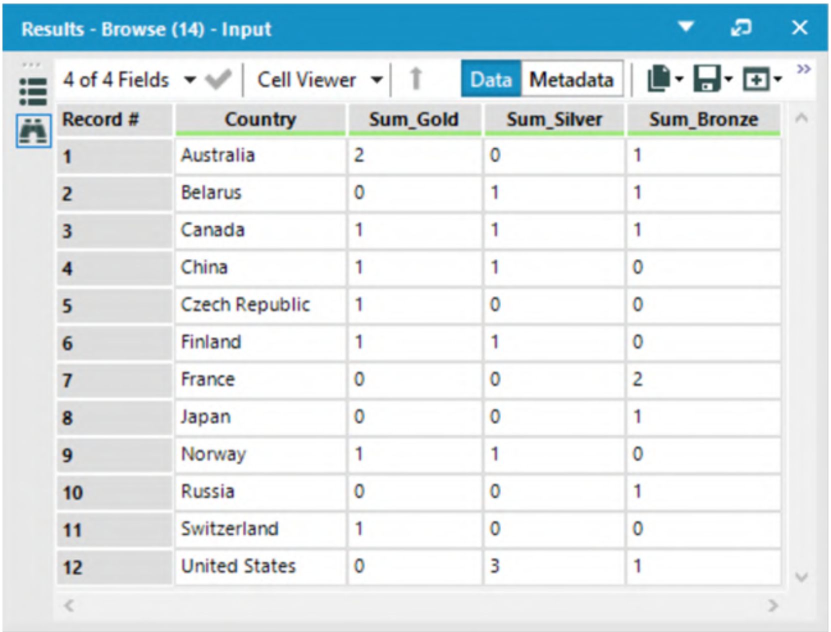


Figure-2-26-Freestyle Skiing Summarize Configuration

Best practice is to place a browse tool after every tool that transforms the data into a significantly different format. Summarize is one of those tools.



Record #	Country	Sum_Gold	Sum_Silver	Sum_Bronze
1	Australia	2	0	1
2	Belarus	0	1	1
3	Canada	1	1	1
4	China	1	1	0
5	Czech Republic	1	0	0
6	Finland	1	1	0
7	France	0	0	2
8	Japan	0	0	1
9	Norway	1	1	0
10	Russia	0	0	1
11	Switzerland	1	0	0
12	United States	0	3	1

Figure-2-27-Freestyle Skiing Browse after Summarize

We can see that there are four columns with the total counts of gold, silver and bronze medals listed for each of the 12 countries that won freestyle skiing medals during 2002 and 2006. Notice the fields are titled *Sum_* followed by the original field name. Alteryx is making sure we know the method used to summarize the data.

The next thing we need to do is determine which country performed the *best*. If we look back at the email, we can see that *best* is defined as a function of the medals won; 3 points for gold, 1.5 points for silver and 1 point for bronze.

Bring a Formula Tool onto the canvas following the Summarize tool and we are going to create a calculation called Score that has the Type Double, with the formula [Score] = 3*[Sum_Gold] + 1.5*[Sum-Silver] + [Sum_Bronze]

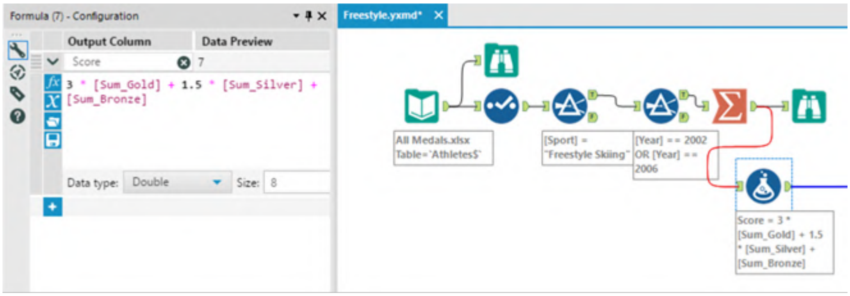
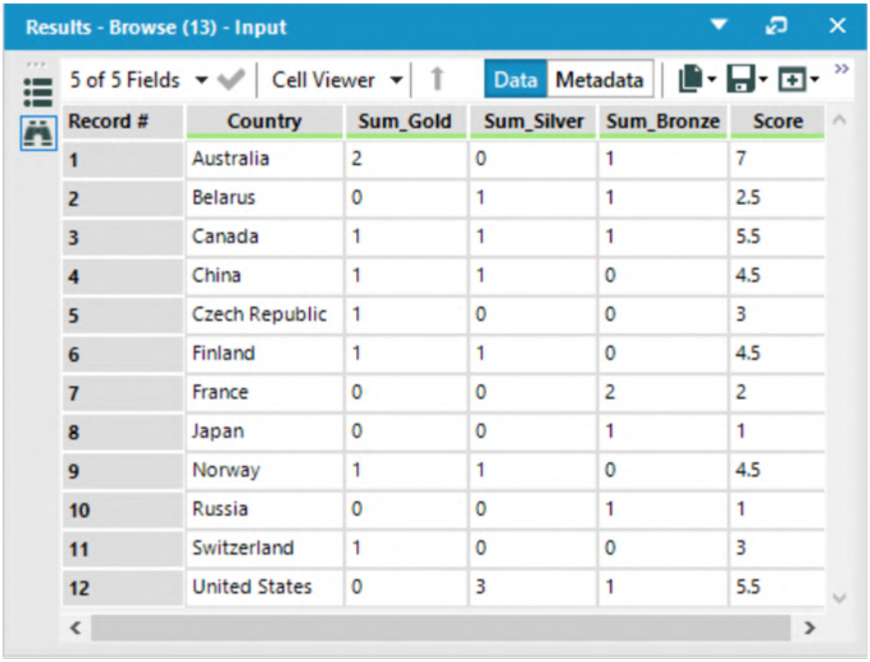


Figure-2-28-Freestyle Skiing - Formula Configuration

We can now add another Browse after the Formula Tool to see what the data looks like.



Record #	Country	Sum_Gold	Sum_Silver	Sum_Bronze	Score
1	Australia	2	0	1	7
2	Belarus	0	1	1	2.5
3	Canada	1	1	1	5.5
4	China	1	1	0	4.5
5	Czech Republic	1	0	0	3
6	Finland	1	1	0	4.5
7	France	0	0	2	2
8	Japan	0	0	1	1
9	Norway	1	1	0	4.5
10	Russia	0	0	1	1
11	Switzerland	1	0	0	3
12	United States	0	3	1	5.5

Figure-2-29-Freestyle Skiing Browse after Formula tool

We see there is a new field called Score that is an unordered data set and with multiple unnecessary values. We can also see that Australia has the highest score and therefore is the answer to the original question. But for good practice, we are going to continue to build this workflow so that no interpretation is needed.

This process is going to take four steps:

1. Reorder the data based on the score field.
2. Select only the top-scoring country.
3. Remove all data other than the name of the best country.
4. Browse that data.

Like we discussed, we will first reorder the data using Sort Tool. We will set up our data in a descending order based on Score as shown below.

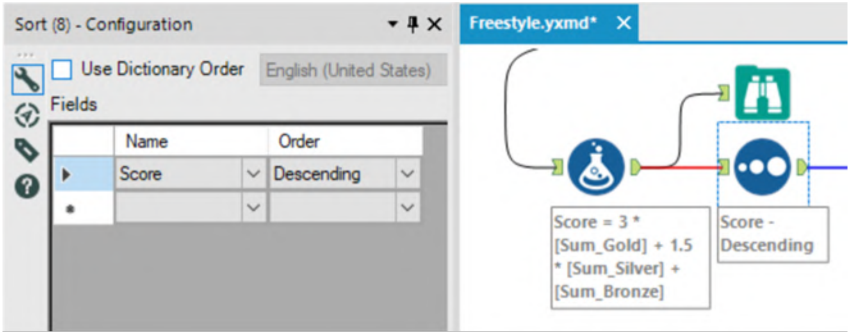


Figure-2-30-Freestyle Skiing Sort Configuration

Next, we just need the first record, so we are going to use the Sample Tool such that we only get Top 1 Record as output from the Sort.

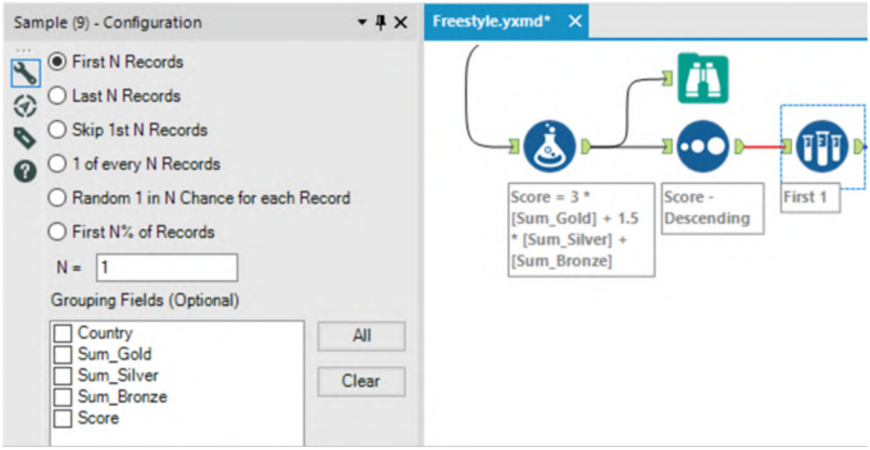


Figure-2-31-Freestyle Skiing Sample Configuration

As we will no longer need rest of the fields, we can use a Select Tool to remove everything that is not country name.

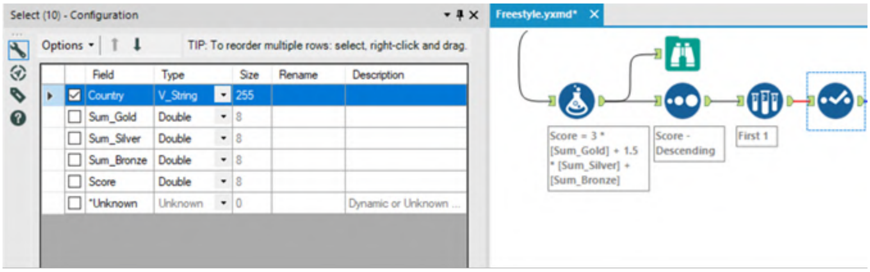


Figure-2-32-Freestyle Skiing Select Configuration

Finally, we can put a browse tool at the end and run the workflow to see the results.

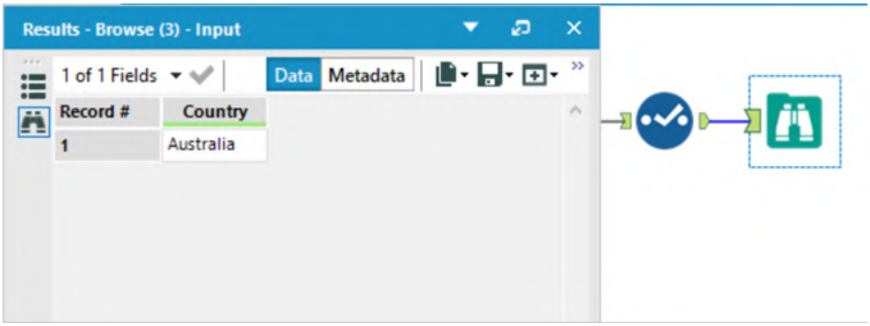


Figure-2-33-Freestyle Skiing Browse after Select

We could have stopped when we first saw that Australia had the highest score in the previous browse tool. The reason we did not, is that when we are performing an analysis, we want our results to be perfectly repeatable. If we had interpreted the previous browse tool incorrectly then there would have been no way of finding out why the error occurred. This is a problem because it makes the individual analyst entirely responsible for the answer and anyone who checked the results could easily find the correct answer where we mistakenly picked the wrong one. Creating the workflow in this way affords us two benefits:

- We would have a second verification that the answer was what we expected.
- Repeatability of the result so we can point to a single issue in the data preparation process that needs fixing instead of not being able to fix it at all.

Here is how the workflow would look like when complete:

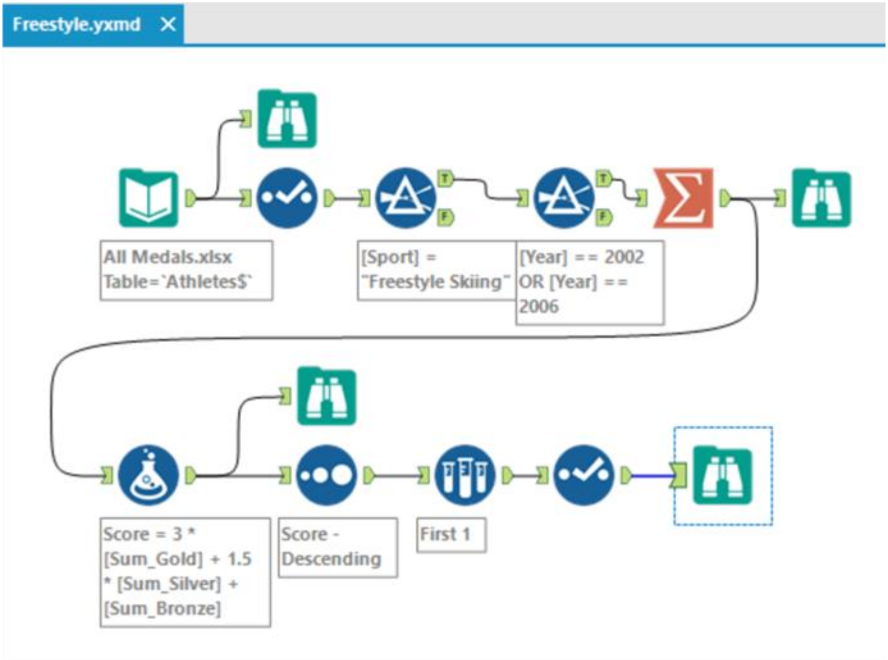


Figure-2-34-Freestyle Skiing Workflow after completion

2.3 Let's Tidy Things Up

From

To... Alteryx Consultants

Cc...

Subject Let's Tidy Things Up

Hey,

That was great!

So the next thing we are going to cover is a question that requires you to produce a dataset instead of just an answer.

The goal of most data manipulation is to get the data in a more useable format. Typically, there are two formats that are most appropriate. Which you create is going to depend on what you are trying to do. The first and more relatable is to have a wide *normalized* data structure, which you can think of as "human friendly". These datasets tend to have multiple columns that have the same metrics in them but are replicated because you have a variable you want to compare across. The other has two common names that we will be using interchangeably; "tidy" is one and "denormalized" is the other. These data sets are categorized by having a single field for each variable and are often very "tall" (long). You can think of it as "computer friendly".

Let's consider the data source on the historical medal counts that we were just working on. In this case, we have a mostly denormalized dataset. Let's take that last step in creating a truly tidy dataset by converting the three columns gold, silver, and bronze into "Medal Type" and "Medal Count" so that downstream systems can process the data better.

Thanks,

This process must include at least four steps:

1) Import the dataset
2) Transpose the dataset
3) Make sure the fields are named correctly
4) Export the dataset

However, we are going to make the data cleaner and also employ best practices. So, our process is:

1) Import the data
2) Browse the data
3) Make sure the data has the right data type
4) Transpose the data
5) Browse the restructured data
6) Make sure the fields are named correctly
7) Remove records that say there were no medals won
8) Browse the data that will be exported
9) Export the dataset

Let's create a new workflow and save it as *Let's Tidy Things Up*.

We need to import the same data that we used in the last example. Bring an input tool onto the canvas, navigate to where we saved this book's data, and connect to the file in *Chapter 2 – All Medals.xlsx*

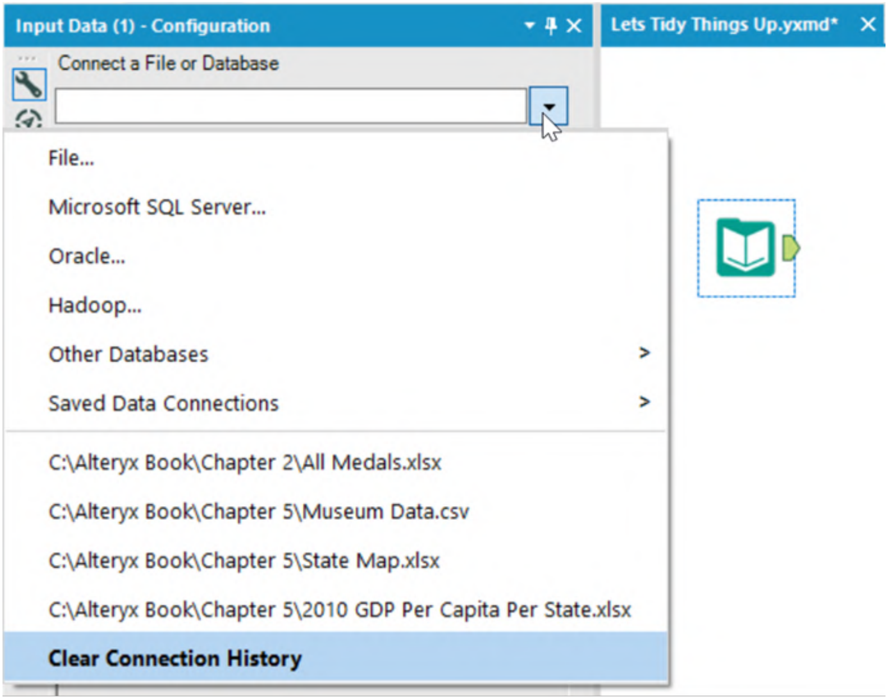


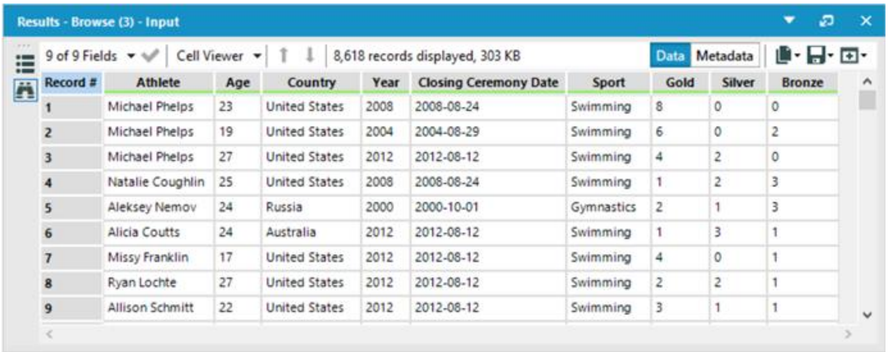
Figure-2-35-Let's Tidy Things Up – Data Input

Now we will put a Browse and select statement following the Input Tool.

Figure-2-36-Let's Tidy Things Up – Medals Data

All the fields have appropriate data types, so we can move directly to the transposition.

Let's run the Module to see how the data is structured by clicking on the Browse tool.



Record #	Athlete	Age	Country	Year	Closing Ceremony Date	Sport	Gold	Silver	Bronze
1	Michael Phelps	23	United States	2008	2008-08-24	Swimming	8	0	0
2	Michael Phelps	19	United States	2004	2004-08-29	Swimming	6	0	2
3	Michael Phelps	27	United States	2012	2012-08-12	Swimming	4	2	0
4	Natalie Coughlin	25	United States	2008	2008-08-24	Swimming	1	2	3
5	Aleksey Nemov	24	Russia	2000	2000-10-01	Gymnastics	2	1	3
6	Alicia Coutts	24	Australia	2012	2012-08-12	Swimming	1	3	1
7	Missy Franklin	17	United States	2012	2012-08-12	Swimming	4	0	1
8	Ryan Lochte	27	United States	2012	2012-08-12	Swimming	2	2	1
9	Allison Schmitt	22	United States	2012	2012-08-12	Swimming	3	1	1

Figure-2-37-Let's Tidy Things Up – Data in Browse Tool

The Transpose tool takes normalized data and de-normalizes it. If we take the data stream coming out of *Select* and pass it into a Transpose, we can make the data tidier.

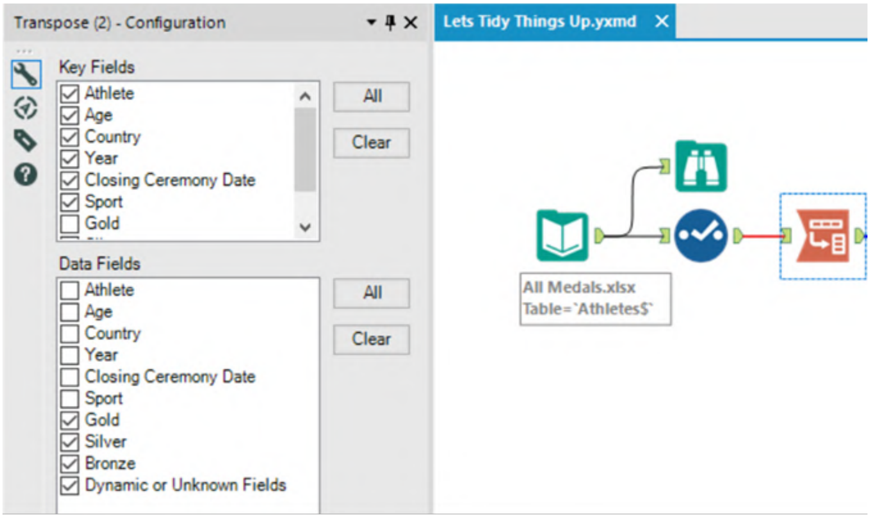
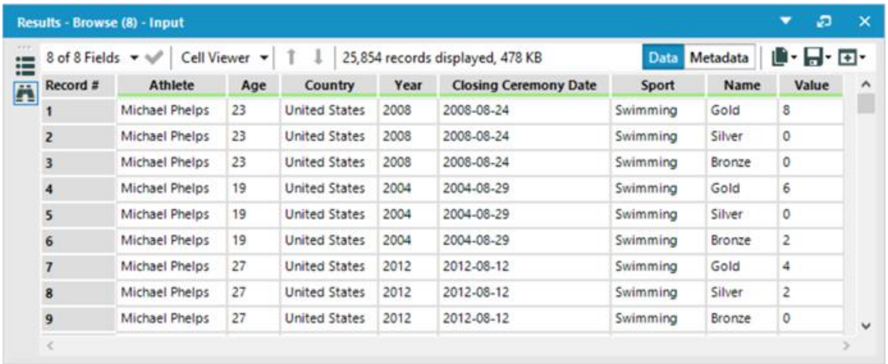


Figure-2-38-Let's Tidy Things Up – Transpose Configuration

We want to keep all the fields as they are, except for gold, silver, and bronze. So, we select all but those three fields under Key Fields and then we select gold, silver, and bronze under the Data Fields. If we want to drop a field entirely – say, Closing Ceremony Date – we can leave it unchecked in both lists.

Best practice is to always include a Browse after a tool that modifies the structure of a data stream. Transpose is one of these tools. Let's add a Browse to the end of the data stream and run it to see what we have.



Record #	Athlete	Age	Country	Year	Closing Ceremony Date	Sport	Name	Value
1	Michael Phelps	23	United States	2008	2008-08-24	Swimming	Gold	8
2	Michael Phelps	23	United States	2008	2008-08-24	Swimming	Silver	0
3	Michael Phelps	23	United States	2008	2008-08-24	Swimming	Bronze	0
4	Michael Phelps	19	United States	2004	2004-08-29	Swimming	Gold	6
5	Michael Phelps	19	United States	2004	2004-08-29	Swimming	Silver	0
6	Michael Phelps	19	United States	2004	2004-08-29	Swimming	Bronze	2
7	Michael Phelps	27	United States	2012	2012-08-12	Swimming	Gold	4
8	Michael Phelps	27	United States	2012	2012-08-12	Swimming	Silver	2
9	Michael Phelps	27	United States	2012	2012-08-12	Swimming	Bronze	0

Figure-2-39-Let's Tidy Things Up – Data in Browse after Transpose

If we compare the top three records from the new Browse to the one that came out of the Input in figure 2-66, we see that we have two fields - *Name* and *Value* and no longer have the fields - *Gold, Silver,* and *Bronze*. We also notice from *Athlete* to *Sport* that all the fields are identical to the first three records in the original dataset. This is because we replicated them for each column we created.

This is one of the reasons that tidy data is not particularly human readable but is highly computer readable. Since all of the information is displayed in each record and there is only a single column to work on, interactive front-end software can work very fast with the data.

To clean this data entirely would mean that we need to rename *Name* and *Value* to field names that will give better context to those fields. Add a Select statement and rename the Name and Value fields - *Medal Type* and *Medal Count* respectively.

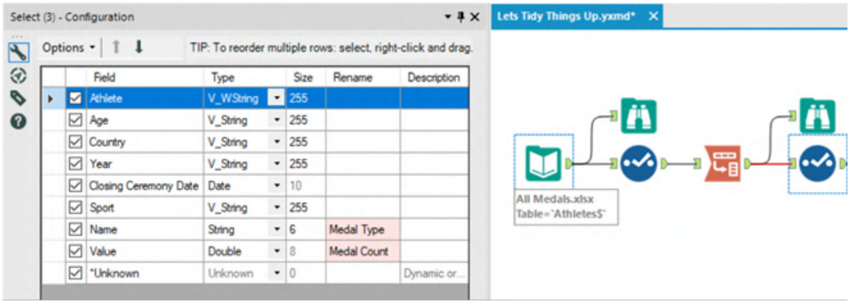


Figure-2-40-Let's Tidy Things Up – Select Configuration

By looking at the values in the last Browse tool we created, we know that there are some records which show zero medals were won. We are going to filter out those data points by adding a Filter tool after the Select.

Our goal is to filter out any records that have zero medals. We are filtering on a numeric field for the first time, which means we should use the Basic Filter to learn about the syntax. The configuration is as shown in the following figure.

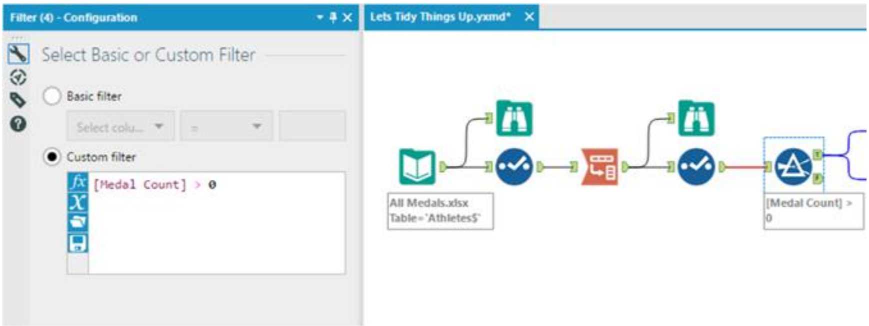


Figure-2-41-Let's Tidy Things Up – Filter Configuration

We can see how if we select Medal count, we have different options in the operator drop-down. This is because numeric fields allow different comparison methods than string fields.

We want to select greater than – ">" – and type "0" in the text box. When we look at the Expression below, we see that it says **[Medal Count] > 0**. This is because we do not put numeric values in quotes.

Alteryx recognized that when we selected a numeric field in the basic filter drop-down, the "0" we typed in meant the number 0 and not the string 0, so it adds the numeric value into the formula.

The last step involves two tools: *Browse* tool and the *Output Data* tool.

Best practice dictates that we put a Browse before every data output so that we do not need to open the file to make sure we have created it correctly.

We now add a Browse and an Output Data tool to the end of the data stream whose output is "True". We are going to write the file to the same folder where we have saved the *Let's Tidy Things Up.yxmd*.

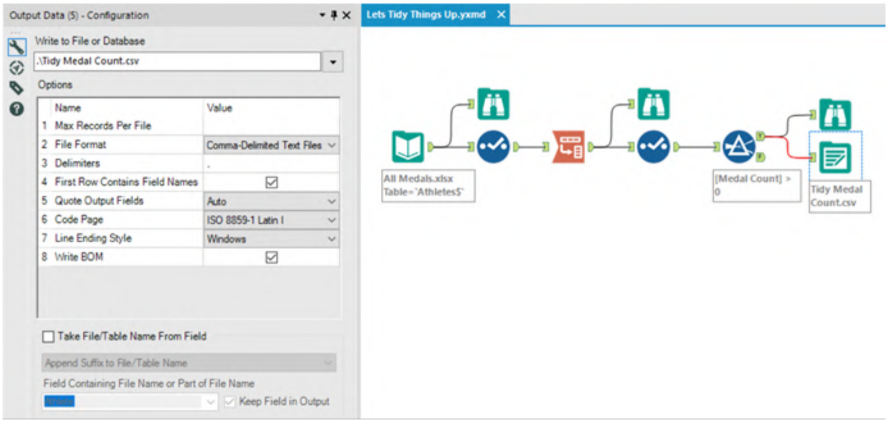


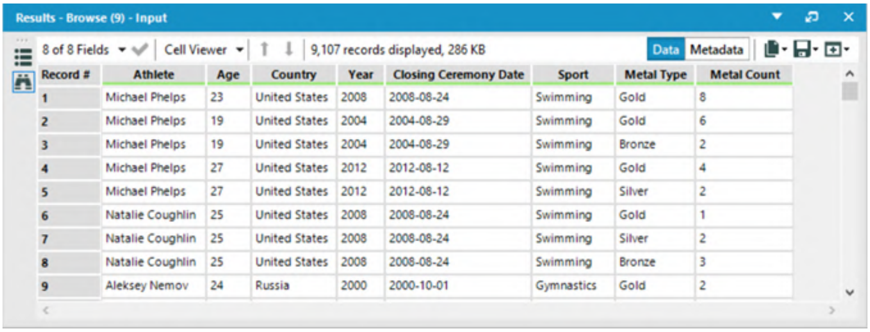
Figure-2-42-Let's Tidy Things Up – Output Data Configuration

To do this, we are going to type.\ *Tidy Medal Data.csv* in the text box labeled *Write to File or Database.*

We just used a relative file path. This allows us to reference to where we currently are. Some basics of relative paths are - ".\" which means the current folder. "..\" means the parent folder (the folder that our current folder is in). ".\Folder Name\" will move our file into a folder below the one where we have our workflow.

We do not necessarily need to use relative paths, but if we are sharing Alteryx files, it is very beneficial to do so. We can use absolute paths (full file locations) by pasting them into this box or navigating through them in the File Browse option.

If we run the module, we will see how the transformed dataset looks like. This ensures that the information written into the .csv was correct.


Results - Browse (9) - Input

8 of 8 Fields | Cell Viewer | 9,107 records displayed, 286 KB | Data | Metadata

Record #	Athlete	Age	Country	Year	Closing Ceremony Date	Sport	Metal Type	Metal Count
1	Michael Phelps	23	United States	2008	2008-08-24	Swimming	Gold	8
2	Michael Phelps	19	United States	2004	2004-08-29	Swimming	Gold	6
3	Michael Phelps	19	United States	2004	2004-08-29	Swimming	Bronze	2
4	Michael Phelps	27	United States	2012	2012-08-12	Swimming	Gold	4
5	Michael Phelps	27	United States	2012	2012-08-12	Swimming	Silver	2
6	Natalie Coughlin	25	United States	2008	2008-08-24	Swimming	Gold	1
7	Natalie Coughlin	25	United States	2008	2008-08-24	Swimming	Silver	2
8	Natalie Coughlin	25	United States	2008	2008-08-24	Swimming	Bronze	3
9	Aleksey Nemov	24	Russia	2000	2000-10-01	Gymnastics	Gold	2

Figure-2-43-Let's Tidy Things Up – Browse after Filter

Here is how the *Let's Tidy Things Up* workflow looks on completion.

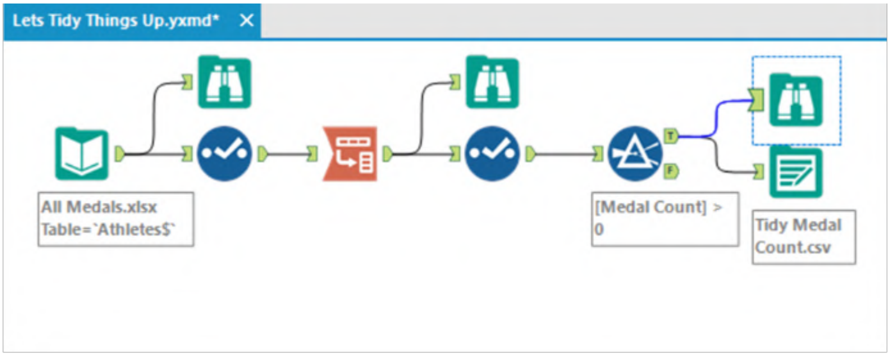
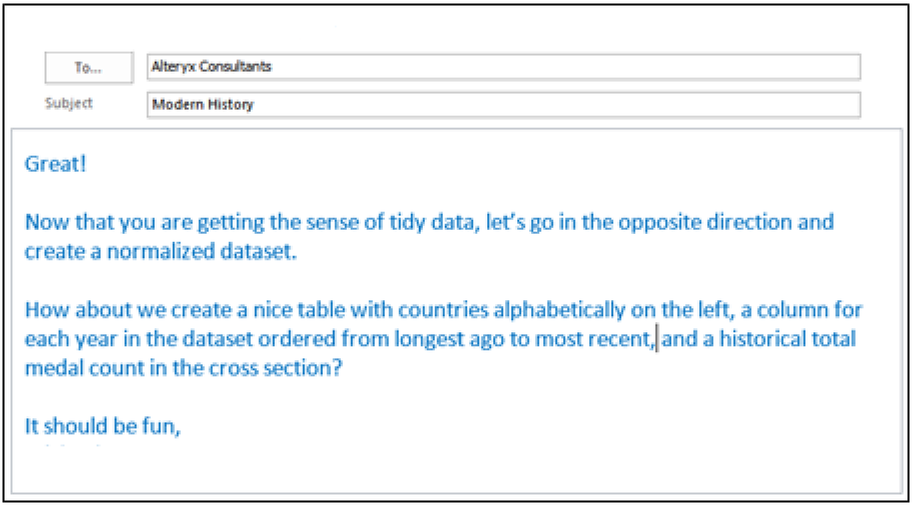


Figure-2-44-Let's Tidy Things Up – Final Workflow after completion

2.4 Modern History



To... Alteryx Consultants

Subject Modern History

Great!

Now that you are getting the sense of tidy data, let's go in the opposite direction and create a normalized dataset.

How about we create a nice table with countries alphabetically on the left, a column for each year in the dataset ordered from longest ago to most recent, and a historical total medal count in the cross section?

It should be fun,

Notice that there is considerably less context built into this email. We often get very sparse information from people and they will assume that we have the context. In this case, it was assumed we are talking about the medal data which we have been working with during the training so far.

This is a much more complicated process than the last exercise, but that is only because the data was set up very well for what we were doing last time, but it isn't here.

We are going to be connecting to the same data source that we have been using, but we are going to use a shortcut in the connection process. Open a new workflow and save it as *Modern History.yxmd*, but make sure that *Let's Tidy Things Up.yxmd* is still open.

Click on the data input in *Let's Tidy Things Up* and copy it. Move over to the Modern History canvas and paste the copied content. We see that the input has been copied over and we do not need to recreate the connection. Connect to the file in *Chapter 2- All Medals.xlsx*.

Best practice will once again bring in Browse and Select tools. But since we know from past experience what the data looks like and how it is read in, we will move directly to the next step.

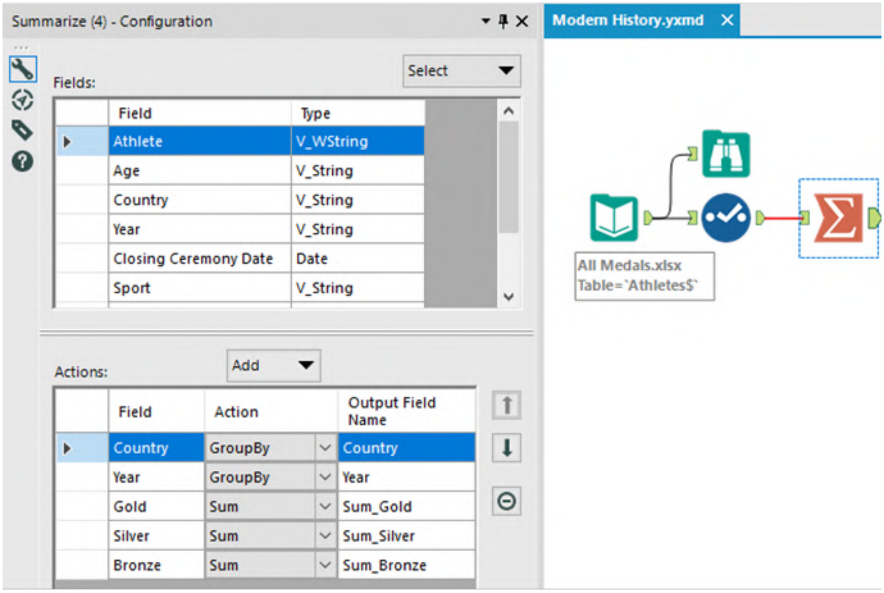


Figure-2-45-Modern History – Summarize Configuration

We know that this data is too granular for our desired result. So, we will summarize it. Based on the email, we know that the only information we will need in the end is the country, year and something to do with the medals.

So, while using the Summary tool, we will group by the country and year fields and take the sum of each of the medal counts to take our first step down this path.

We now add a formula that creates a *Total Medal* count by adding the gold, silver and bronze fields for each record. (Remember that we used a Summarize tool so we should have a Browse tool.)

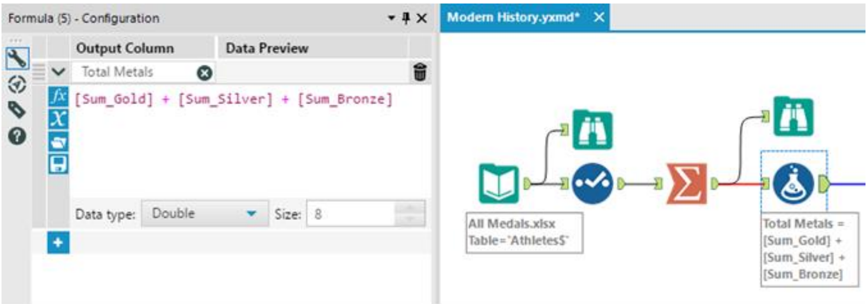


Figure-2-46-Modern History – Formula Configuration

We can now add a Select tool that will allow us to keep only the Country, Year and Total Medals fields, which we will use to create the table.

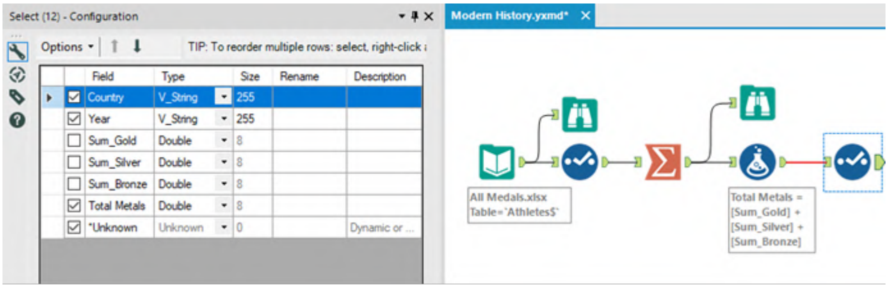


Figure-2-47-Modern History – Select Configuration

To get the historical medal count, we will need to take the running total along with the country and year. But because Running Total is a tool where order matters, we need to sort the data.

We will sort the Country and Year in ascending order to help us in two ways: Initially, this will help in creating the order for the Running Total, but it will also help us with the order of records and columns when we normalize the data set.

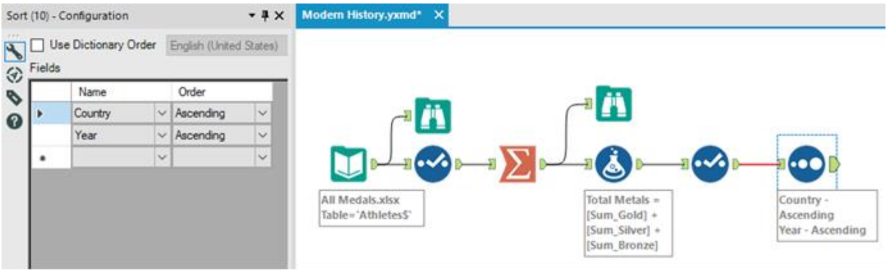


Figure-2-48-Modern History – Sort Configuration

Now that we have the data in a specific order, we can create the Running Total for each country, across years. To do this, we *Group By* Country and *Create Running Total* on Total Medals. This will create the running sum of Total Medals down the data set (as time increases) and have that count restart every time a new country shows up.

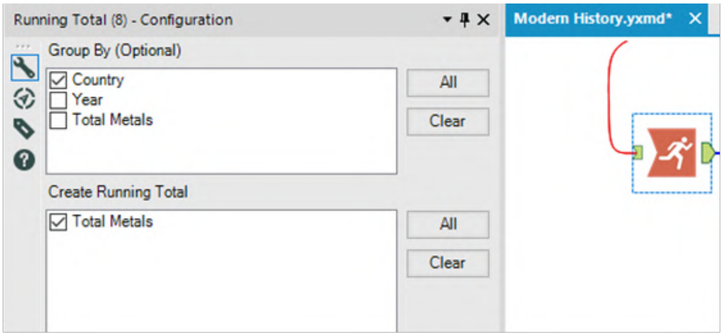
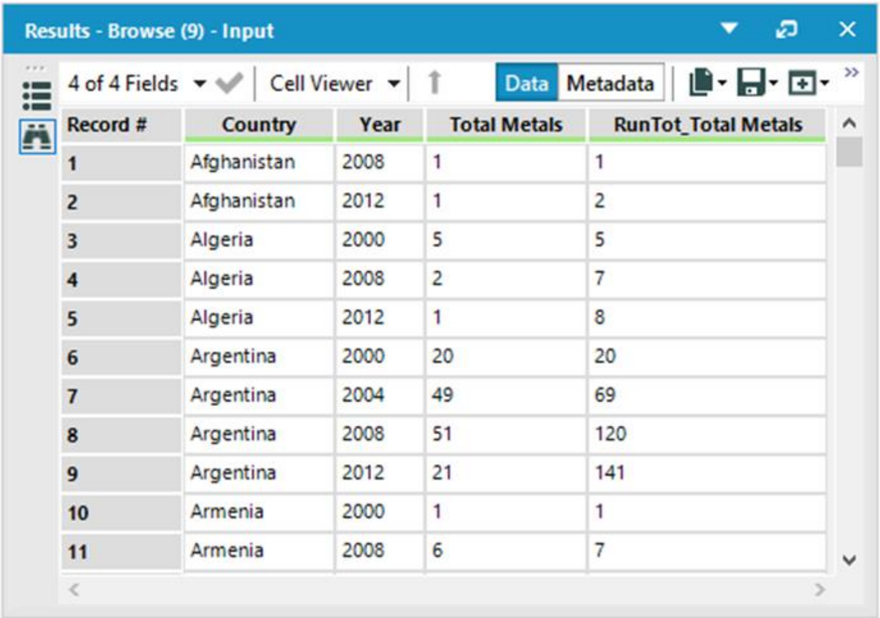


Figure-2-48-Modern History – Running Total Configuration

Let's take a look at what we have created so we can get a better sense of what the process so far has done.



Record #	Country	Year	Total Metals	RunTot_Total Metals
1	Afghanistan	2008	1	1
2	Afghanistan	2012	1	2
3	Algeria	2000	5	5
4	Algeria	2008	2	7
5	Algeria	2012	1	8
6	Argentina	2000	20	20
7	Argentina	2004	49	69
8	Argentina	2008	51	120
9	Argentina	2012	21	141
10	Armenia	2000	1	1
11	Armenia	2008	6	7

Figure-2-49-Modern History – Browse after Running Total

If we add a Browse tool and run the workflow, we can see that we have an alphabetical list of countries with a record for every year they won a medal. We can also see the year is increasing as we move down the list within a country. We then see the Total Medal count for that year and the running total of medals that the country has won from one year to the next in a field called RunTot_Total Medals.

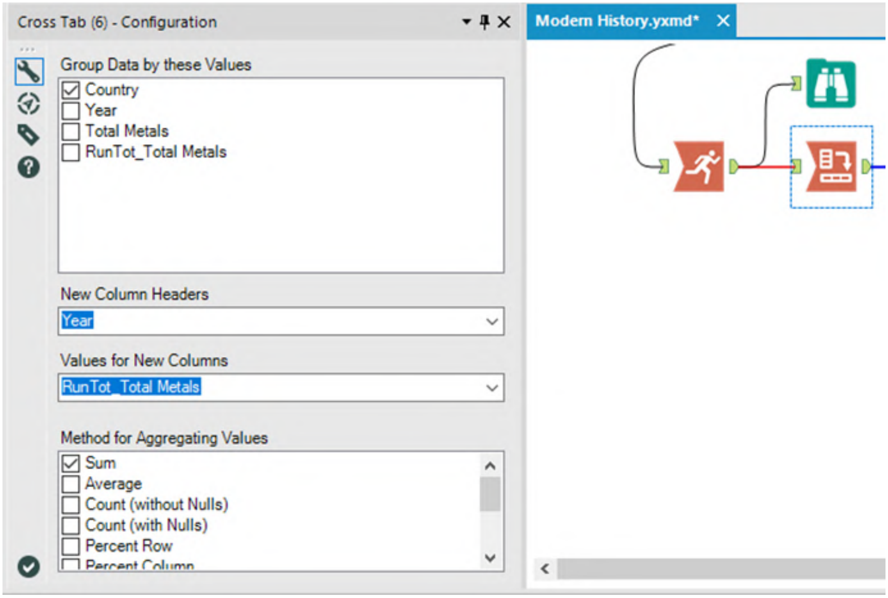
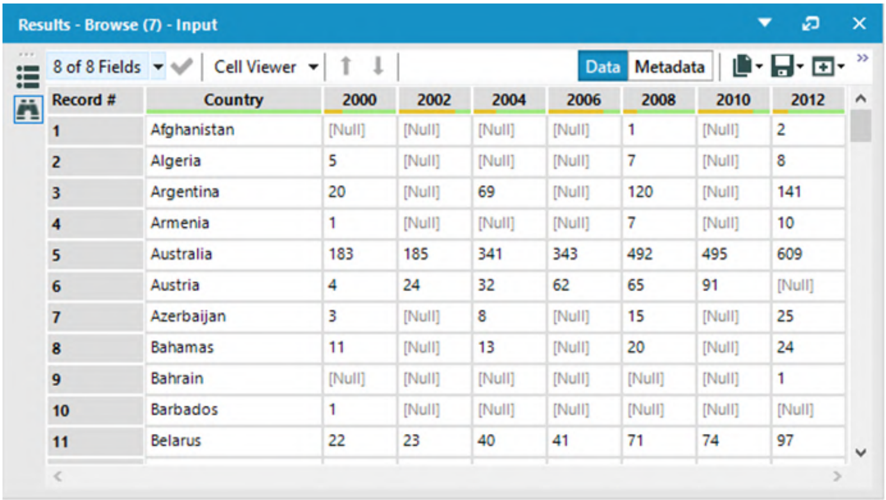


Figure-2-50-Modern History – Cross Tab

The next step in this process is to convert the data into a Cross Tab. If we add the Cross-Tab tool to the end of the data stream and apply the settings like in the above image, we will be close to our goal.

Let's add a browse tool and see what we have so far.



Record #	Country	2000	2002	2004	2006	2008	2010	2012
1	Afghanistan	[Null]	[Null]	[Null]	[Null]	1	[Null]	2
2	Algeria	5	[Null]	[Null]	[Null]	7	[Null]	8
3	Argentina	20	[Null]	69	[Null]	120	[Null]	141
4	Armenia	1	[Null]	[Null]	[Null]	7	[Null]	10
5	Australia	183	185	341	343	492	495	609
6	Austria	4	24	32	62	65	91	[Null]
7	Azerbaijan	3	[Null]	8	[Null]	15	[Null]	25
8	Bahamas	11	[Null]	13	[Null]	20	[Null]	24
9	Bahrain	[Null]	[Null]	[Null]	[Null]	[Null]	[Null]	1
10	Barbados	1	[Null]	[Null]	[Null]	[Null]	[Null]	[Null]
11	Belarus	22	23	40	41	71	74	97

Figure-2-51-Modern History – Browse after Cross Tab

The results seen in the image are close to what we want but not exactly what we want. We get the correct running totals for the years when each country won medals, however, we get nulls for the years they did not.

Now, what we need to do is create a series of formulae which will replace the nulls with zero or the previous value. As we need to create formulae, we will have to use the formula tool; but this time, we have to create seven similar calculations because we need to replace the values in seven different fields.

Let's think through these formulae. We want to change the value in the cell only if it is null. If the value in the column that we are fixing is 2000, then it should be replaced with 0, and if it is not 2000, then it should be replaced with whatever value is in the previously fixed column.

For those familiar with conditional statements, the syntax for an if-then statement is:

IF b1 THEN x ELSEIF b2 THEN y ELSE z ENDIF

For those unfamiliar with conditional statements, the concept is: Given a true or false (Boolean) expression, the calculation should do one of two things. The logic is if something is true, then do that; else, if the previous is false and something else is true, do the second option; else, do the default.

The other thing we need to know in creating these formulae is the test to see if something is null. The function used is:

IsNull(x)

Both of these syntaxes are under the functions tab in the Formula tool if we need to reference them.

The formulae that we need are:

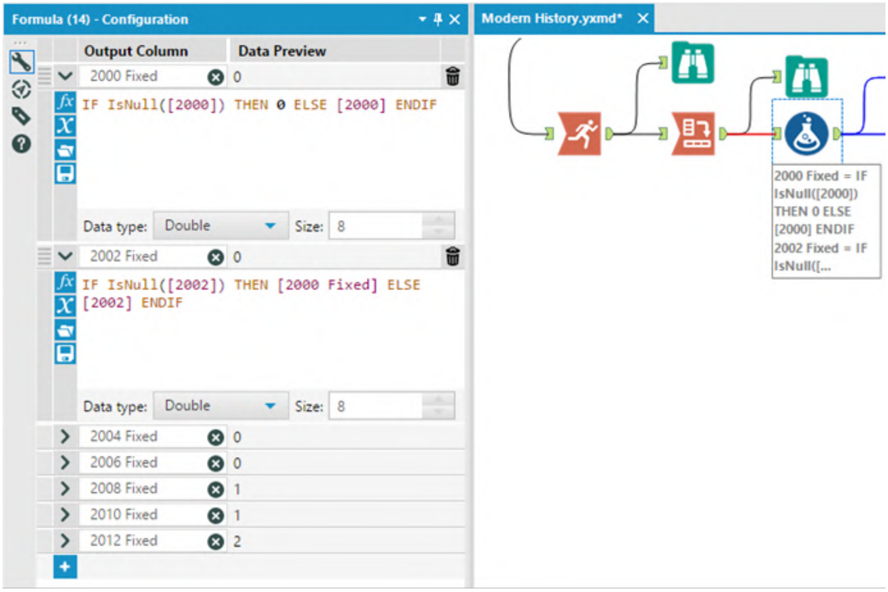
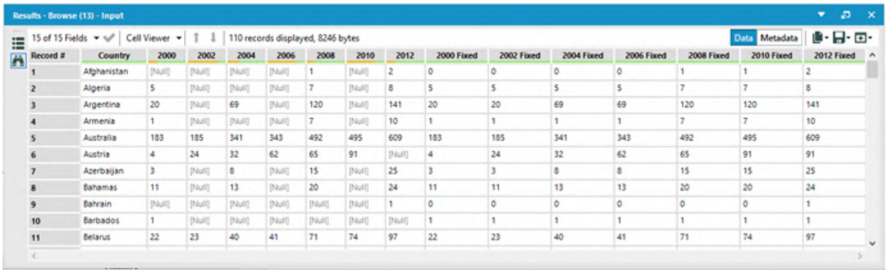


Figure-2-52-Modern History – Formula Configuration

Add a Formula tool to the end of the data stream and add the seven formulae which we see here with corresponding field names. We can also add a browse tool after that to see what we have created.

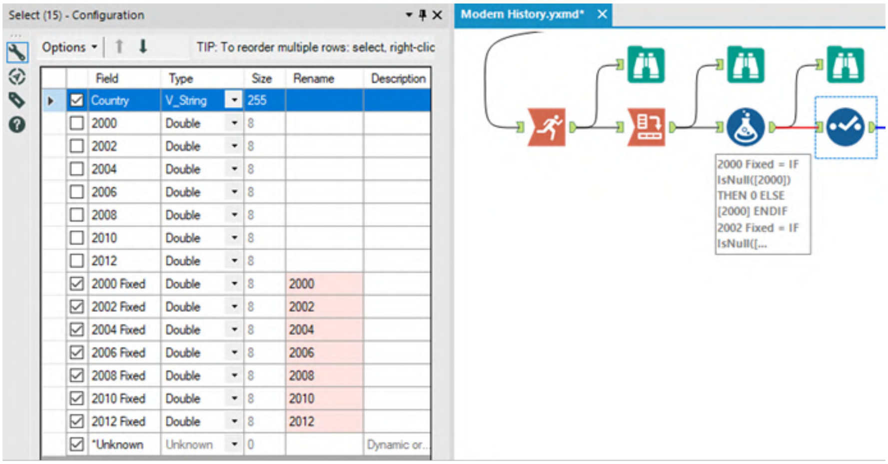
We can see that we have two sets of fields: those with the original sparse data and those with the new dense data.



Record #	Country	2000	2002	2004	2006	2008	2010	2012	2000 Fixed	2002 Fixed	2004 Fixed	2006 Fixed	2008 Fixed	2010 Fixed	2012 Fixed
1	Afghanistan	[Null]	[Null]	[Null]	[Null]	1	[Null]	2	0	0	0	0	1	1	2
2	Algeria	5	[Null]	[Null]	[Null]	7	[Null]	8	5	5	5	5	7	7	8
3	Argentina	20	[Null]	69	[Null]	120	[Null]	141	20	20	69	69	120	120	141
4	Armenia	1	[Null]	[Null]	[Null]	7	[Null]	10	1	1	1	1	7	7	10
5	Australia	183	185	341	343	492	495	609	183	185	341	343	492	495	609
6	Austria	4	24	32	62	65	91	[Null]	4	24	32	62	65	91	91
7	Azerbaijan	3	[Null]	8	[Null]	15	[Null]	25	3	3	8	8	15	15	25
8	Bahamas	11	[Null]	13	[Null]	20	[Null]	24	11	11	13	13	20	20	24
9	Bahrain	[Null]	[Null]	[Null]	[Null]	[Null]	[Null]	1	0	0	0	0	0	0	1
10	Barbados	1	[Null]	[Null]	[Null]	[Null]	[Null]	[Null]	1	1	1	1	1	1	1
11	Belarus	22	23	40	41	71	74	97	22	23	40	41	71	74	97

Figure-2-53-Modern History – Browse after Formula

The next thing that we need to do is remove and rename the columns that we have, so add a Select tool to the end of the data stream.



	Field	Type	Size	Rename	Description
☑	Country	V_String	255		
☐	2000	Double	8		
☐	2002	Double	8		
☐	2004	Double	8		
☐	2006	Double	8		
☐	2008	Double	8		
☐	2010	Double	8		
☐	2012	Double	8		
☑	2000 Fixed	Double	8	2000	
☑	2002 Fixed	Double	8	2002	
☑	2004 Fixed	Double	8	2004	
☑	2006 Fixed	Double	8	2006	
☑	2008 Fixed	Double	8	2008	
☑	2010 Fixed	Double	8	2010	
☑	2012 Fixed	Double	8	2012	
☑	*Unknown	Unknown	0		Dynamic or...

Figure-2-54-Modern History – Select Configuration

Now the data preparation is completed. We need to write the results because we were asked for a data set and not a specific answer. We should add a Browse tool and an Output Data tool to end the data flow. Save the output as *Historical Medal Count*.csv.

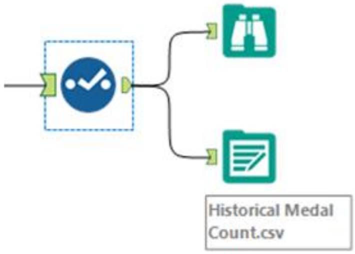


Figure-2-55-Modern History – Output

After doing these steps, the final workflow is as shown in the below figure.

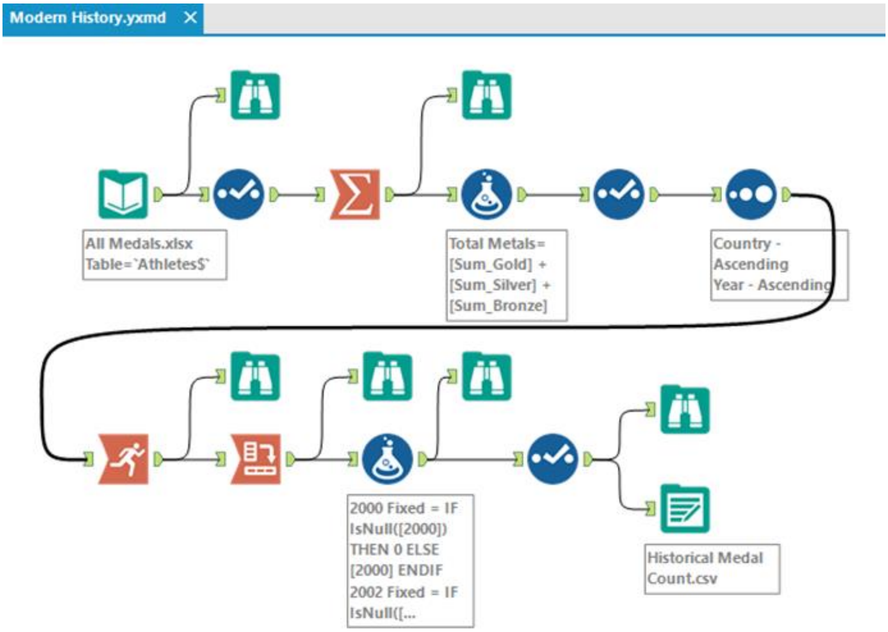
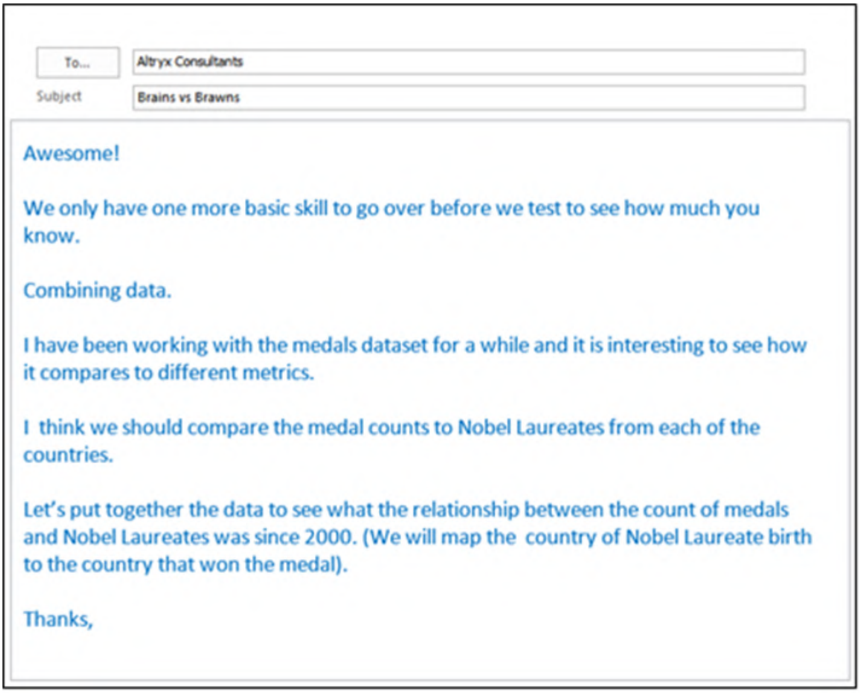


Figure-2-56-Modern History – Complete Workflow

2.5 Brains vs Brawns



To... Altryx Consultants

Subject Brains vs Brawns

Awesome!

We only have one more basic skill to go over before we test to see how much you know.

Combining data.

I have been working with the medals dataset for a while and it is interesting to see how it compares to different metrics.

I think we should compare the medal counts to Nobel Laureates from each of the countries.

Let's put together the data to see what the relationship between the count of medals and Nobel Laureates was since 2000. (We will map the country of Nobel Laureate birth to the country that won the medal).

Thanks,

Since we are combining data, let's revisit the analogy presented in the preface. When we look at a river, we see there are tributaries all along its length. Each of these tributaries may have gone through different terrain and could have started from different sources. When they merge together, they add whatever they carried along with them into the river they form.

To relate it to the task at hand, tributaries are branches of our data stream that come together, and when they come together, we have a richer data stream because we have the information that comes from everything contributing to it.

We are going to start by preparing the medals data to be joined. We'll create a table with two columns called Country and Medal Count. connect to the file *All Medals.xlsx* in *Chapter 2 – All Medals.xlsx and Nobel Laureates.csv.*

In order to do this, we are going to follow the following steps:

1. Import data
2. Transpose and rename the columns so that the data is tidy
3. Filter out the 0 medal records
4. Summarize the data so that we only have one record per country and the total medal count
5. Rename the medal count column *Total Medal Count*

Since we have covered the tools and the concepts used in this exercise in previous exercises, overall flow should look familiar. Please rebuild the following workflow with the following configurations.

The properties windows for each of these tools as well as the data stream that is produced are shown in the following figures.

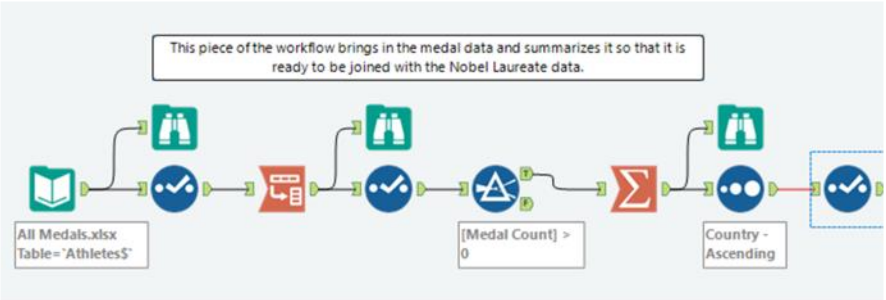


Figure-2-57-Brains vs Brawns – Medal Count Preparation

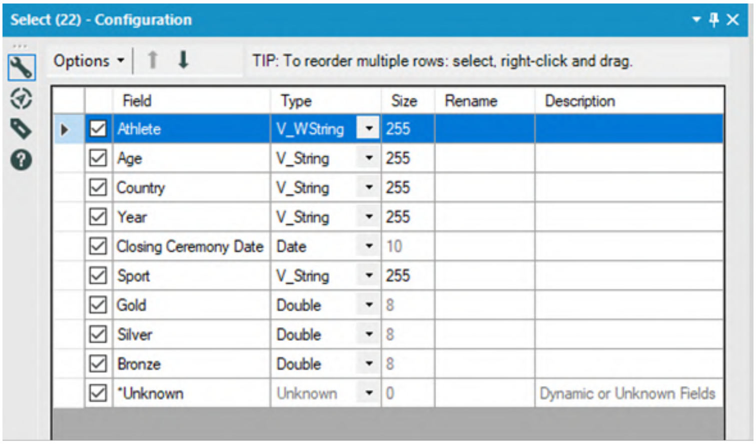


Figure-2-58-Brains vs Brawns – Initial Steps - Select

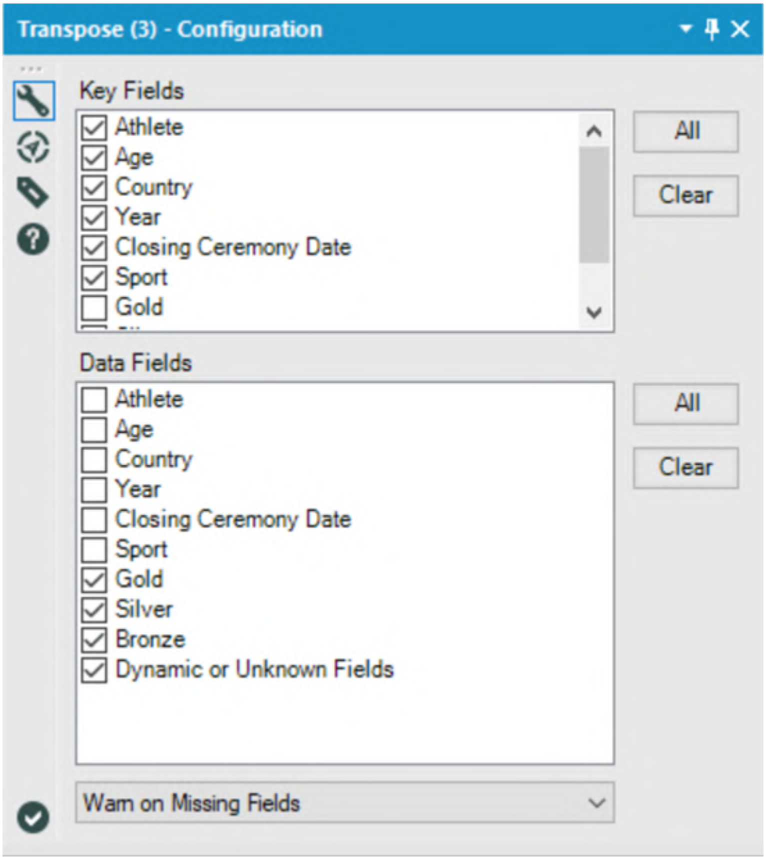


Figure-2-59-Brains vs Brawns – Initial Steps - Transpose

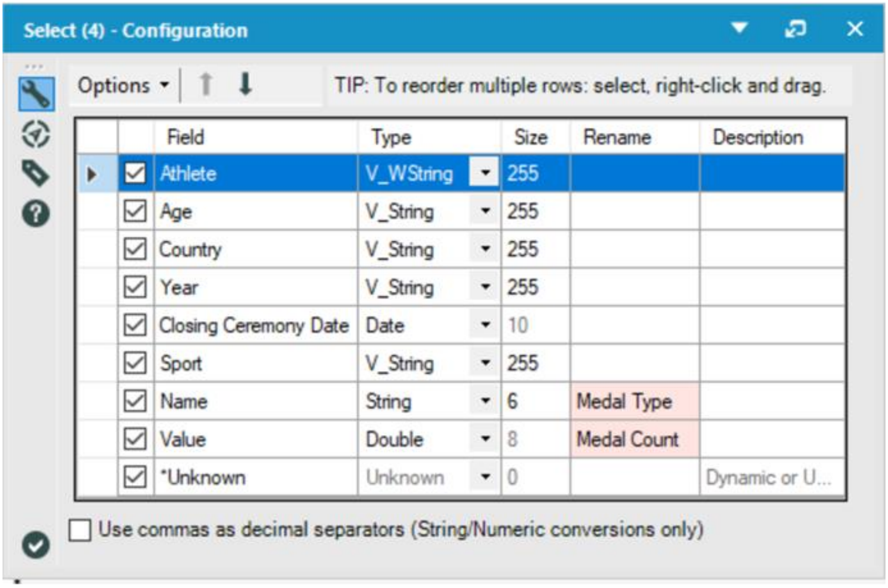


Figure-2-60-Brains vs Brawns – Initial Steps - Select

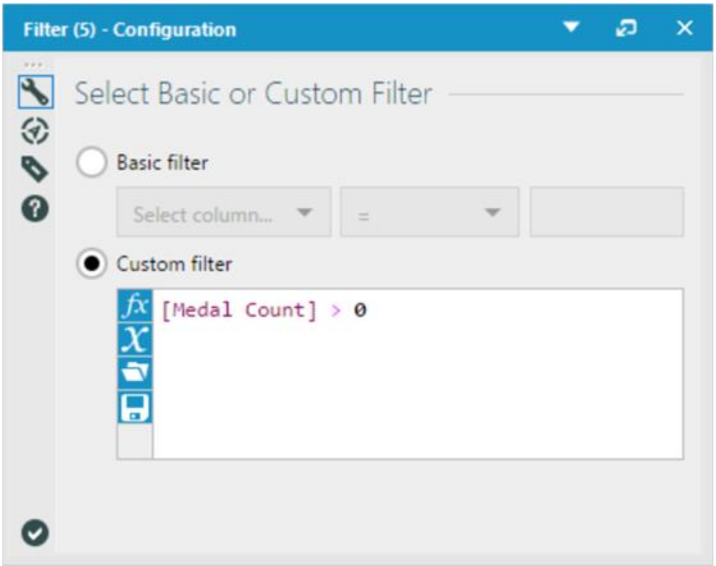


Figure-2-61-Brains vs Brawns – Initial Steps - Filter

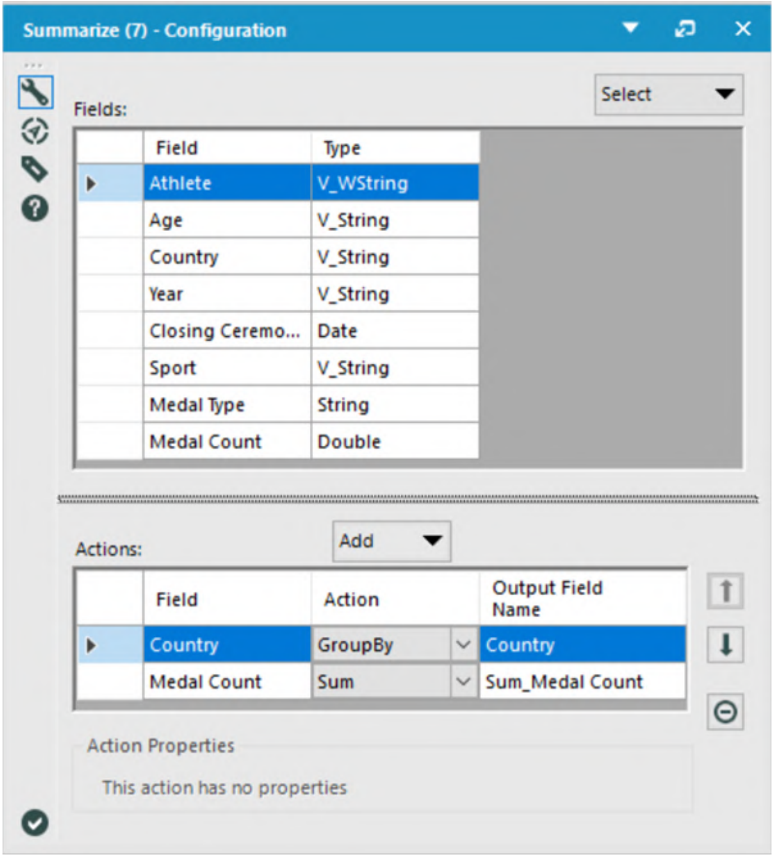


Figure-2-62-Summarize Configuration

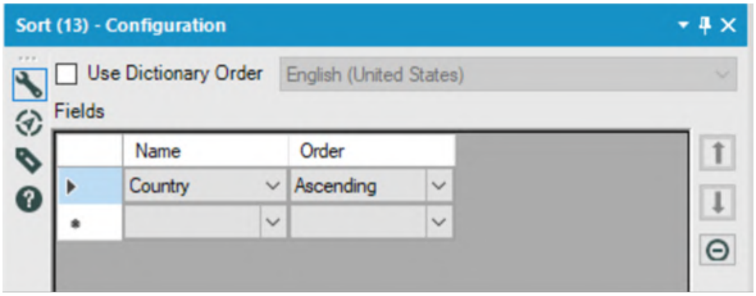


Figure-2-63-Brains vs Brawns – Initial Steps - Sort

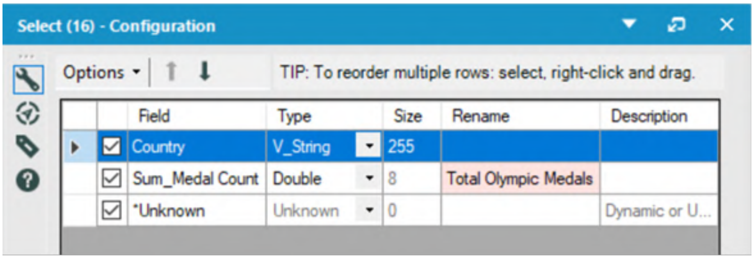


Figure-2-64-Brains vs Brawns – Initial Steps - Select

Now that we have the data in the above stream prepared to be combined, we should prepare the other contributing data stream.

Let us open the file called *Nobel Laureates.csv* in the folder Chapter 2 -- The Games > Nobel Laureates. (Remember that we should always bring in a Browse and Select Tool with an input.)

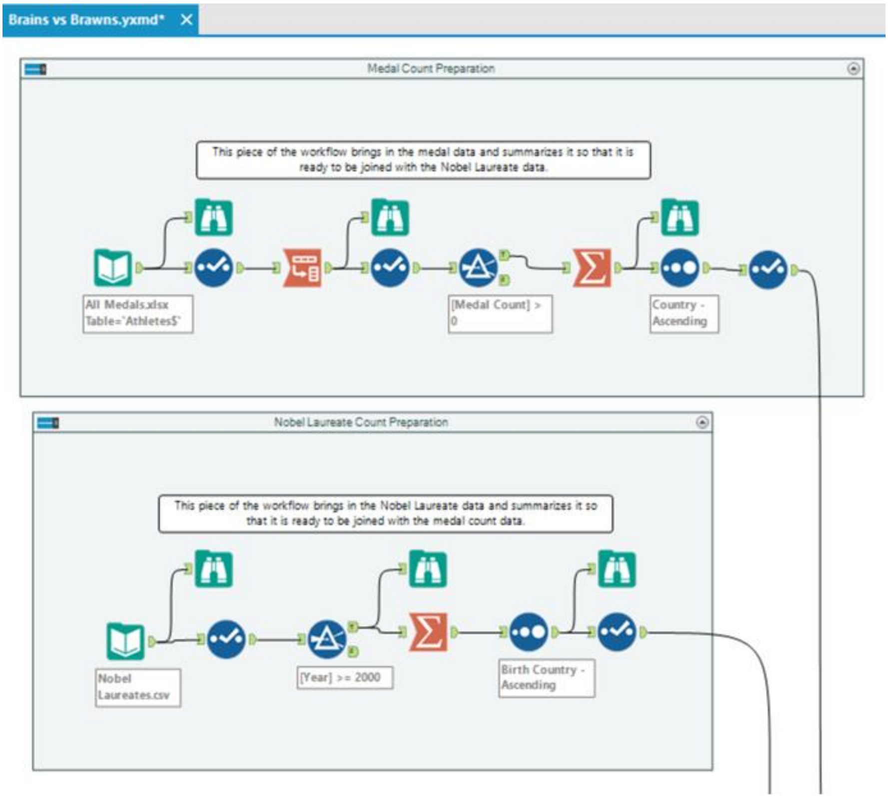
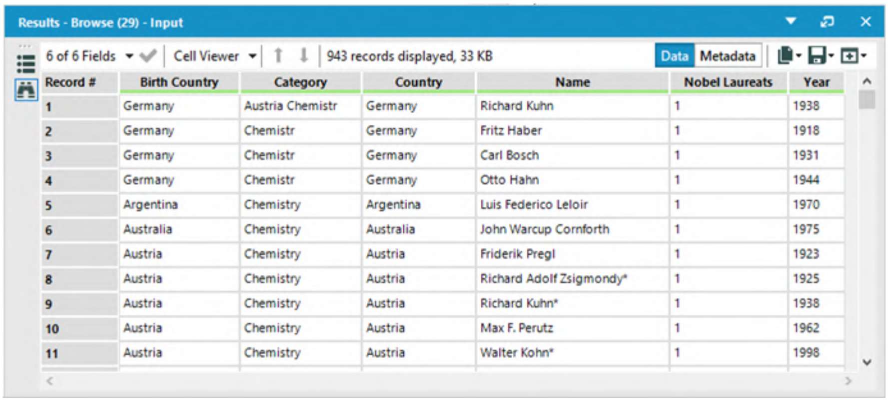


**Figure-2-65-Brains vs Brawns –
Running unrelated analyses simultaneously**

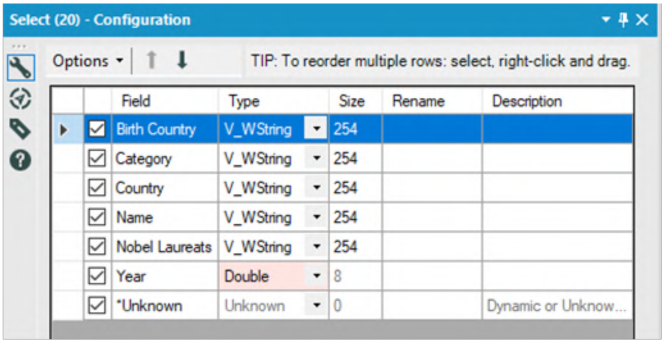
Notice that we now have two completely separate workflows. This is a useful feature because we can run unrelated analyses at the same time, which aids in testing and in conditional application development.

If we run the workflow, we can look at the structure of the *Nobel Laureates* dataset. Here, we want to make sure that the field we plan on joining (Birth Country) is of the same type as Country, in the medal data stream.



Record #	Birth Country	Category	Country	Name	Nobel Laureats	Year
1	Germany	Austria Chemistr	Germany	Richard Kuhn	1	1938
2	Germany	Chemistr	Germany	Fritz Haber	1	1918
3	Germany	Chemistr	Germany	Carl Bosch	1	1931
4	Germany	Chemistr	Germany	Otto Hahn	1	1944
5	Argentina	Chemistry	Argentina	Luis Federico Leloir	1	1970
6	Australia	Chemistry	Australia	John Warcup Cornforth	1	1975
7	Austria	Chemistry	Austria	Friderik Pregl	1	1923
8	Austria	Chemistry	Austria	Richard Adolf Zsigmondy*	1	1925
9	Austria	Chemistry	Austria	Richard Kuhn*	1	1938
10	Austria	Chemistry	Austria	Max F. Perutz	1	1962
11	Austria	Chemistry	Austria	Walter Kohn*	1	1998

Figure-2-66-Brains vs Brawns – Browse



Field	Type	Size	Rename	Description
Birth Country	V_WString	254		
Category	V_WString	254		
Country	V_WString	254		
Name	V_WString	254		
Nobel Laureats	V_WString	254		
Year	Double	8		
*Unknown	Unknown	0		Dynamic or Unknow...

Figure-2-67-Brains vs Brawns, Nobel Laureates - Select

As we can see that the data structure is similar to the medals file, we can start our preparation for the join.

We want to limit this data to years starting from 2000. One way we can do this is by converting Year to a Double Type and set up a filter to be **[Year] >= 2000**.

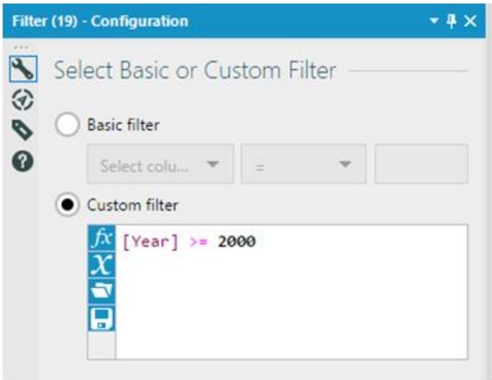


Figure-2-68-Brains vs Brawns, Nobel Laureates - Filter

Since we only need to know the total number of Nobel Laureates for each country of birth, we can summarize the data by country of birth.

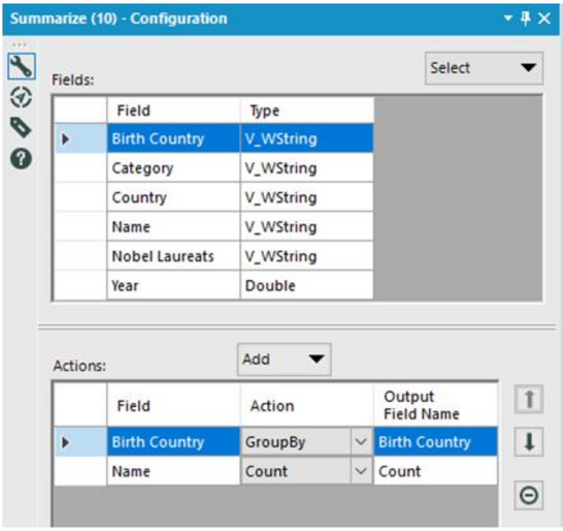


Figure-2-69-Brains vs Brawns, Nobel Laureates - Summarize

Now, we will sort the data in an alphabetical order of the country of birth using the Sort tool.

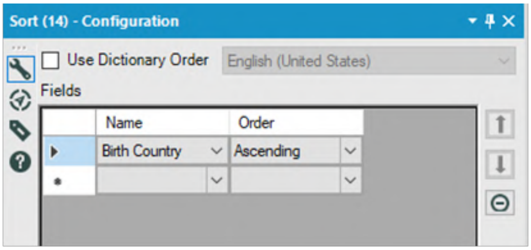
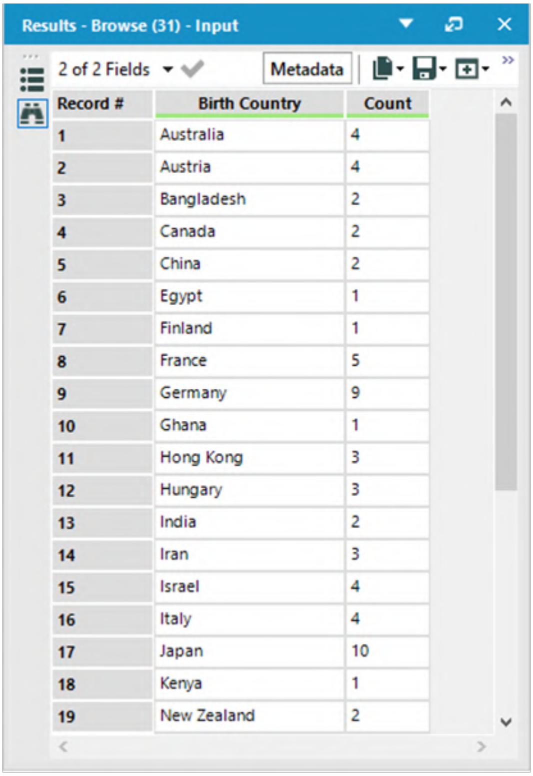


Figure-2-70-Brains vs Brawns, Nobel Laureates -Sort

We will put a Browse tool after Sort tool and see how the data looks like.



Record #	Birth Country	Count
1	Australia	4
2	Austria	4
3	Bangladesh	2
4	Canada	2
5	China	2
6	Egypt	1
7	Finland	1
8	France	5
9	Germany	9
10	Ghana	1
11	Hong Kong	3
12	Hungary	3
13	India	2
14	Iran	3
15	Israel	4
16	Italy	4
17	Japan	10
18	Kenya	1
19	New Zealand	2

Figure-2-71-Brains vs Brawns, Nobel Laureates –Browse after Sort

We see that we have a list of countries and a count of the number of Nobel Laureates. However, it is unclear what the number is because the field is called *Count*. We should rename it *Total Nobel Laureates* using Select tool.

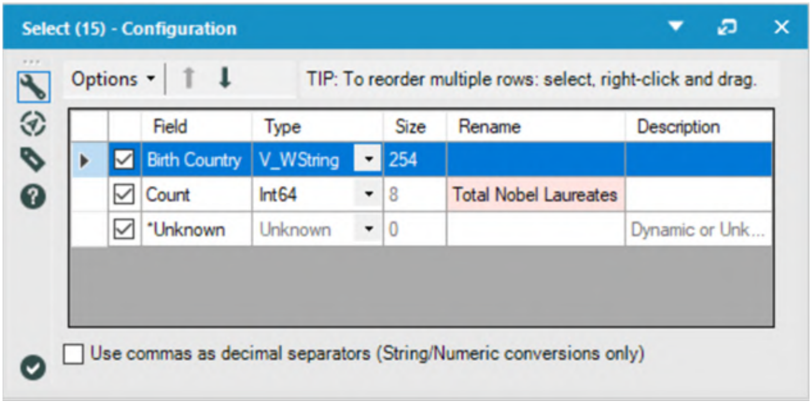


Figure-2-72-Brains vs Brawns, Nobel Laureates – Rename Count using Select

We now have two data streams ready to be merged. We want to align the two datasets so that matching countries from each of the data streams share the same record, which means we want to join the data. As we do not want to lose any data points if we have countries in one dataset but not the other, we will want to unite the three outputs from the join into a single data stream.

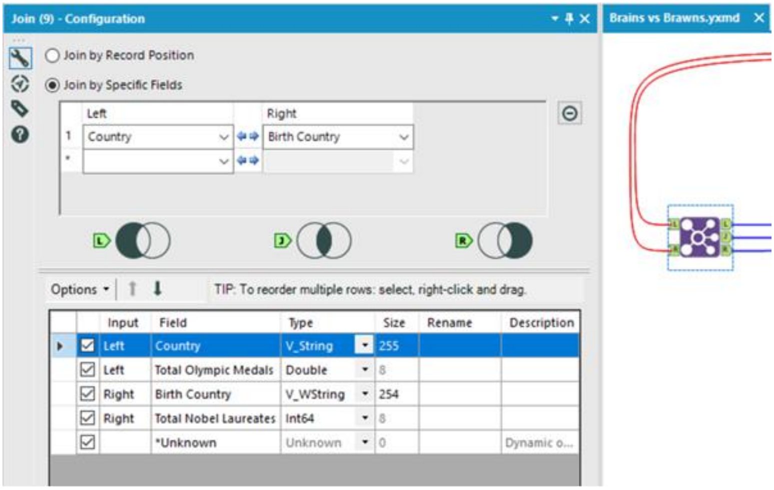


Figure-2-73-Brains vs Brawns, Nobel Laureates – Join Configuration

We want to join on Country field from the Left (Input L) with Birth Country field from the Right (Input R).

In this instance, it is important that we keep both the joining fields because we intend to combine all three outputs in the next step. However, if this was not our intention, we could have removed the joining field from one of the two inputs.

Best practice is to give useful names to every connection that enters a multiple connection anchor.

Thus, we can see in the following image that we have relabeled the connections from *#1*, *#2*, and *#3* to *Left*, *Join*, and *Right*.

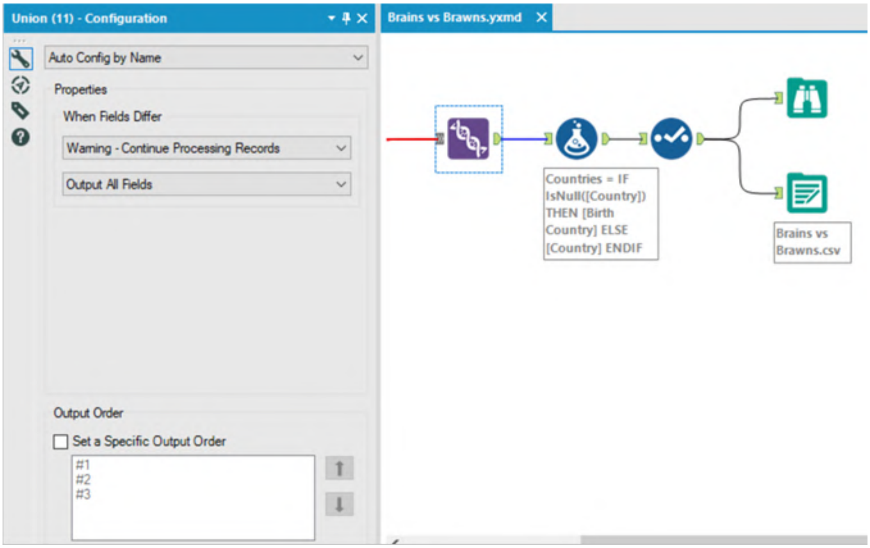


Figure-2-74-Brains vs Brawns, Nobel Laureates – Union Configuration

Since we are doing a union of three output streams of a Join tool, we know that we will have matching column names. This allows us to use the *Auto Config by Name* setting for the Union tool and leave the rest of the defaults.

We need to add a browse tool again as we have just altered the structure of the data. This is to make sure the data looks the way we expect. Notice that we are doing this after the Union and not the Join. That is because when we are combining the three output streams of a Join tool using a Union, we are performing a single logical step called an outer join. As this is a single step, we should check both the tools if any issue arises.

We are getting close to our goal; however; the data stream is also starting to become complex. So, we should take a minute to annotate what we have so it will be easier to follow later. We are going to add *Tool Containers* and *Comments* to the two contributing data streams so we can easily identify different parts of this data stream. We can create the comments and containers like we see in the following images.

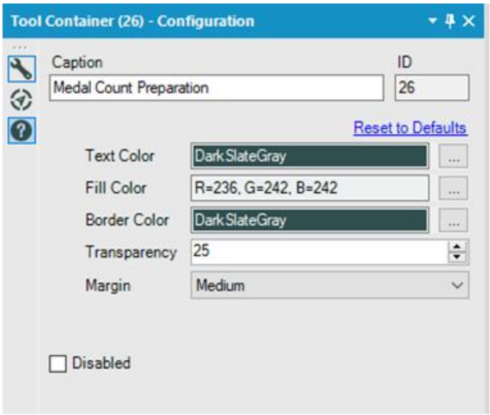


Figure-2-75-Brains vs Brawns, Medal Count Preparation – Tool Container Configuration

We can now drag the appropriate tools into the tool containers so the data stream is easier to understand.

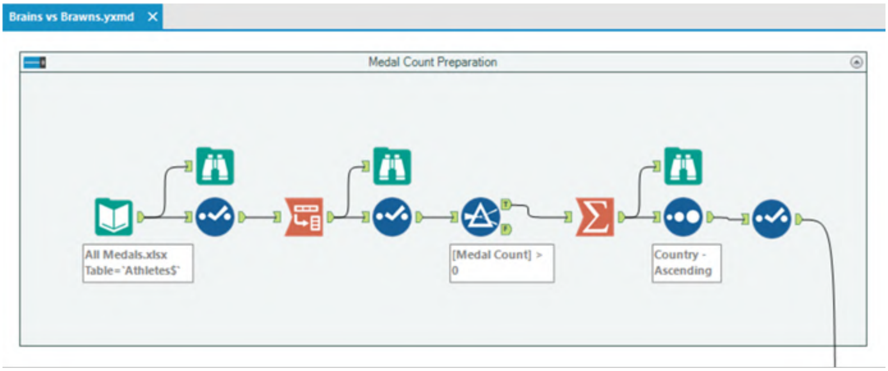


Figure-2-76-Brains vs Brawns - Medal Count Preparation stream with Tool Container

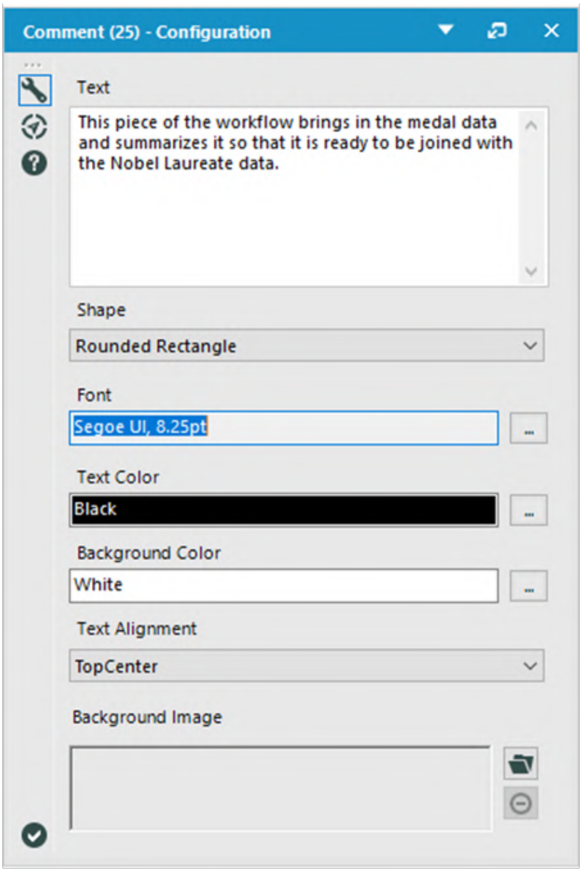


Figure-2-77-Brains vs Brawns, Medal Count Preparation – Comment Configuration

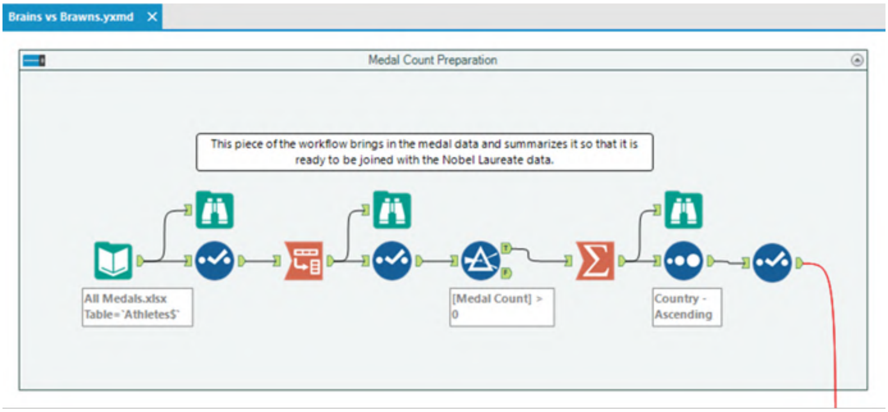


Figure-2-78-Brains vs Brawns - Medal Count Preparation Comment

Looking at the data stream this way is helpful, but if we click on the arrows at the top-right corner, we can condense what we are looking at.

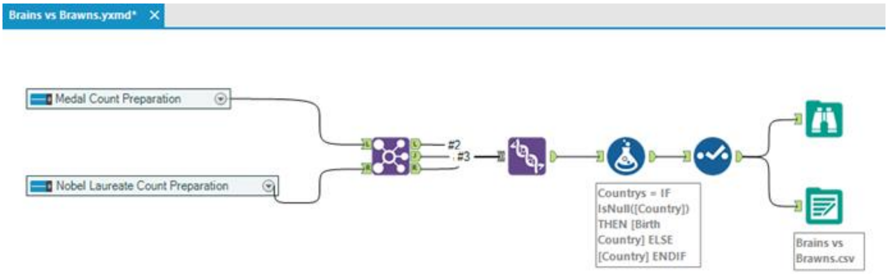


Figure-2-79-Brains vs Brawns – Simplified Workflow

We now see the medal count preparation and the Nobel Laureate count preparation as two separate processes instead of a series of tools. Now that we have made the data stream easier to understand, we should finish building the workflow.

We can observe from the Browse that the country names matched the names in both the Country and Birth Country fields. Let's create a conditional formula, **Countries** with the formula: **IF IsNull ([Country]) THEN [Birth Country] ELSE [Country] ENDIF**. This will take the Country value unless it is null and the Birth Country if it is.

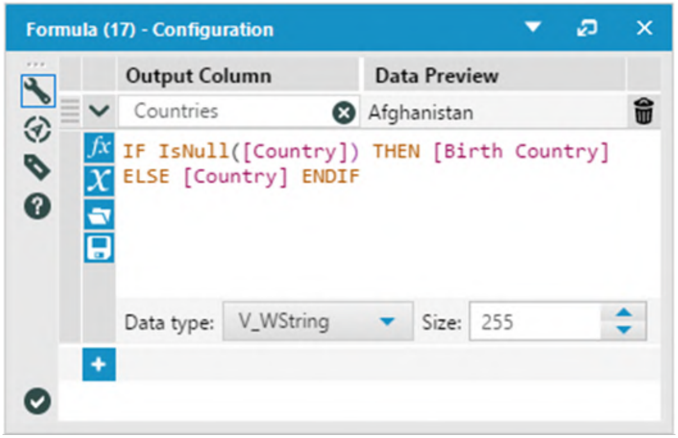


Figure-2-80-Brains vs Brawns, Final Output – Formula Configuration

Now we only need to clean up the data and export it to a .csv file. Add a Select tool to move *Countries* to the top as dimensions are expected to the left and uncheck *Country* and *Birth Country*. Then export the file to *Brains vs Brawns.csv*.

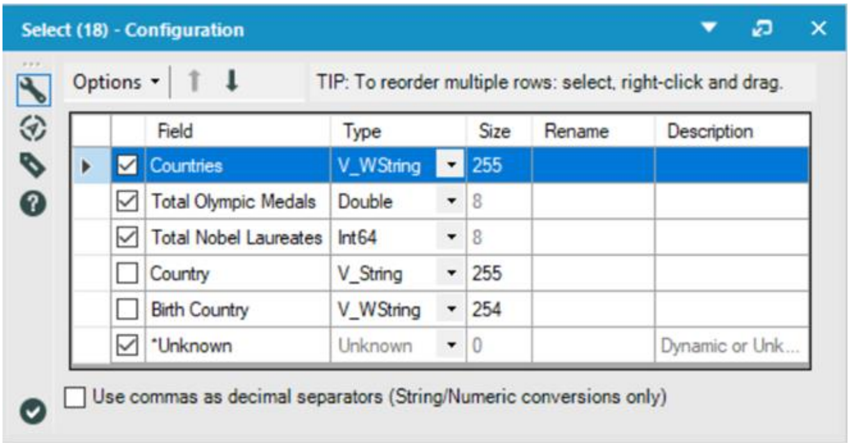
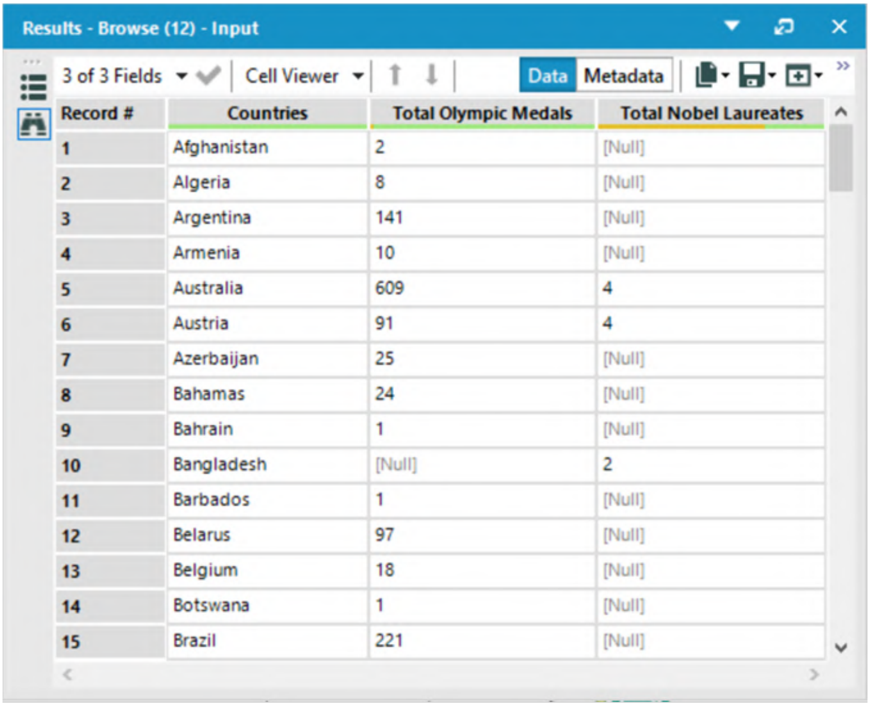


Figure-2-81-Brains vs Brawns, Final Output – Select Configuration

The output of the entire workflow can be explored by adding a Browse tool after the Select tool. After running the workflow, the result will be as shown below.



Record #	Countries	Total Olympic Medals	Total Nobel Laureates
1	Afghanistan	2	[Null]
2	Algeria	8	[Null]
3	Argentina	141	[Null]
4	Armenia	10	[Null]
5	Australia	609	4
6	Austria	91	4
7	Azerbaijan	25	[Null]
8	Bahamas	24	[Null]
9	Bahrain	1	[Null]
10	Bangladesh	[Null]	2
11	Barbados	1	[Null]
12	Belarus	97	[Null]
13	Belgium	18	[Null]
14	Botswana	1	[Null]
15	Brazil	221	[Null]

Figure-2-82-Brains vs Brawns, Final Output – Browse

The final workflow is as shown in the following figure.

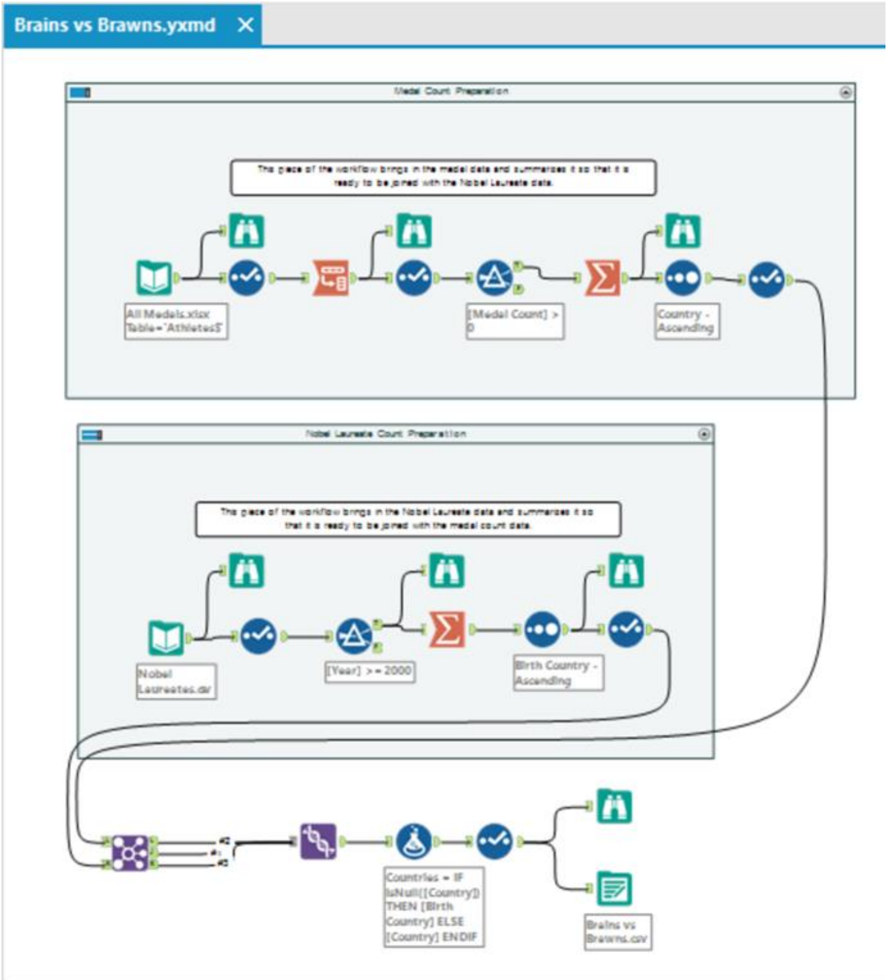


Figure-2-83-Brains vs Brawns – Complete Workflow

2.6 How are we doing?

Send

To... Alteryx Consultants

Cc...

Subject How are we doing?

Hey,

I just got a call from the Ad Agencies marketing team. They have got some shopper marketing data over a period of time and the forecast for the same.

They want us to help them with the following:

a. Get all the actuals into one dataset
b. Find the totals, Spend values for Products, Retailers and Ad Campaigns

And further,

a. Find the spend for Products from forecast. Compare the values from the respective Part A.
   If the difference in spend is 1000 then provide flags to highlight the forecast as over or under estimate
b. Find the spend for Retailers from forecast. Compare the values from the respective Part A, Arrange values from High to Low
c. Find the spend for Ad Campaign from forecast. Compare the values from the respective Part A.
   If the difference in spend is 2000 then provide flags to highlight them as over/under estimate.

I am into meetings for the rest of the afternoon, I cannot show them the turn around, but, I am sure you know enough to get these in place.

If we can get this back to them in next couple of hours, I think it would guarantee that they go with us.

Thanks

CHAPTER 3

Unisex Baby Names

From

To... Alteryx Consultants

Cc...

Subject Gender Swapped

Attached Unisex Names.txt

Hey,

A major newspaper is considering writing an article on the change in assignment of names to different sexes, and they would like our help finding out if there is anything definitive that they can say.

They provided the attached file that has unisex baby names from 1880 to 2013 and the count of children given each name by sex.

This is a pretty interesting dataset, so I am going to work on this with you.

The first things they want to know are: Are there any names that between 1880 and 1889 were completely assigned to one sex and then completely assigned to another between 2004 and 2013? If so, which were they?

Thanks,

3.1 Tools & Concepts

3.1.1 Imputation
Concept – Handling Nulls

<table>
<tr><td rowspan="2">
Figure-3-1-Imputation</td><td>Tool Palette: Preparation</td></tr>
<tr><td>Replaces a specific value in numeric fields. It is most commonly used to handle Null values in formulas.
For more details use the link below.
bit.ly/2HWxpHx</td></tr>
</table>

3.1.2 Multi-Field Formula
Concept – Multiple Record Calculations

<table>
<tr><td rowspan="2">
Figure-3-2-Multi-Field Formula</td><td>Tool Palette: Preparation</td></tr>
<tr><td>Provides the ability to create a formula that will be reused across multiple fields.
For more details use the link below.
bit.ly/2Joaqo2</td></tr>
</table>

3.1.3 Multi-Row Formula
Concept – Repetitive Calculations

	Tool Palette: Preparation
 Figure-3-3-Multi-Row Formula	Provides the ability to create a formula that will reference other records. For more details use the link below. bit.ly/2KcFXKX

3.1.4 Text Input
Concept – Ad Hoc Data

	Tool Palette: In/Out
 Figure-3-4-Text Input	Allows the creation of datasets to be used by typing in the data. For more details use the link below. bit.ly/2qVqI0n

3.1.5 Data Cleansing
Concept – Problem Analysis

	Tool Palette: Preparation
 Figure-3-5-Data Cleansing	Fixes common data quality issues using a variety of parameters. For more details use the link below. bit.ly/2JofuZA

3.2 Gender Swapped

From

To... Alteryx Consultants

Cc...

Subject Gender Swapped

Attached Unisex Names.txt

Hey,

A major newspaper is considering writing an article on the change in assignment of names to different sexes, and they would like our help finding out if there is anything definitive that they can say.

They provided the attached file that has unisex baby names from 1880 to 2013 and the count of children given each name by sex.

This is a pretty interesting dataset, so I am going to work on this with you.

The first things they want to know are: Are there any names that between 1880 and 1889 were completely assigned to one sex and then completely assigned to another between 2004 and 2013? If so, which were they?

Thanks,

Let us refer to the *Unisex Names.txt* file in the folder *Chapter 3 – Unisex Names.txt*. It is a tab-delineated file. We need to make sure the file pop-up window looks like the image represented here. Let us save the workflow as *Gender Swapped*.

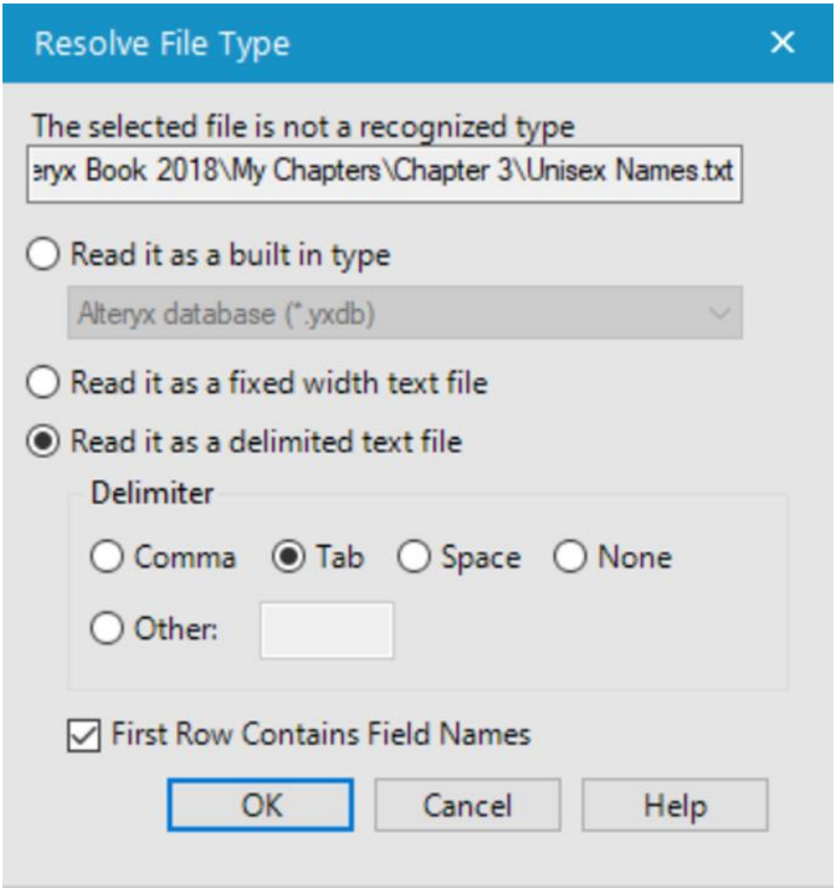
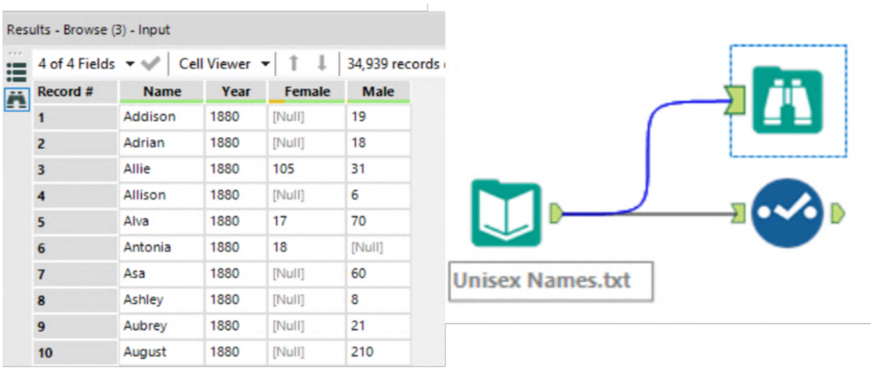


Figure-3-6 – Tab delineated pop-up window

Add a *Browse* tool and a *Select* tool to the input. After running the workflow, we should be able to see the records in the file.



Record #	Name	Year	Female	Male
1	Addison	1880	[Null]	19
2	Adrian	1880	[Null]	18
3	Allie	1880	105	31
4	Allison	1880	[Null]	6
5	Alva	1880	17	70
6	Antonia	1880	18	[Null]
7	Asa	1880	[Null]	60
8	Ashley	1880	[Null]	8
9	Aubrey	1880	[Null]	21
10	August	1880	[Null]	210

Figure-3-7-Gender Swapped Browse

If we take a look at the data, we will see that some numeric fields have *nulls* in them. Whenever values are missing, Alteryx fills those cells with a "[Null]" placeholder. We could leave these values as nulls; however, it is easier to work with numeric fields when they do not have nulls in them. Instead, what we will do is impute the nulls to 0s.

Unfortunately, Alteryx did not recognize these fields as numeric, so we need to make modifications in the *Select* tool so that *Female* and *Male* are converted into *Double*.

Since we know we need to work with two 10-year periods, we should also convert *Year* to *Double* because it will make our formulas simpler and faster when we filter and flag the data.

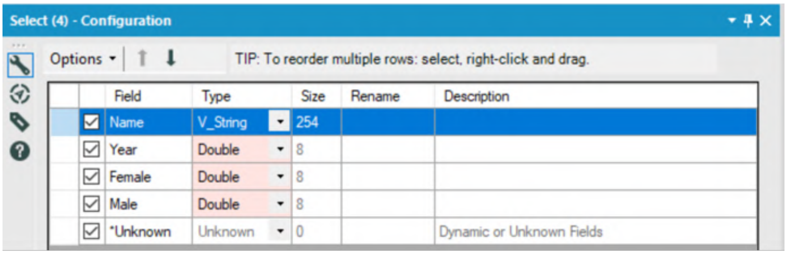


Figure-3-8 – Gender Swapped Select

If we add the *Imputation* tool with the settings as seen in the image, the tool will convert all of the null fields in *Year, Female* and *Male* fields to zeros.

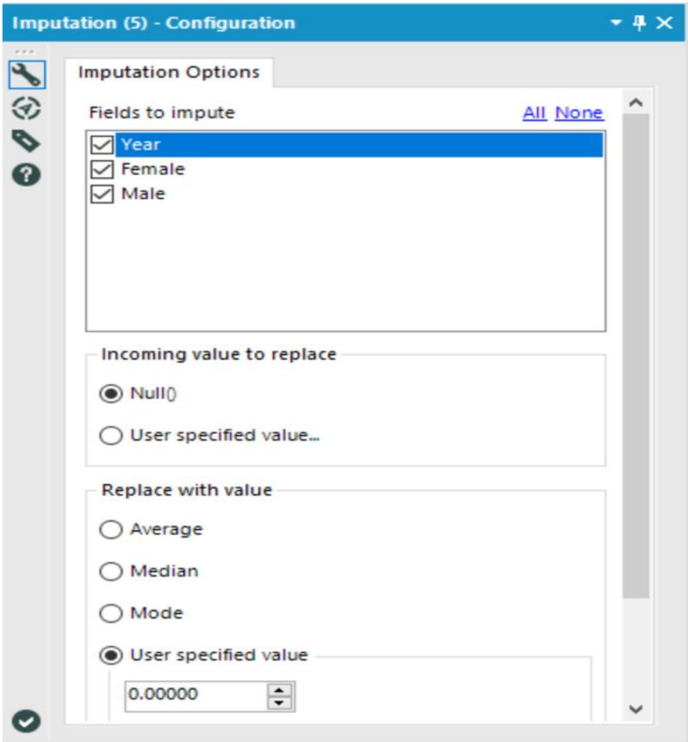


Figure-3-9-Gender Swapped Imputation Configuration

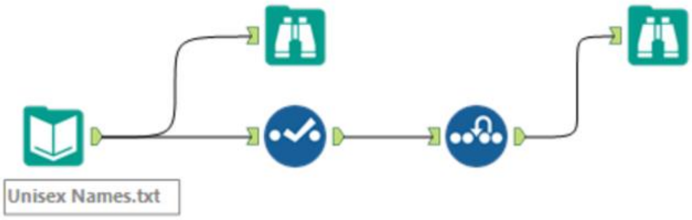
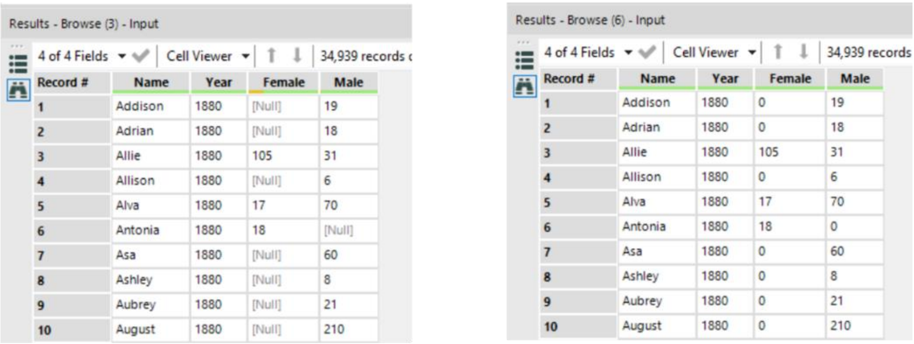


Figure-3-10-Gender Swapped data stream

Now let's see how the data has changed.



Record #	Name	Year	Female	Male
1	Addison	1880	[Null]	19
2	Adrian	1880	[Null]	18
3	Allie	1880	105	31
4	Allison	1880	[Null]	6
5	Alva	1880	17	70
6	Antonia	1880	18	[Null]
7	Asa	1880	[Null]	60
8	Ashley	1880	[Null]	8
9	Aubrey	1880	[Null]	21
10	August	1880	[Null]	210

Record #	Name	Year	Female	Male
1	Addison	1880	0	19
2	Adrian	1880	0	18
3	Allie	1880	105	31
4	Allison	1880	0	6
5	Alva	1880	17	70
6	Antonia	1880	18	0
7	Asa	1880	0	60
8	Ashley	1880	0	8
9	Aubrey	1880	0	21
10	August	1880	0	210

Figure-3-11- Gender Swapped Browse – Before and After

We can create a filter that limits the data set to *Years before 1890 or years after 2003* to isolate the two date ranges 1880 to 1889 and 2004 to 2013. We will add a filter to the end of the data stream. Use the expression box below to verify the Boolean formula.

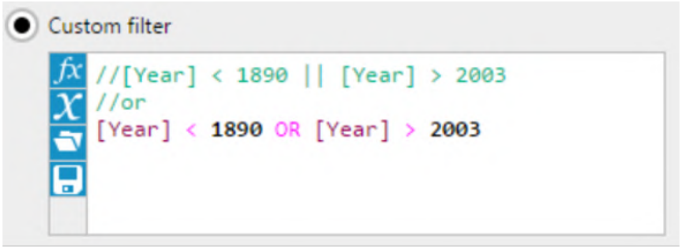


Figure-3-12 – Gender Swapped expression box

Notice that in this expression, we have two lines starting with //. These two forward slashes mean that the line is a comment. It is a way to tell Alteryx to ignore that specific line from the calculation. We used it to show the two intended methods of filtering this data, but it can be used to maintain old versions of the formulas or leave descriptions of the formula for future benefits.

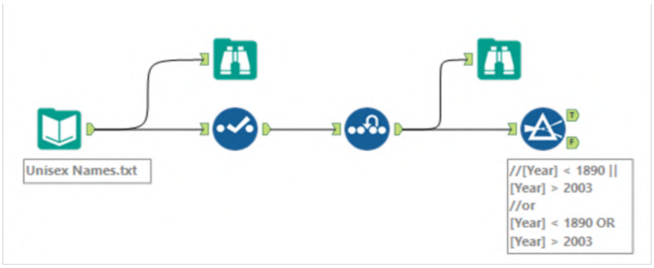


Figure-3-13-Gender Swapped data stream

The next thing we can do is create a *flag* for the first and last 10 years for the dataset. We want to do this to create a comparison between the two-time frames. In order to do this, we will create a string field that will have *First 10 Years* or *Last 10 Years* in the field.

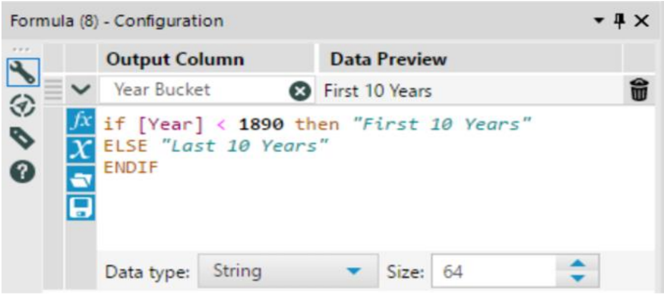


Figure-3-14-Gender Swapped Formula Configuration

We need to change the field type to string with a conditional formula to test if the record is in the first 10 years. Since the data is already filtered, we do not need to test the other values.

Now that we identified the data with the *Year Bucket* field, we can summarize the data to find out the total number of babies given each name during the 10 years we want to analyze.

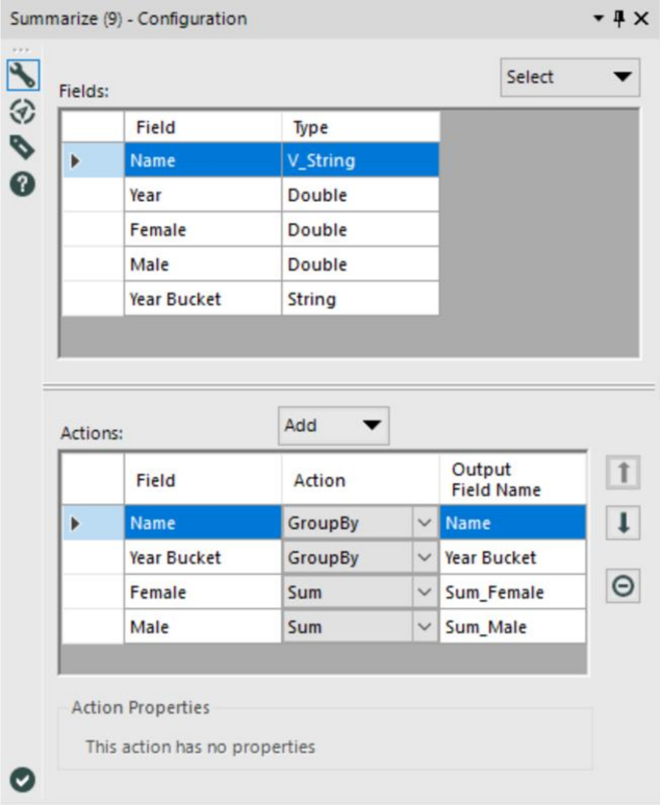


Figure-3-15- Gender Swapped Summarize Configuration

The data stream after *Summarize* looks as below:

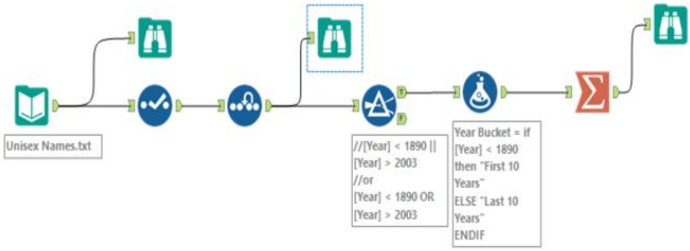
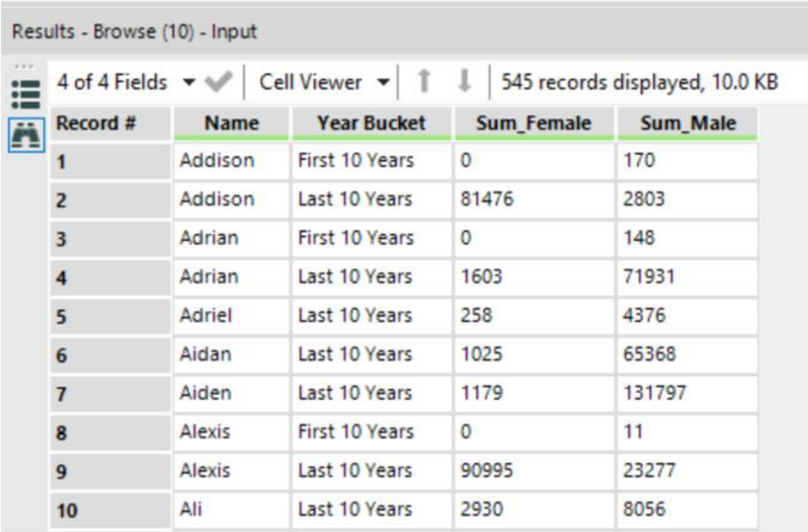


Figure-3-16-Gender Swapped data stream after summarize

We will next run the workflow and look at the *Browse* tool to see the data we have.



Results - Browse (10) - Input

4 of 4 Fields | Cell Viewer | 545 records displayed, 10.0 KB

Record #	Name	Year Bucket	Sum_Female	Sum_Male
1	Addison	First 10 Years	0	170
2	Addison	Last 10 Years	81476	2803
3	Adrian	First 10 Years	0	148
4	Adrian	Last 10 Years	1603	71931
5	Adriel	Last 10 Years	258	4376
6	Aidan	Last 10 Years	1025	65368
7	Aiden	Last 10 Years	1179	131797
8	Alexis	First 10 Years	0	11
9	Alexis	Last 10 Years	90995	23277
10	Ali	Last 10 Years	2930	8056

Figure-3-17-Gender Swapped Browse Tool

From the *Browse* tool, we can see that we have the data structured in a way we need. However, since we were asked to find the baby names that were previously only assigned to one sex and now are only assigned to the other, we need a way to check each of the names to see if they meet these criteria.

We will do this by using *Multiple Row Formula* to identify which records should be kept. Since we are using a Multi Row Formula, we need to make sure the data is in the right order. So, we will first use a *Sort* tool.

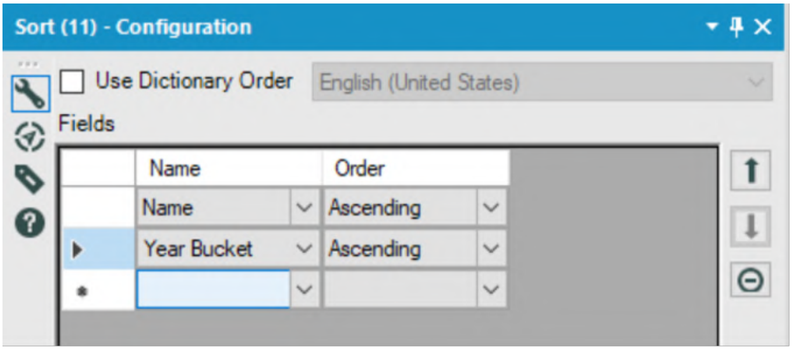


Figure-3-18-Gender Swapped Sort Configuration

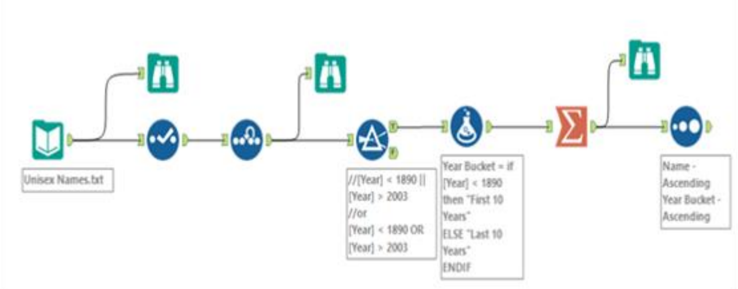


Figure-3-19-Gender Swapped data stream after adding sort tool

Now that we have made sure the data is sorted, we need to tackle the complex question. In order to do so, we will break it down into smaller, simpler ones:

- Which names occur in both the first and last 10-year lists?
- Which names have at least one count of zero?
- Which names have a female count alternate between something and zero?
- Which names have a male count alternate between something and zero?

By asking these four relatively simple questions and filtering out the data that does not meet the criteria, we can answer the complex question we have been asked. We will use the same field as we move forward to filter the data, which we will call *Potential Swap*. We will create it in the first *Multi-Row Formula* tool and update it in all subsequent ones. Let us see how we can create this formula.

At this point, much of this formula should be familiar. We are looking at a conditional statement that tests two things, and if both are true, we write *Remove*; else, we write *Check*. What is different about this is that we have special operators in the field names *Row-1:* and *Row+1:*. These allow us to look at the row (record) above and below the current one so we can use the value there. In this case, we are checking to see if the name of the current row (*[Name]*) equals the previous or following rows.

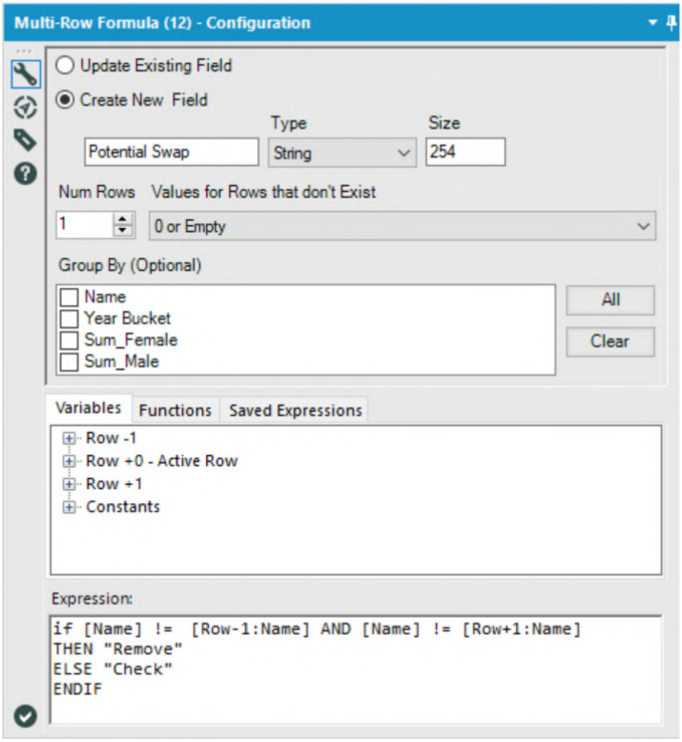


Figure-3-20-Gender Swapped Multi-Row Formula Configuration

Those familiar with Boolean logic may have understood what the formula here actually does. The formula checks to see if the *[Name]* does not match both the previous and following rows. This is a logically equivalent statement called the *contrapositive*, and we are bringing it up here to demonstrate two things: First, *!=* is the *logical operator* that means *does not equal*. The second is to show if we are having trouble with a logical statement. There are multiple ways one can approach it.

Between each *Multi-Row Formula*, we can add a *Filter* to improve speed by removing records we know are not needed for analysis. Let us add a series of alternating *Filters* and *Multi-Row Formulas* to answer the remaining three questions.

All *Filter* configuration windows for the next three steps should look like the image represented here.

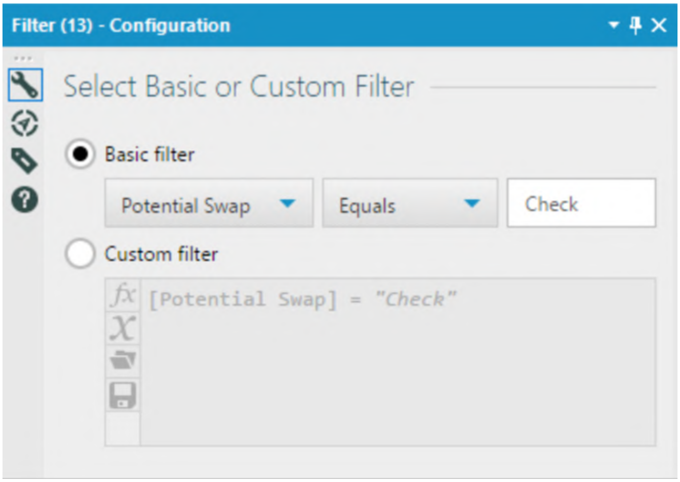


Figure-3-21 Gender Swapped Filter Configuration

The remaining *Multi-Row Formula* properties windows should look like the provided image, with the following table of equations in the expression box.

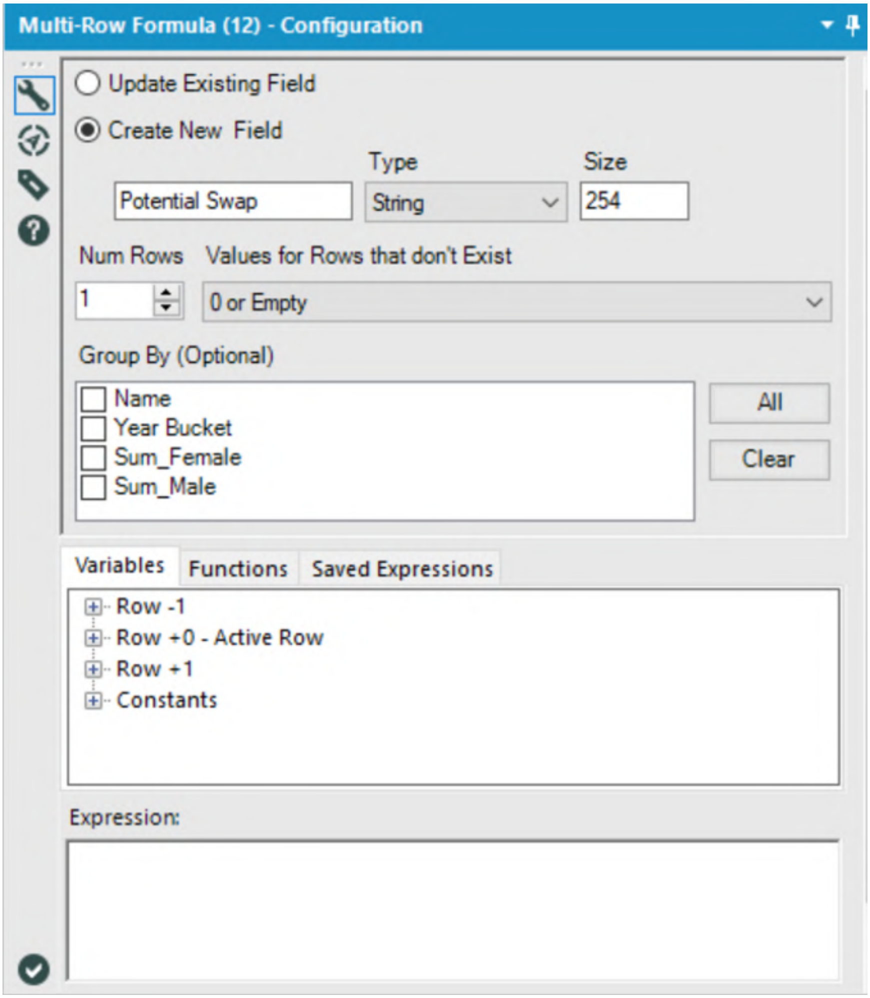


Figure-3-22-Gender Swapped remaining Multi-Row Formula

1	IF [Sum_Female] == 0 or [Sum_Male] == 0 THEN "Check" ELSEIF [Name] == [Row-1:Name] AND ([Row-1:Sum_Female] == 0 or [Row-1:Sum_Male] == 0) THEN "Check" ELSEIF [Name] == [Row+1:Name] AND ([Row+1:Sum_Female] == 0 or [Row+1:Sum_Male] == 0) THEN "Check" ELSE "Remove" ENDIF
2	IF [Name] = [Row-1:Name] AND [Sum_Female] != 0 AND [Row-1:Sum_Female] != 0 THEN "Remove" ELSEIF [Name] = [Row+1:Name] AND [Sum_Female] != 0 AND [Row+1:Sum_Female] != 0 THEN "Remove" ELSE "Check" ENDIF
3	IF [Name] = [Row-1:Name] AND [Sum_Male] != 0 AND [Row-1:Sum_Male] != 0 THEN "Remove" ELSEIF [Name] = [Row+1:Name] AND [Sum_Male] != 0 AND [Row+1:Sum_Male] != 0 THEN "Remove" ELSEIF [Name] = [Row+1:Name] AND [Sum_Male] !=0 AND [Row+1:Sum_Male] !=0 THEN "Remove" ELSE "Check" ENDIF

Figure 3-23 - Gender Swapped Formula

At this point, the data stream is quite long, so we have moved the tools below one another to make it easier to see.

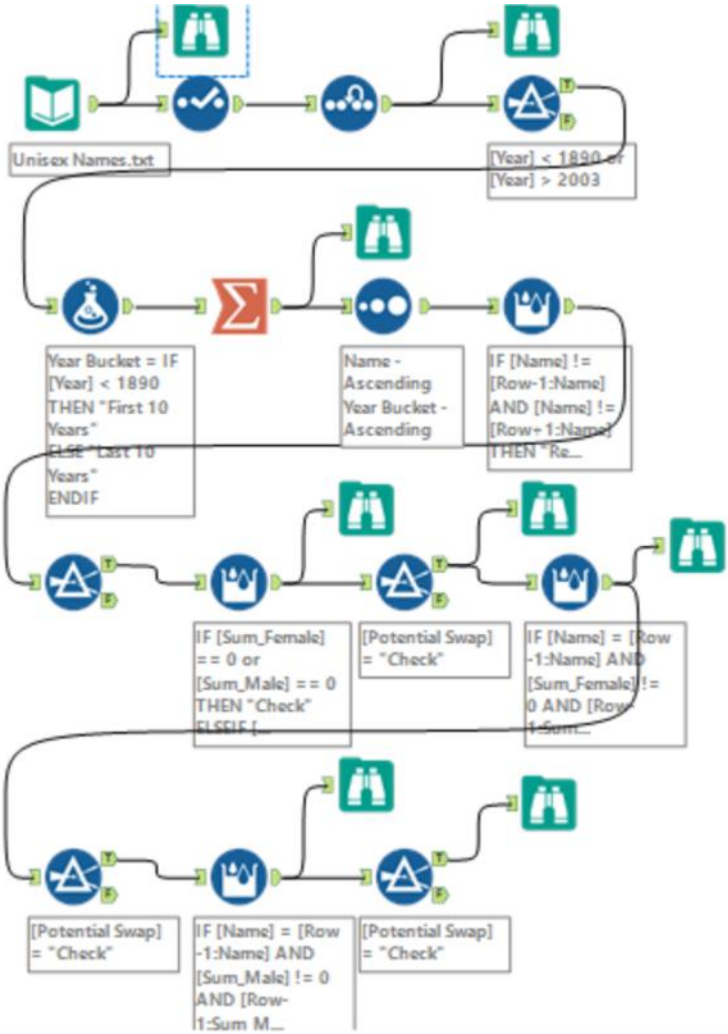
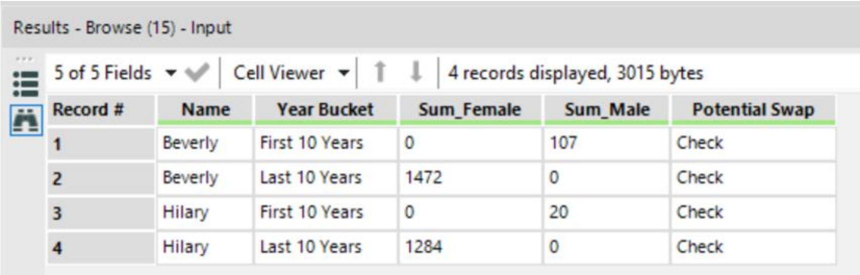


Figure-3-24-Gender Swapped Data Stream

We will notice that a *Browse* tool has been added here. Typically, if the formulas are being tested, then a *Browse* tool should be used on both the *True* and *False* outputs of every *Filter* tool. However, since these formulas were already tested, we are just going to take a look at the end result.



Record #	Name	Year Bucket	Sum_Female	Sum_Male	Potential Swap
1	Beverly	First 10 Years	0	107	Check
2	Beverly	Last 10 Years	1472	0	Check
3	Hilary	First 10 Years	0	20	Check
4	Hilary	Last 10 Years	1284	0	Check

Figure-3-25-Gender Swapped Browse Configuration

We can now see that Beverly and Hilary are the only names that meet our criteria.

However, just like in the *Freestyle Skiing* example, we should make Alteryx give us exactly the results we will give to the newspaper so we do not make a mistake. This time, all we need to do is *Summarize* by the *Name* field to have the results that we are looking for.

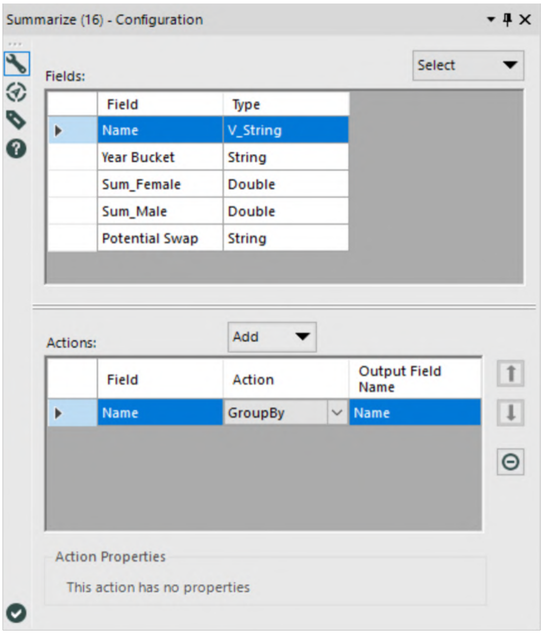


Figure-3-26-Gender Swapped Summarize Configuration

Running the *Browse* tool shows us the following:

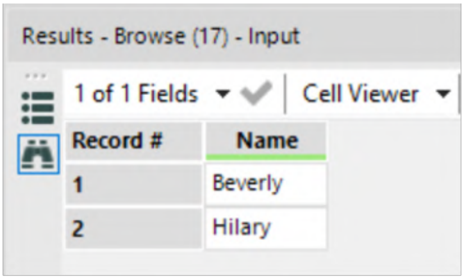


Figure-3-27-Gender Swapped Browse after summarize

The *Gender Swapped* data stream should look like the image on the next page when it is complete.

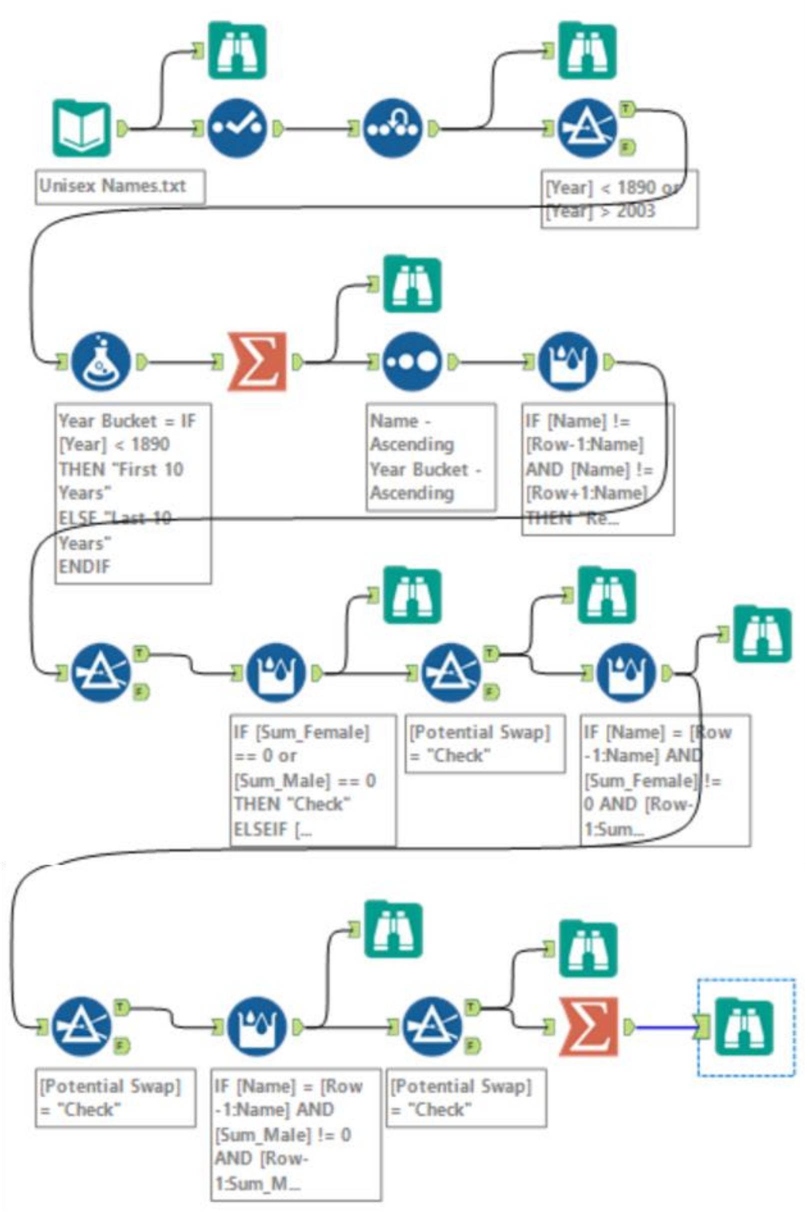


Figure-3-28-Gender Swapped Data Stream when complete

3.3 What about Me?

From

To... Alterys Consultants

Cc...

Subject FW: What About Me?

That is pretty interesting.

Now that we have taken a look at which names were completely reassigned, it would be interesting to find out if your name has a history of being unisex.

A quick look will show you that Michael doesn't even appear in the data set, so I'm going to use the name Andrea, but feel free to use your name.

Let's see what the yearly percent breakdown for male and female children was for the name you choose.

Thanks,

We will add a few things to this list to adhere to best practices, but the necessary steps we need are:

1. Import the data.
2. Clean up the fields.
3. Input a name.
4. Limit the records by that name.
5. Create calculated fields that show us the percent breakdown.
6. Export the data

We already know what issues we have in the dataset, so we can copy the last data stream until the *Imputation* tool.

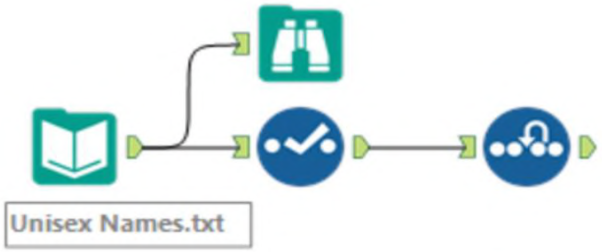


Figure-3-29-Reuse Gender Swapped for What about me

Now that we have this, the next step is to add the name to the data stream. If we add a *Text Input* with a column called *Name*, and name as the only record, below is what we will get.

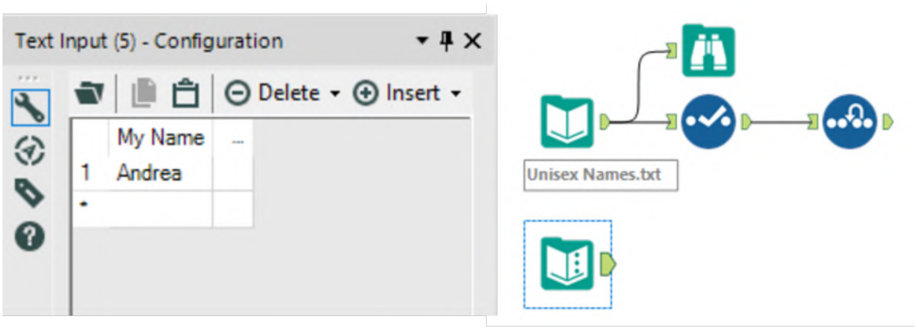


Figure-3-30-What about me Text Input

Since the *Text Input* will show us all of the data, we do not need to add a *Browse*. But it is a good practice to add the *Select* so that we can ensure that the data types from both data streams match.

Unisex Names Select **Text Input Select**

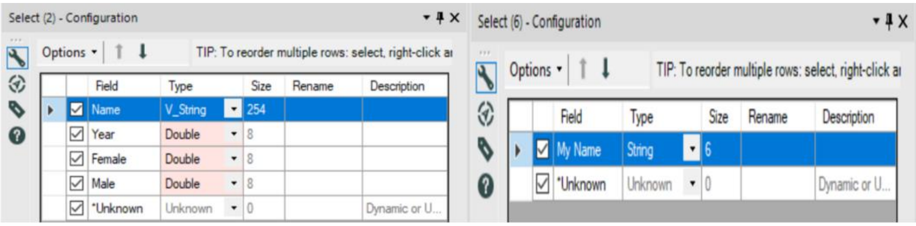


Figure-3-31-What about me Select Configuration

Though we have different types in our data field, we do not need to convert them because they are both strings. If they were not, it would cause an issue in joining the data. Due to the way we have approached this problem, we can use the *Join* tool to filter the data to the appropriate records. So, we will take only the joined section and not keep the name field coming out of the right (*Text Input*).

4 of 4 Fields | Cell Viewer

Record #	Name	Year	Female	Male
1	Andrea	1881	5	0
2	Andrea	1884	7	0
3	Andrea	1885	6	0
4	Andrea	1886	9	0
5	Andrea	1887	7	0
6	Andrea	1888	7	0
7	Andrea	1889	10	0
8	Andrea	1890	6	0
9	Andrea	1891	8	0
10	Andrea	1892	16	0

Figure-3-32-What about me Browse Configuration

Running the *Browse* tool for the *J* output will show us the following:

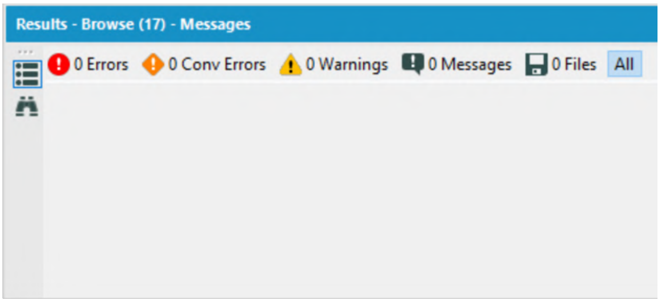


Figure-3-33-What about me browse configuration

If we see the image here, we see that no records have been returned. It is because the name entered in the *Text Input* does not match anything in the *Unisex Baby Names* dataset. Change the *Text Input* name to *Andrea* to follow along more easily.

At this point, we have covered steps 1 through 4. The next step is for us to create the percent breakdown for both male and female babies in each year. Since we intend to perform the same calculation on two different fields, we can use the *Multi-Field Formula* tool to accomplish this.

If we add the *Multi-Field Formula* tool after the *J* output from the *Join* with the following configuration, we will be able to create the percent of the total for Male and Female babies each year.

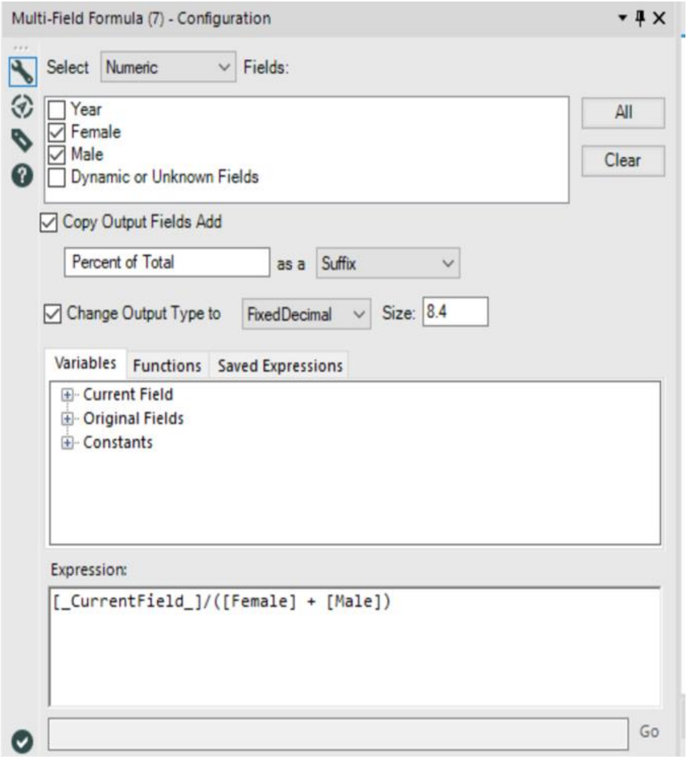
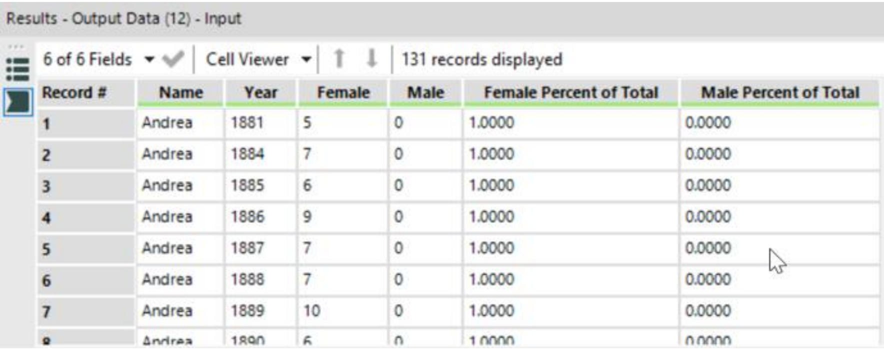


Figure-3-34-What about me Multi-Field Formula Configuration



Record #	Name	Year	Female	Male	Female Percent of Total	Male Percent of Total
1	Andrea	1881	5	0	1.0000	0.0000
2	Andrea	1884	7	0	1.0000	0.0000
3	Andrea	1885	6	0	1.0000	0.0000
4	Andrea	1886	9	0	1.0000	0.0000
5	Andrea	1887	7	0	1.0000	0.0000
6	Andrea	1888	7	0	1.0000	0.0000
7	Andrea	1889	10	0	1.0000	0.0000
8	Andrea	1890	6	0	1.0000	0.0000

Figure-3-35-What about Me-Output

Now, all that's left is to write the file out to *What About Me.csv*. The *What About Me* data stream should look like the below image when it is complete.

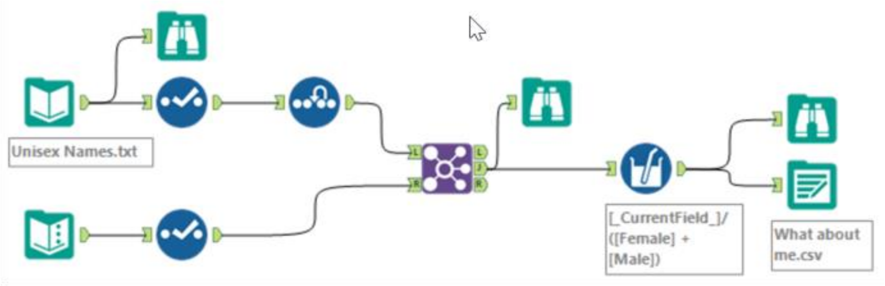


Figure-3-36 What about Me Data Stream When Complete

3.4 What's In A Name?

From

To... Alteryx Consultants

Cc...

Subject FW: What's In A Name

Hey,

So I heard back from the newspaper, and they decided to run the article. But they are curious about something that they think will be an interesting spin.

See the email below.

I don't have time to help you out this time, but I'm interested in the result. Let me know when you've got it.

Thanks,

From: Erin
To: Michael Davis
Subject: Unisex Baby Names

Hello,

What you spun up was good, but I think that we are going in a different direction, as long as the numbers work out.

For some context:

When someone tells us their name, we make assumptions about them, one of the first being their sex.

I want to know the 25 names most likely to be heard and have an incorrect assumption made about them concerning sex, based on this data.

When I say "most likely" let's call it the most amount of years when names had at least a 25% chance of referring to males and females. (If there are ties, they should be alphabetical. You can remove "Unknown" and "Baby" from the list of names).

Deadline's approaching,
Erin

CHAPTER 4

The Direct Approach

To... Alteryx Consultants

Subject What's The Policy On That?

Hey,

As I'm sure you're aware, we are working for a small company.

Frank has asked me to build a process that helps us stay on top of our new policies. I will be using a software program that, given a link, can automatically open the file or web address that it points to. But first, I need to prep the data.

I'm going to have you sit with me on this so you can see some more of Alteryx's functionalities.

They only want policies that were published in the last 30 days to show up.

Thanks.

4.1 Tools & Concepts

4.1.1 Append Fields

Concept- Appending Data/Cartesian Product

 **Figure-4-1-Append Fields**	**Tool Palette: Join**
	Adds each record from *S (source)* to the end of each record in *T (target)* providing a Cartesian product. For more details use the link below. bit.ly/2que88J

4.1.2 Auto Field

Concept- Set Data Types

 **Figure-4-2-Auto Field**	**Tool Palette: Preparation**
	Automatically sets the field to the smallest possible size and type to accommodate data in each string column. For more details use the link below. bit.ly/2GUx9eF

4.1.3 Date Time Now

Concept- Current Date/Time

	Tool Palette: In/Out

 **Figure-4-3-Date Time Now**	Gets the system time when the module starts executing. For more details use the link below. bit.ly/2EGyDDz

4.1.4 Date Time

Concept- Working with Dates

 **Figure-4-4-DateTime**	**Tool Palette: Parse**
	Converts between String and Date format fields. For more details use the link below. bit.ly/2ISpQAP

4.1.5 Directory

Concept-Working with Directories

 **Figure-4-5-Directory**	**Tool Palette: In/Out**
	Creates a data stream that has the contents of a Directory or Folder. For more details use the link below. bit.ly/2JKtFcv

4.2 What's The Policy On That?

To... Alterxy Consultants

Subject What's The Policy On That?

Hey,

As I'm sure you're aware, we are working for a small company.

Frank has asked me to build a process that helps us stay on top of our new policies. I will be using a software program that, given a link, can automatically open the file or web address that it points to. But first, I need to prep the data.

I'm going to have you sit with me on this so you can see some more of Alteryx's functionalities.

They only want policies that were published in the last 30 days to show up.

Thanks.

Since we need to search for files, we are going to be using the *Directory* tool. Our finalized policies are published as *.pdf* files, so we can use that to limit our search. We have three divisions responsible for publishing policies right now, and they all publish them to subfolders in *Chapter 4 –What's The Policy On That*. Save the folder and workflow at the same location.

The plan is to:

- Bring in a list of all of the *PDF* files in the directory.
- Get the current date.
- Tag the files to form 3 buckets.
    a. Files created in last 30 days from today (current date).
    b. Files created in between 30 to 90 days ago from today.
    c. Files created prior to 90 days from today.
- Export the data to *New Policies.xlsx*.

When we bring the *Directory* tool onto the canvas and navigate to the *Policy* folder in the *Directory* file browse, we can use the expression **.pdf* to find all *PDF* files in the directory. But since we need to search all of the subfolders (subdirectories), we need to check the box at the bottom of the configuration window.

Figure-4-6-Directory Configuration

<u>Properties Window</u>:
The *Directory Configuration* window has three components.

- *Directory* lets us navigate to a folder.
- *File Specification* allows us to type the generic format for a file name to be found in the *Directory*, where * denotes zero or more characters or spaces, while ? denotes exactly one character or space.
- *Include SubDirectories,* when checked, includes each of the subfolders when it's looking for files. Otherwise, it will only look in the directory that we have navigated to.

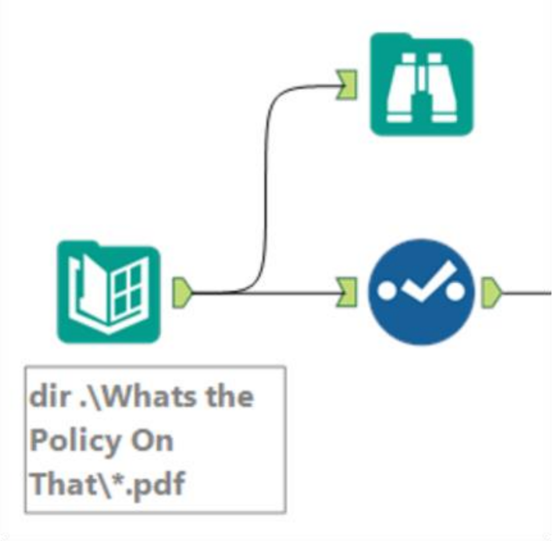


Figure-4-7-Policy Data Stream

As usual, we will add a *Browse* and S*elect* tool following the data connection. Let's look at the *Select* tool to see what's in the data stream.

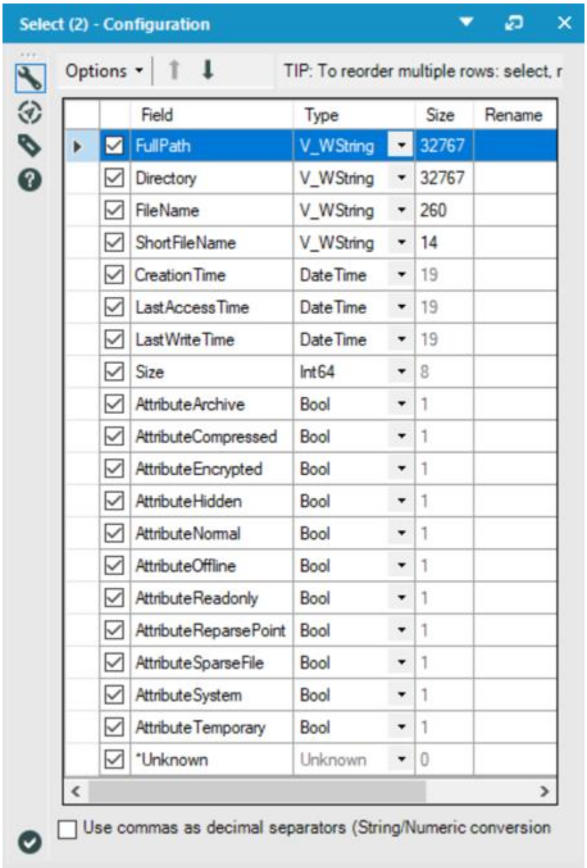


Figure-4-8-Select Configuration

We can see the data stream has 19 fields. Whenever we use the *Directory* tool, it will return these 19 metadata fields about each of the files that were found. See *Appendix K* for details on each.

In our case here, we will only be working with the *FullPath*, *Directory*, *FileName*, and *CreationTime* fields. Now that we have the appropriate information from the list of files, the next thing we need is to add the *current date* to the workflow. This is where the *Date Time Now* tool comes in. Please note here that the annotations are hidden to save space.

We will set the output format of the *Date Time* input to *MM-dd-yyyy*. It's important to note that the *DateTime* tool creates a single field with a single record, which is in the format as specified in the configuration. The value is created as String data type. We need to convert to a Date data type using the *DateTime* tool.

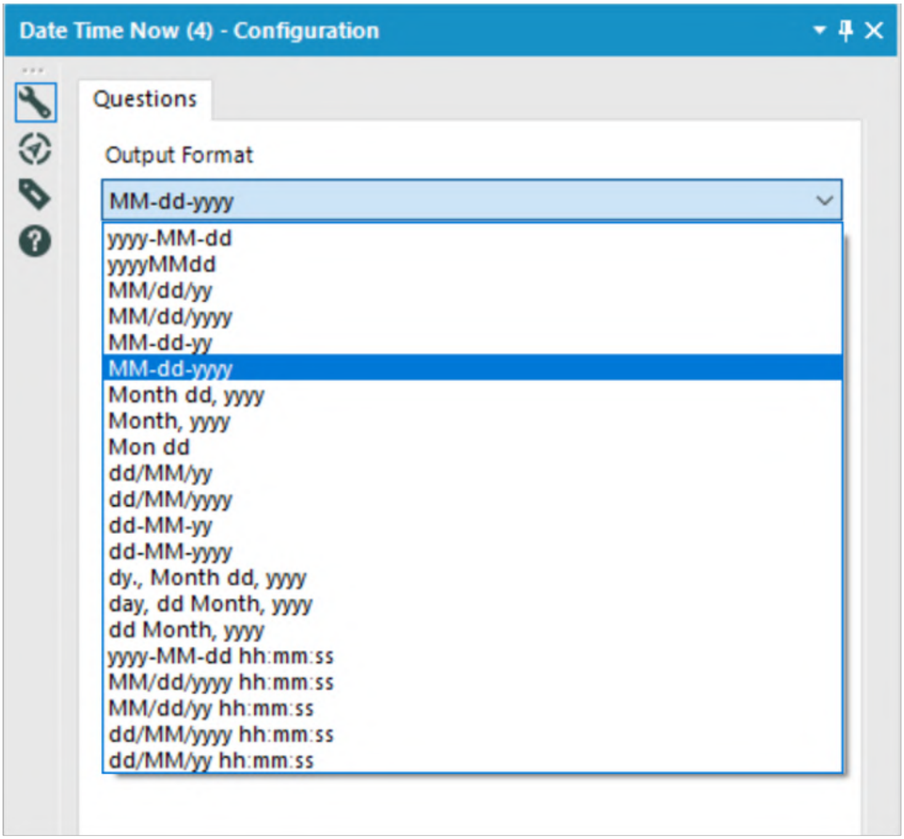


Figure-4-9-Date Time Now Configuration

<u>Properties Window</u>:
The *Date Time Now Configuration* window allows us to select the format of the string required in the output.

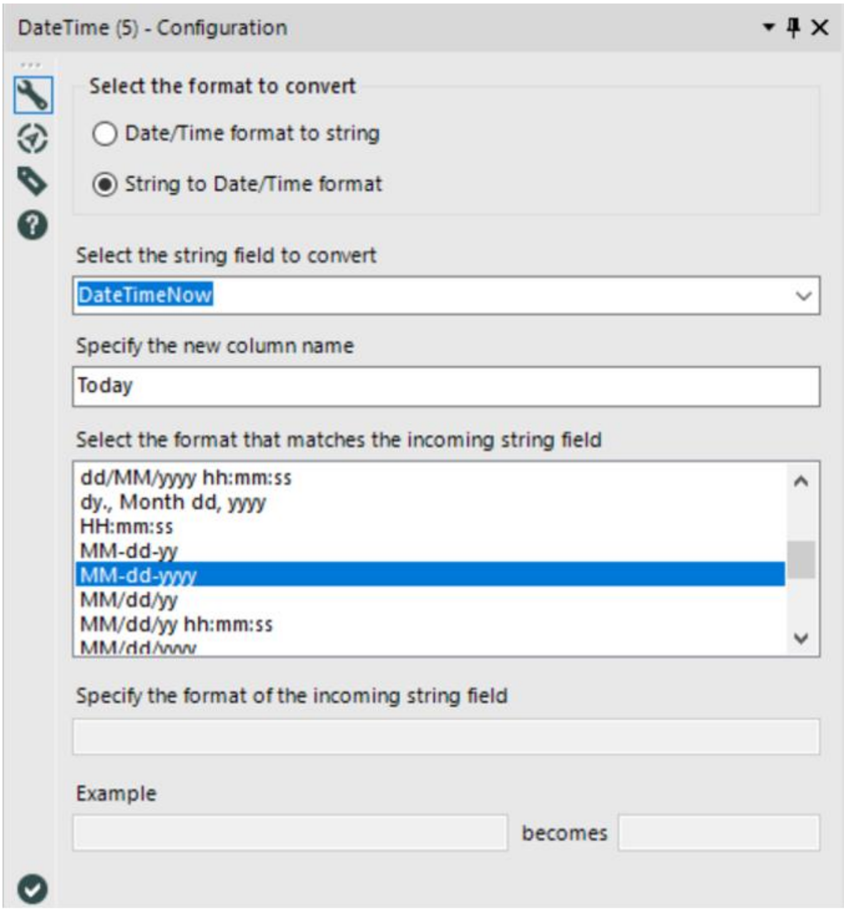


Figure-4-10-Date Time Configuration

Properties Window:
The *DateTime Configuration* window has four components.

- *Conversion Mode* allows for conversion from or to a string field.
- *Formatted Input String Field (Input Date/Time Field to be formatted)* is the field we want to convert.
- *Format of Input String (Desired Format of Output String)* is the format the input string is in.
- *Output Date/Time Field (Output Formatted String Field)* lets us name the field we are creating.

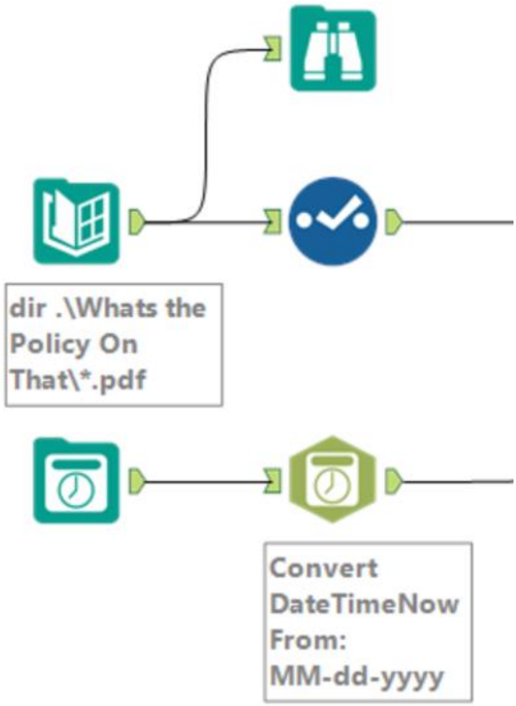


Figure-4-11-Policy Data Stream

Simultaneously, the Auto Field tool is being used to optimize the field lengths for fields which are String data type. We can use select tool as well to manually change the data type, but Auto Field is used as a best practice here.

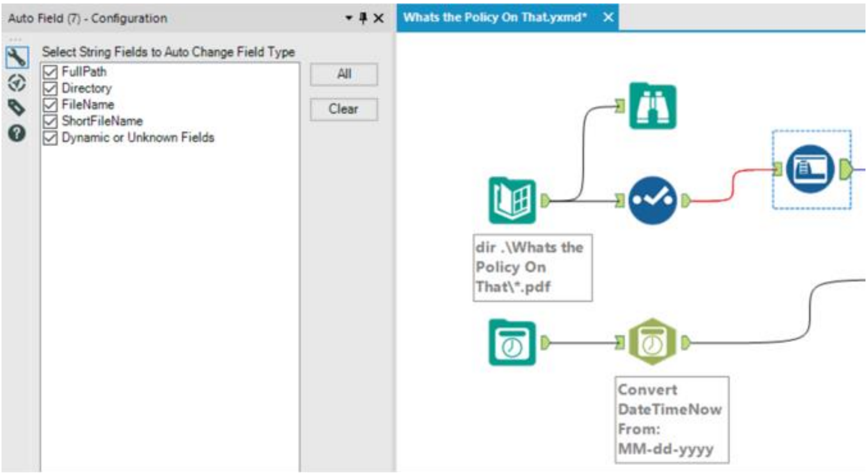


Figure-4-12-Auto Field Configuration

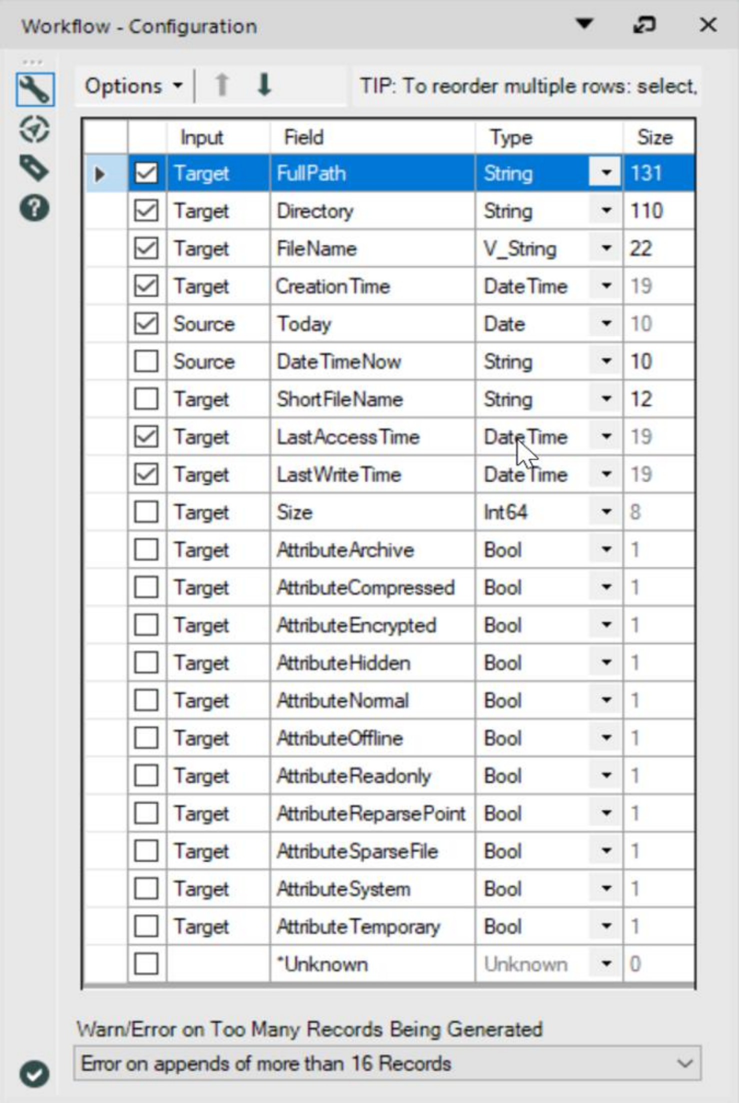
<u>Properties Window</u>: The *Auto Field configuration* window has only one setting. • *Select-String Fields to Auto Change Field Type* section allows us to select the columns for which optimization of field type and size is required.

Now that we have these two data streams ready to be combined, the question becomes: What method do we use?

A *Union* would allow us to combine the data. Unfortunately, since we will be performing a calculation to see if the dates were in the last 30 days, we need the *Today* in every record not added to the end of the dataset. A union will not work.

A *Join* would allow us to do this if we had a field we could match, which means we could use a function on both data streams that just writes 1 to the field, and then join on that. But that's difficult to explain and maintain.

Fortunately, Alteryx has a tool called *Append Fields* that does exactly what we want to do with the *Formulas* and a *Join*, in an optimized way. We shall connect the list of *PDFs* to the *Target (T)* input and the date to the *Source (S)* input. When we test with a *Browse* tool, the Today field would have been added as an extra column to every row making it a cartesian product.


Workflow - Configuration

Options TIP: To reorder multiple rows: select,

	Input	Field	Type	Size
☑	Target	FullPath	String	131
☑	Target	Directory	String	110
☑	Target	FileName	V_String	22
☑	Target	CreationTime	DateTime	19
☑	Source	Today	Date	10
☐	Source	DateTimeNow	String	10
☐	Target	ShortFileName	String	12
☑	Target	LastAccessTime	DateTime	19
☑	Target	LastWriteTime	DateTime	19
☐	Target	Size	Int64	8
☐	Target	AttributeArchive	Bool	1
☐	Target	AttributeCompressed	Bool	1
☐	Target	AttributeEncrypted	Bool	1
☐	Target	AttributeHidden	Bool	1
☐	Target	AttributeNormal	Bool	1
☐	Target	AttributeOffline	Bool	1
☐	Target	AttributeReadonly	Bool	1
☐	Target	AttributeReparsePoint	Bool	1
☐	Target	AttributeSparseFile	Bool	1
☐	Target	AttributeSystem	Bool	1
☐	Target	AttributeTemporary	Bool	1
☐		*Unknown	Unknown	0

Warn/Error on Too Many Records Being Generated

Error on appends of more than 16 Records

Figure-4-13-Append Field Configuration

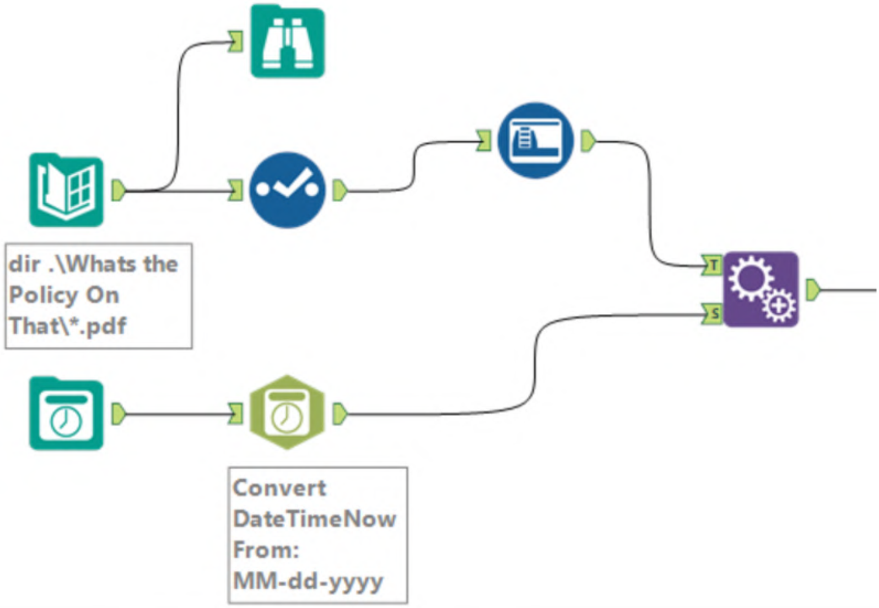


Figure-4-14-Policy Data stream

<u>Properties Window:</u>
The *Append Fields Configuration* window is identical to the *Select Configuration* window, with two exceptions.

- There is an additional element in the metadata section called *Input*. It identifies if the data is coming from the *Target (T)* or *Source (S)* inputs for the tool.
- There is a *Warn/Error on Too Many Records Being Generated*, which allows us to decide if and how we should be alerted to a high rate of replication of the *Target* field.

In order to create buckets, we need a *Formula* tool to create a new field called *Bucket*. This is done using the formula.

```
IF [CreationTime] > DateTimeAdd([Today], -30, "days")
THEN 1 ELSEIF
[CreationTime] <= DateTimeAdd([Today], -30, "days")
and
[CreationTime] > DateTimeAdd([Today], -90, "days")
THEN 2 ELSE 3 ENDIF
```

The *Bucket* will logically divide the data based on provided conditions.

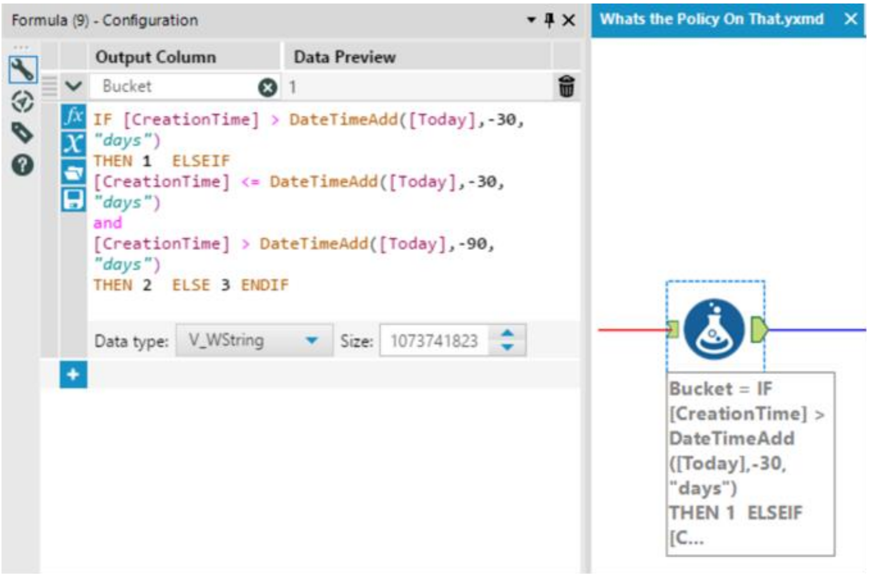


Figure-4-15-Formula Configuration

Then we use a *Filter* tool to restrict that data with Bucket value equal to 1.

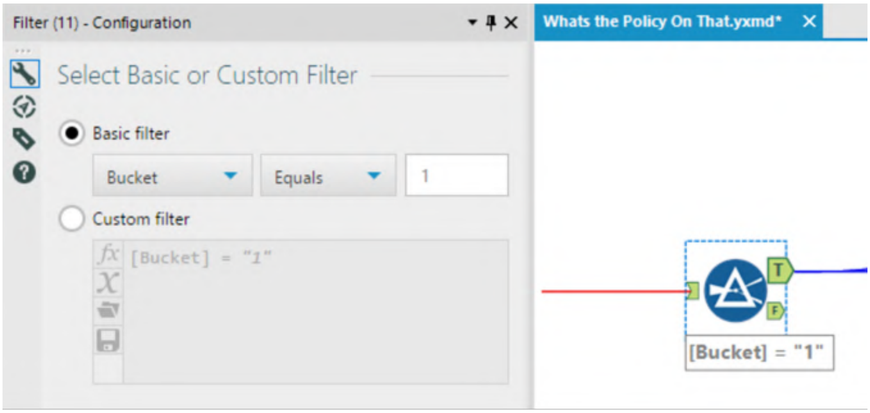


Figure-4-16-Filter Configuration

The last thing is to output the data to a file called *New Policies.xlsx* with a new sheet. The process is set up such a way that every time the workflow is executed, it deletes the contents of the existing sheet and overwrite with new values for the current date (based on the system time).

Make sure that the workflow is saved in the same location where *What's The Policy On That* folder is saved.

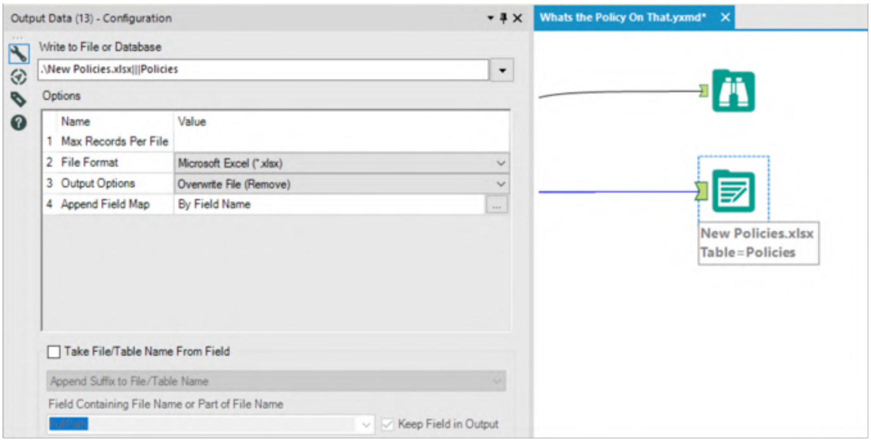


Figure-4-17-Output Data Configuration

The *What's The Policy On That?* data stream looks like the one below when it's complete.

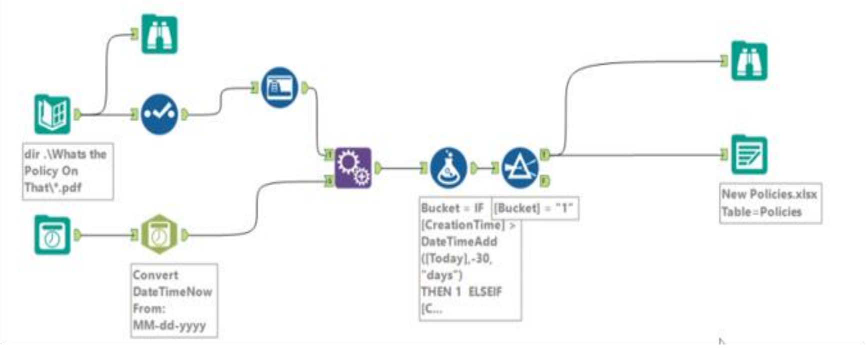


Figure-4-18-Policy Data Stream After Completion

4.3 Where, Oh Where Have the Three Files Gone?

To... Alterxy Consultants

Subject Where, Oh Where Have The Three Files Gone?

Hey,

I know this is a little unorthodox, but it will be a good way to score points with me and my boss.

Usually, I would take care of things like this, but Frank needs me in a meeting.

Frank has lost some important files, and they don't remember what the files are called.

Frank knows they saved the files somewhere on their network drive under a particular folder each, and that you will recognize the names of the files as soon as you find them, but searching the network drive is something Frank doesn't have time to do.

I need you to build a directory and sub directory search for an Excel file (.xlsx) in the "Excel Files" folder, a PDF file (.pdf) in the "PDF files" folder, and an image file (.png) in the "Images" folder in "Chapter 4 – The Direct Approach" > "Looking for Files." Look at the results of each of the three independent file lists, and identify the ones we need.

After that, bring the data together into an output file with the two columns "File Name" and "Full Path", so that by the time the meeting is over, we can direct Frank right to the files.

I really appreciate this.

CHAPTER 5
Cultural Musing

To... Alteryx Consultants

Subject Culturally Rich

Attached 2010 GDP Per Capita Per State.xlsx; Museum Data.csv; State Map.xlsx

Hey,

We have some downtime, and I would like to explore the idea that the richer the state, the more likely it is to have museums.

We are going to use the attached list of 2014 Q3 Museums data, 2010 GDP data, and the 2010 Census data.

If you don't have the 2010 Census data package yet, you can download it at http://downloads.alteryx.com/data.html. Just walk through the installer, and when you see the uninstall screen, make sure nothing is selected.

Let's walk through the data prep.

Thanks.

5.1 Tools & Concepts

5.1.1 Allocate Input

Concept - Using Census Data

 **Figure-5-1-Allocate Input**	**Tool Palette: Demographic Analysis**
	Gets input from demographic data using specific packages. For more details, use the link below. bit.ly/2qVh1in

5.1.2 Find and Replace

Concept - Data Mapping

 **Figure-5-2-Find and Replace**	**Tool Palette: Join**
	Replaces information in a data stream by entering the *F* (find) (target) input with information that matches in the *R* (replace) (source) input. For more details, use the link below. bit.ly/2HIEa24

5.1.3 Join Multiple
Concept – Complex Joins

 **Figure-5-3-Join Multiple**	**Tool Palette: Join**
	Performs a full outer join or an inner join between data streams sharing a set of key fields. For more details, use the link below. bit.ly/2HUoOVG

5.1.4 Text To Columns
Concept – Splitting field members at delimiters

 **Figure-5-4-Text To Columns**	**Tool Palette: Parse**
	Breaks string fields into multiple string fields based on a delimiter. For more details, use the link below. bit.ly/2qTTrTN

5.1.5 Random % Sample

Concept – Generate a random number or % of record

 **Figure-5-6-Random% Sample**	**Tool Palette: Preparation**
	Returns an expected number of records resulting in a random sample of the incoming data stream. For more details, use the link below. bit.ly/2HoB3ZA

5.1.6 XML Parse

Concept – Read and Parse the XML snippet

 **Figure-5-6-XML Parse**	**Tool Palette: Parse**
	Reads in XML snippets and parses them into individual fields. For more details, use the link below. bit.ly/2HWpEBi

5.2 Culturally Rich

To... Alteryx Consultants

Subject Culturally Rich

Attached 2010 GDP Per Capita Per State.xlsx; Museum Data.csv; State Map.xlsx

Hey,

We have some downtime, and I would like to explore the idea that the richer the state, the more likely it is to have museums.

We are going to use the attached list of 2014 Q3 Museums data, 2010 GDP data, and the 2010 Census data.

If you don't have the 2010 Census data package yet, you can download it at http://downloads.alteryx.com/data.html. Just walk through the installer, and when you see the uninstall screen, make sure nothing is selected.

Let's walk through the data prep.

Thanks.

The files required for this chapter have been placed in clearly marked sub-folders in *Chapter 5 – Culturally Rich*. We will start by bringing in each of the four data sources to see what we have. Let us open all three files and then connect to the *Census* data.

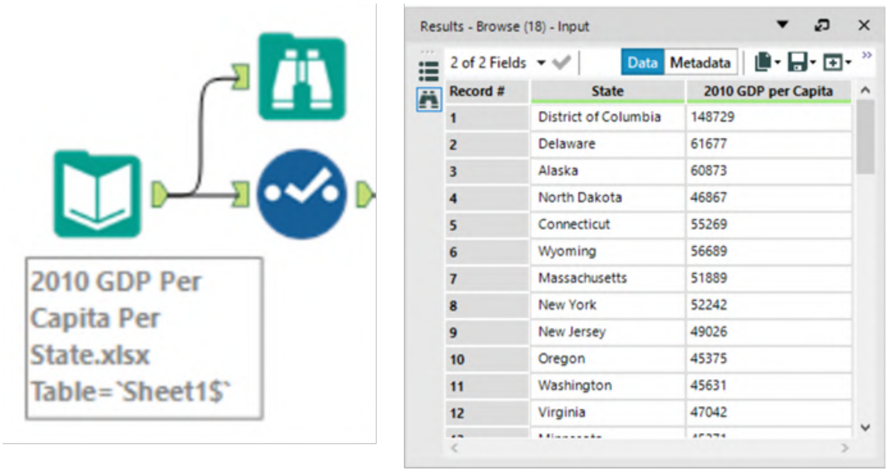


Figure-5-7 – Culturally Rich – GDP Data

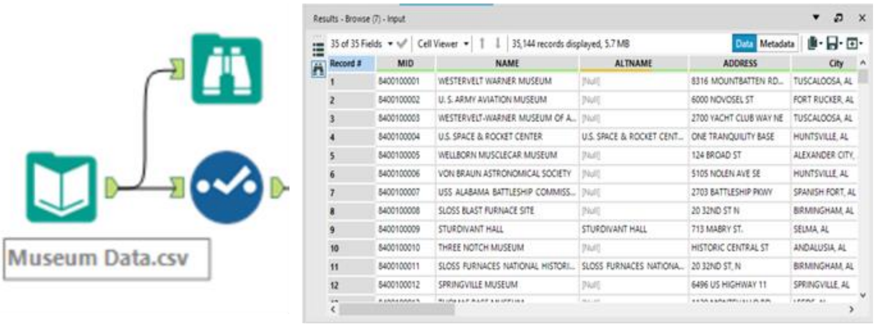
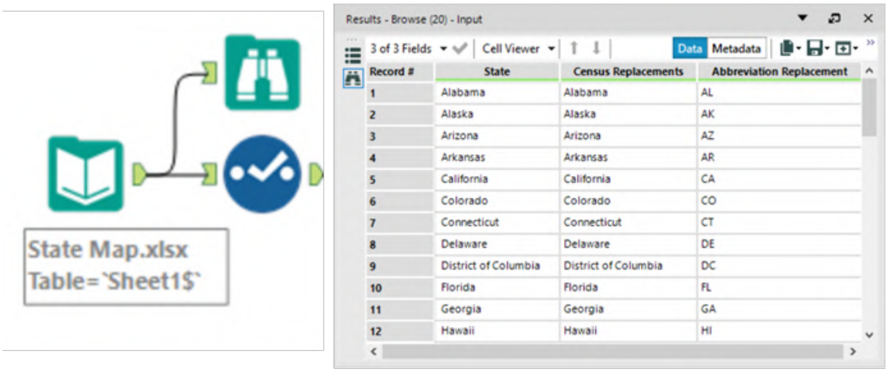


Figure-5-8 – Culturally Rich – Museum Data

In these two files, the names of the states appear in different formats, with the *GDP per Capita* file having the entire state name spelled out, and the *Museum* data having only two-letter state abbreviation within the city field.



Record #	State	Census Replacements	Abbreviation Replacement
1	Alabama	Alabama	AL
2	Alaska	Alaska	AK
3	Arizona	Arizona	AZ
4	Arkansas	Arkansas	AR
5	California	California	CA
6	Colorado	Colorado	CO
7	Connecticut	Connecticut	CT
8	Delaware	Delaware	DE
9	District of Columbia	District of Columbia	DC
10	Florida	Florida	FL
11	Georgia	Georgia	GA
12	Hawaii	Hawaii	HI

Figure-5-9- Culturally Rich – State Map

However, the third file has three separate columns with either the complete state name or its abbreviation. This is because *State Map.xlsx* is a file created especially for the purpose of field mapping by linking the data sources from the *GDP*, *Museum* and *Census* data sets, which all have state identifiers in different formats.

The following are the observations from the data in the three files we connected to.

- The *GDP per Capita* data does not need any preparation before the join.
- The *State* associated with each museum needs to be parsed out of *City* and then mapped to the *GDP per Capita* name.
- The *State Map.xlsx* file can be used for mapping all data sources together.

Let us now bring in the *Census* data in order to plan what needs to be done with that data stream.

We shall use an *Allocate Input* tool on the canvas, then choose the *US Census 2010 – Most Recent Vintage* dataset. Let us select the *States* option under *Pick Geography* and click the checkbox to select all states.

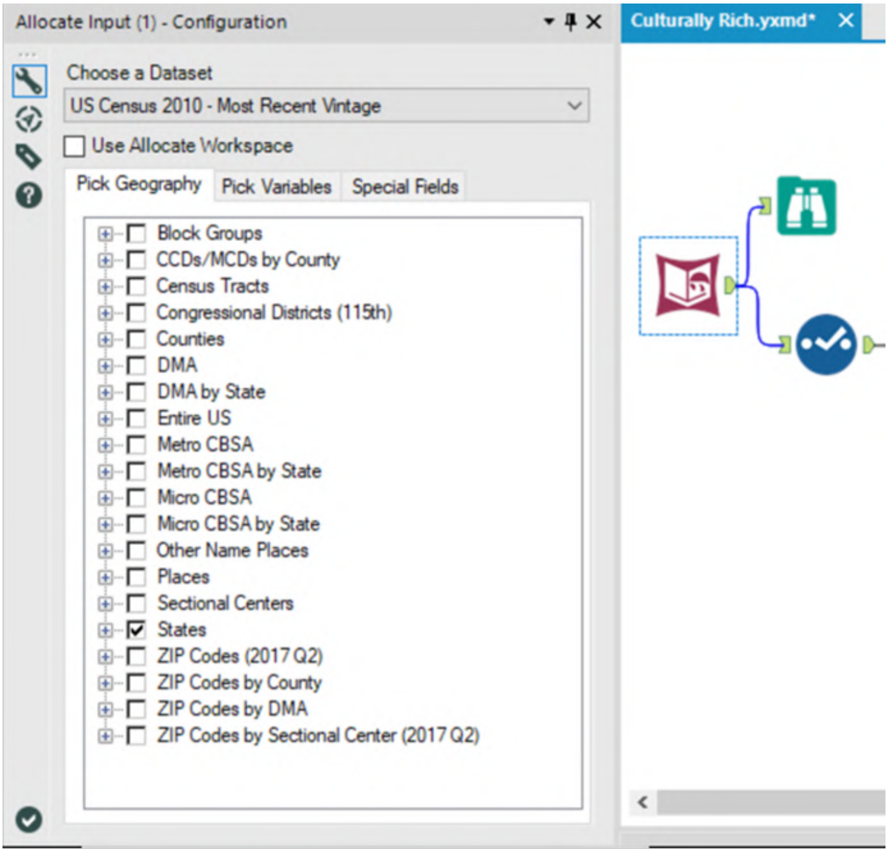


Figure-5-10 – Culturally Rich
Allocate Input Configuration – Pick Geography

The purpose of bringing in the *Census* data is to compare the *GDP* to the number of museums in each state instead of the *GDP per Capita.* We can use the two data sources to generate the GDP value.

Now that the goal is identified, figuring out the variable needed becomes easy.

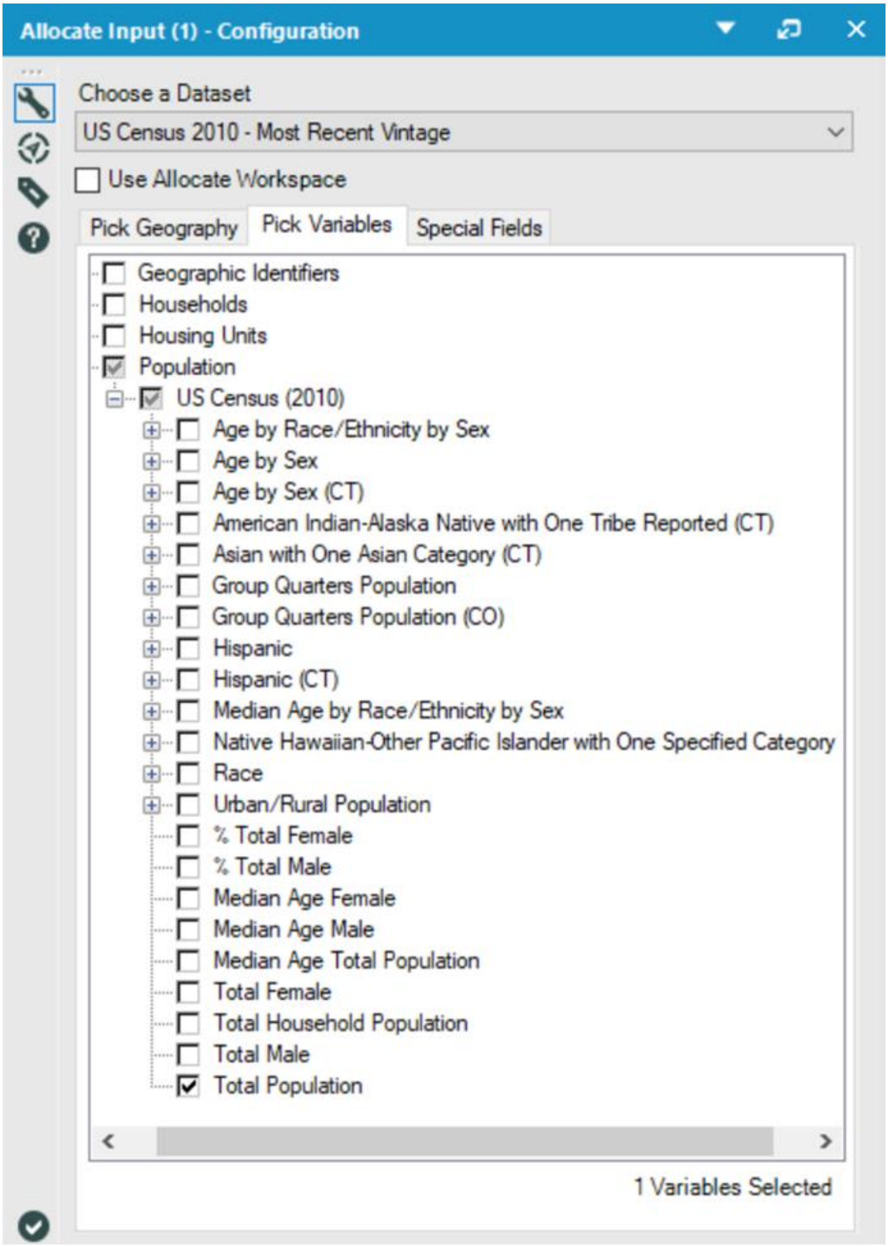


Figure-5-11 – Culturally Rich
Allocate Input Configuration – Pick Variables

Under *Population* and *US Census (2010)*, the last option is *Total Population*. This signifies the total number of people that have lived in each chosen State (Geography).

Since we need only this information, click on the *Special Fields* (as shown in the image below) and uncheck all options.

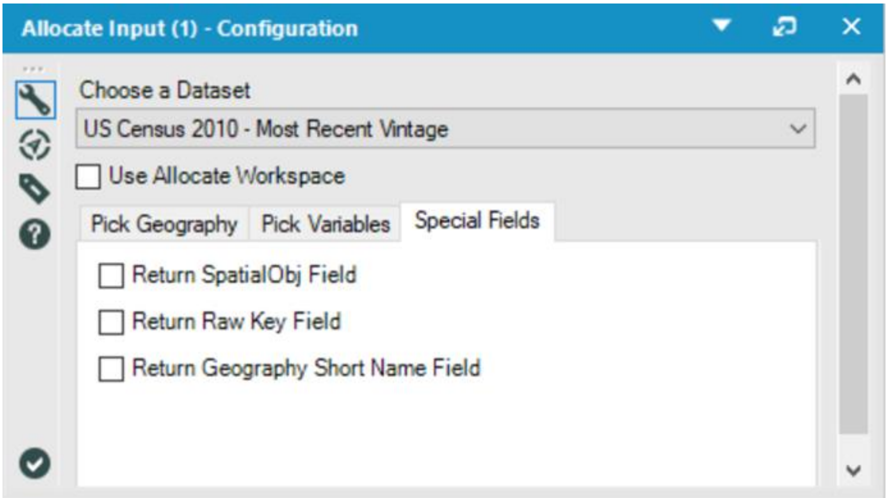
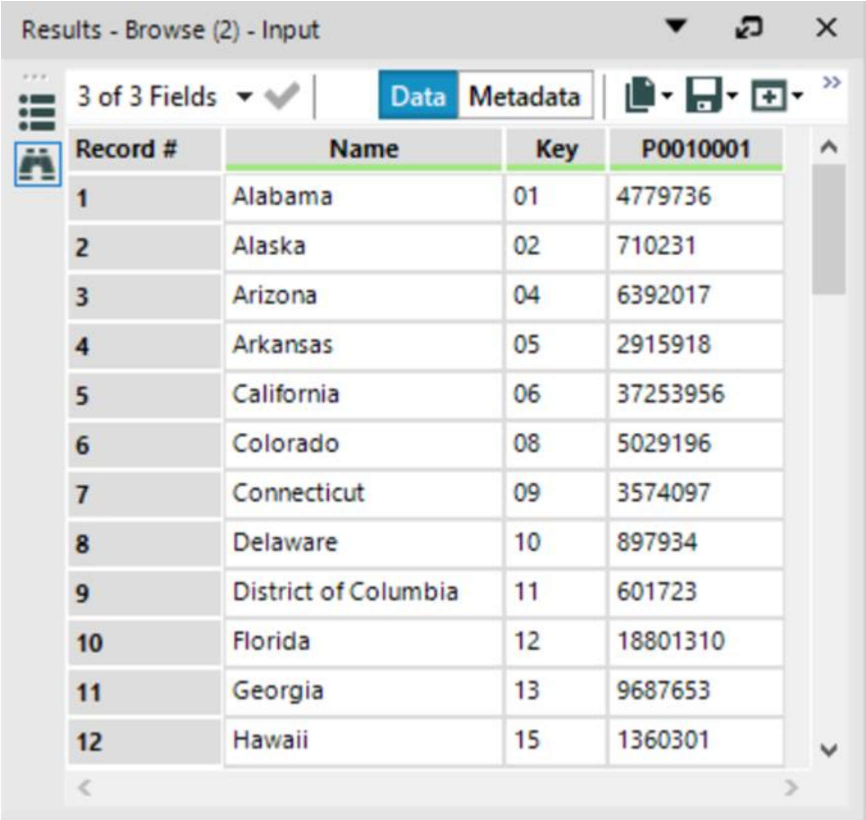


Figure-5-12 – Culturally Rich
Allocate Input Configuration – Special Fields

Now that we have started with the final data stream, let us take a look at the data that comes out of the *Census* data connection.



Record #	Name	Key	P0010001
1	Alabama	01	4779736
2	Alaska	02	710231
3	Arizona	04	6392017
4	Arkansas	05	2915918
5	California	06	37253956
6	Colorado	08	5029196
7	Connecticut	09	3574097
8	Delaware	10	897934
9	District of Columbia	11	601723
10	Florida	12	18801310
11	Georgia	13	9687653
12	Hawaii	15	1360301

Figure-5-13 – Culturally Rich Allocate Input Browse Configuration

The following two points about this data stand out:

- Even though we have only selected one geography and one variable, we still have three fields. This is because the *Key* field uniquely identifies all geographies so that even if we have regions with the same name, they have unique identifiers. In this case, because they are States, we do not need the *Key*.
- There is a field called *P0010001*. This is because the data is stored with keyed column headers. We will simply rename the field to *2010 Population* in the *Select* tool, as shown in the following image.

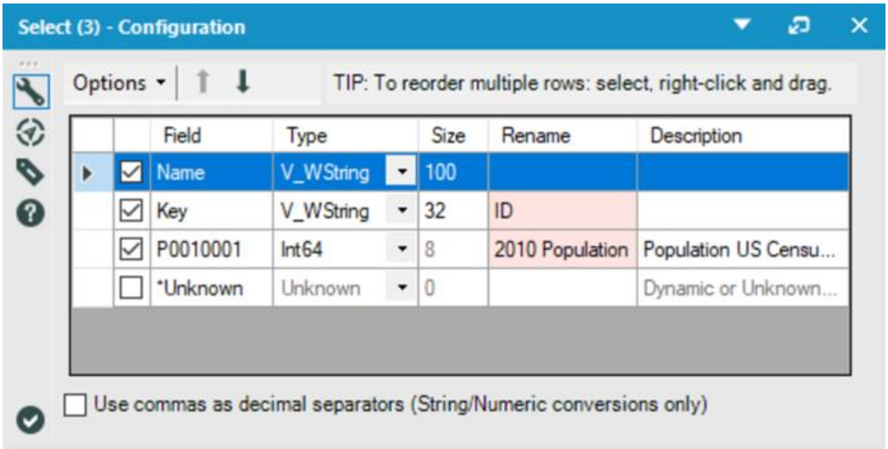


Figure-5-14 – Culturally Rich Select Configuration

Now, the remaining preparation is to map the state names from the data sources back to the *GDP per Capita State* names before joining these data streams.

Let us start with mapping the *Census* data since it requires only one step before the join. In order to do this, we will use the *Find Replace* tool. Connect *Census* data stream to the *Find (F)* input and the mapping data stream to the *Replace (R)* input as shown in the following image.

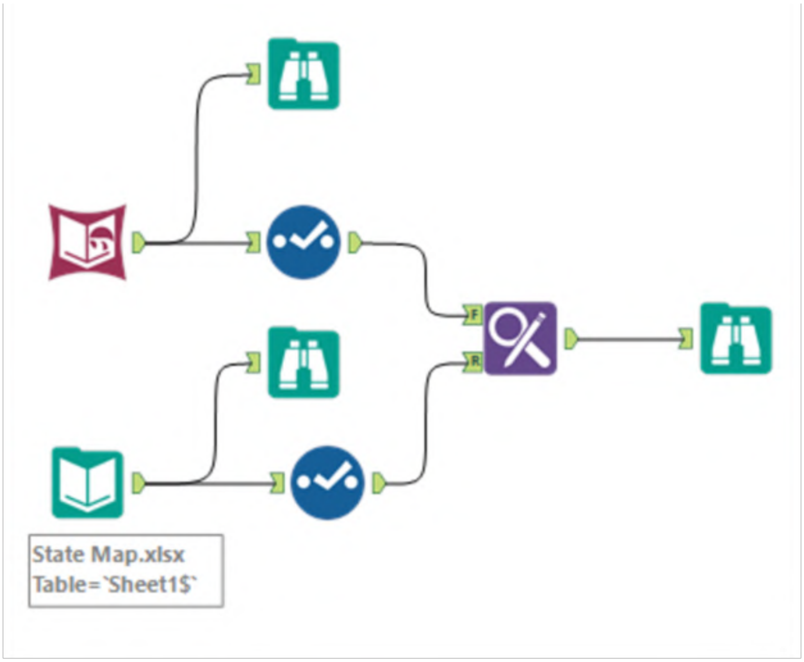


Figure-5-15 – Culturally Rich
Connecting census data to Find and Status Map data to Replace

Now, let us take a look at the settings of *Find Replace* tool.

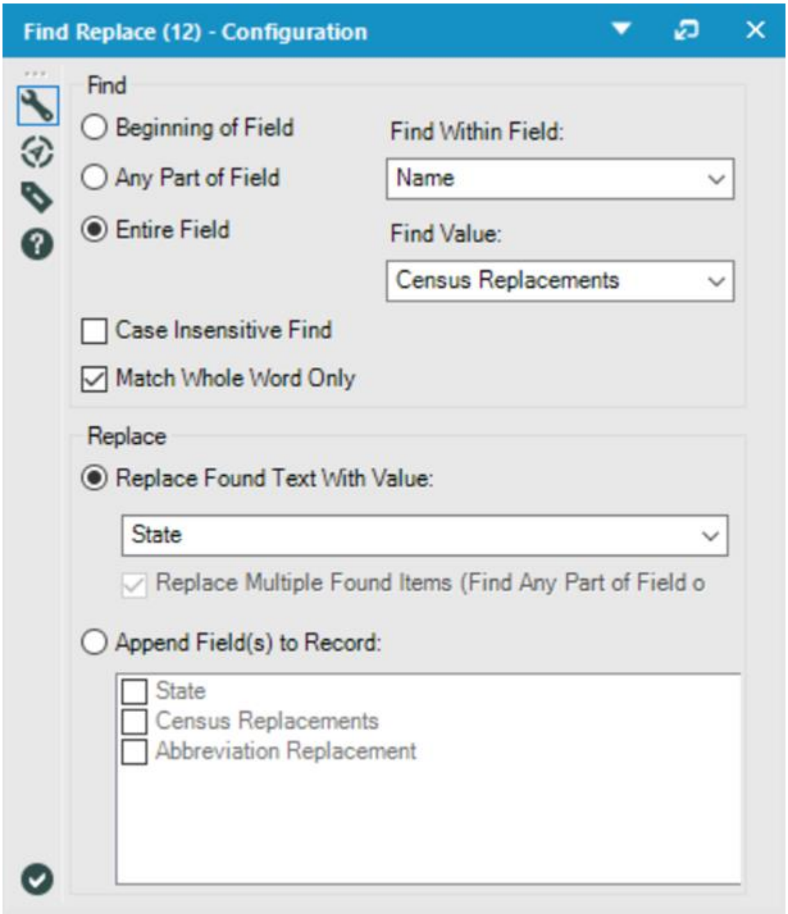


Figure-5-16 – Culturally Rich Census first Find Replace Configuration

Since the map has been structured in such a way that the field *Census Replacements* has all of the *Census State* names in it, we will be looking for the *Entire Field* matches of the *Census Replacements* in the *Census Name* field. We also want to make sure that we replace the fields found in the *State* field (which has the *State* names from the *GDP* data). Since the *Census* data stream and the *GDP* data stream share a mapping field, let us get the *Museums* data ready.

We will first remove all of the fields except *Name* and *City* in the *Select* tool following the *Input Data* tool. Then, we will use Random % Sample tool with following settings to randomly retrieve 50% *Museums* data.

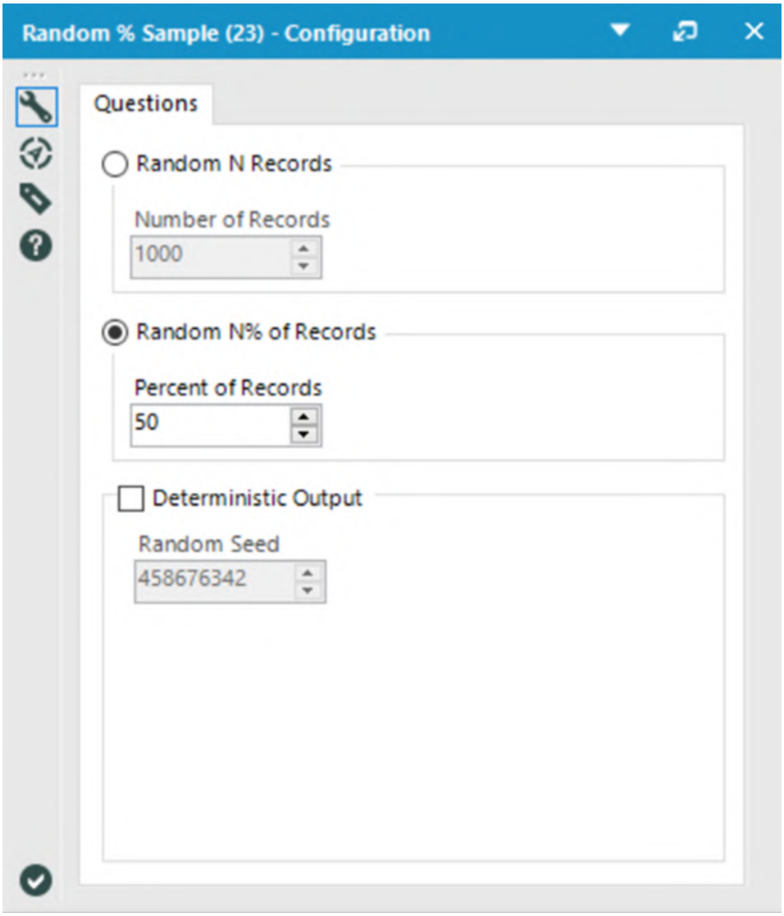


Figure-5-17 – Culturally Rich Random % Sample Configuration

Next, we will split the data. To do this, use the *Text to Columns* tool and add it to the end of the *Museums* data stream with the following settings.

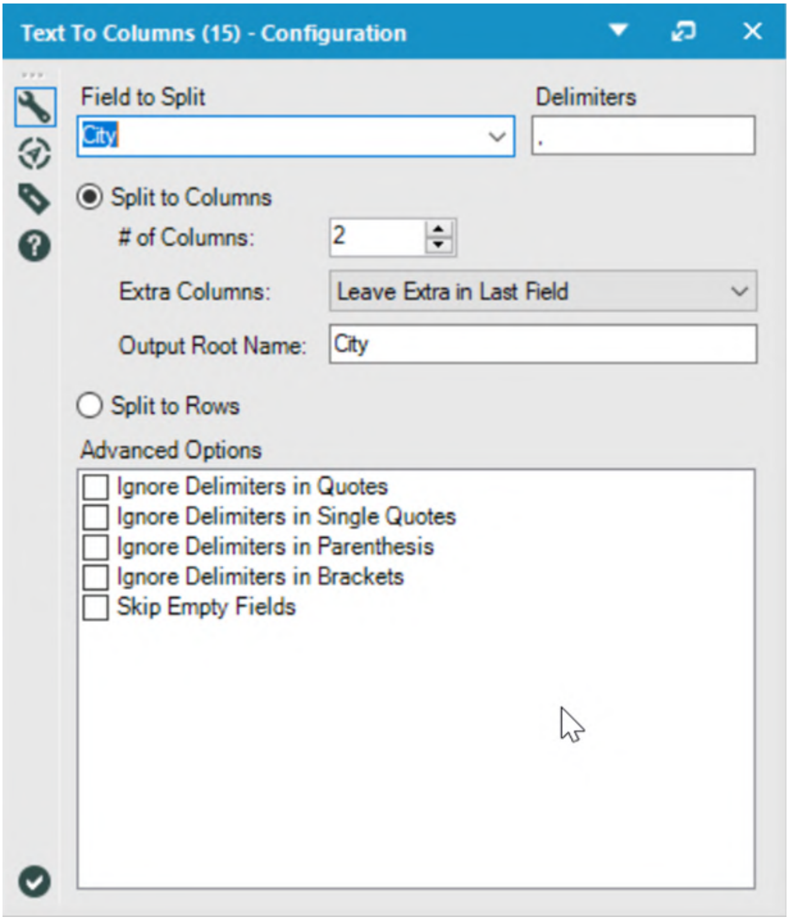
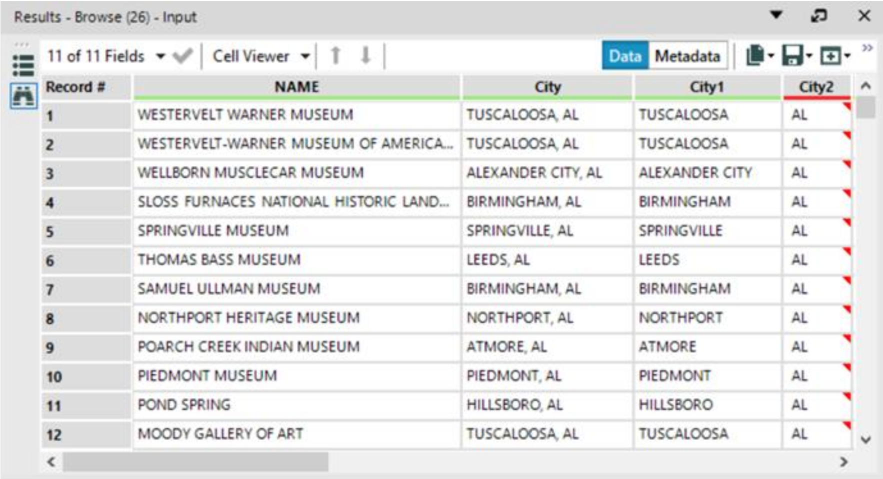


Figure-5-18 – Culturally Rich
Text to Columns Configuration to split at a separator

Then, let us look at the updated data after it has been modified with the *Text to Columns* field.



Record #	NAME	City	City1	City2
1	WESTERVELT WARNER MUSEUM	TUSCALOOSA, AL	TUSCALOOSA	AL
2	WESTERVELT-WARNER MUSEUM OF AMERICA...	TUSCALOOSA, AL	TUSCALOOSA	AL
3	WELLBORN MUSCLECAR MUSEUM	ALEXANDER CITY, AL	ALEXANDER CITY	AL
4	SLOSS FURNACES NATIONAL HISTORIC LAND...	BIRMINGHAM, AL	BIRMINGHAM	AL
5	SPRINGVILLE MUSEUM	SPRINGVILLE, AL	SPRINGVILLE	AL
6	THOMAS BASS MUSEUM	LEEDS, AL	LEEDS	AL
7	SAMUEL ULLMAN MUSEUM	BIRMINGHAM, AL	BIRMINGHAM	AL
8	NORTHPORT HERITAGE MUSEUM	NORTHPORT, AL	NORTHPORT	AL
9	POARCH CREEK INDIAN MUSEUM	ATMORE, AL	ATMORE	AL
10	PIEDMONT MUSEUM	PIEDMONT, AL	PIEDMONT	AL
11	POND SPRING	HILLSBORO, AL	HILLSBORO	AL
12	MOODY GALLERY OF ART	TUSCALOOSA, AL	TUSCALOOSA	AL

Figure-5-19 – Culturally Rich Modified Museum Data

When we look at the *Browse* tool, we can see that *City2* has a *red triangle* at the top-right corner of the cell. It is an alert indicating that Alteryx recognizes a potential issue with the data. If we hover the mouse over the cell, we will see a little textbox pop-up that reads “This cell has leading spaces”. This means the first character of the cell is a space (attributed to the split and a spacer after “,” delimiter) and that we should address it. We will do this by using the *Formula* tool to create a new field, we will call *State*.

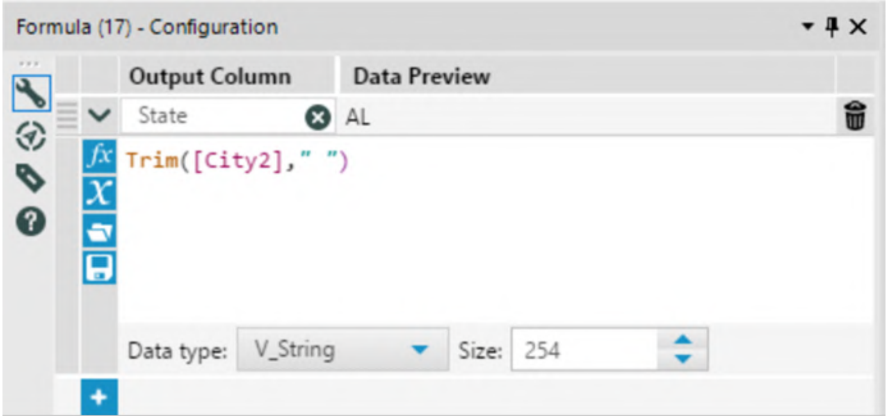


Figure-5-20- Culturally Rich Museum data using Formula

This time, we will be creating a string. If we look at the below *String* functions, we can find three functions designed to remove one string from another. We will use the *Trim* function because we do not know if there are cells having trailing spaces which would cause issues with mapping.

Since we are removing spaces from the field, we could simply write the formula *Trim([City2])* because the function removes white spaces by default. However, for the purpose of clarity and readability, the best practice is to explicitly define all the variables. In this case, it would mean writing the formula as *Trim([City2]," ")*.

Then, add a Select tool to rename *City1* as *City* and uncheck the *City2* field to remove it from the data stream.

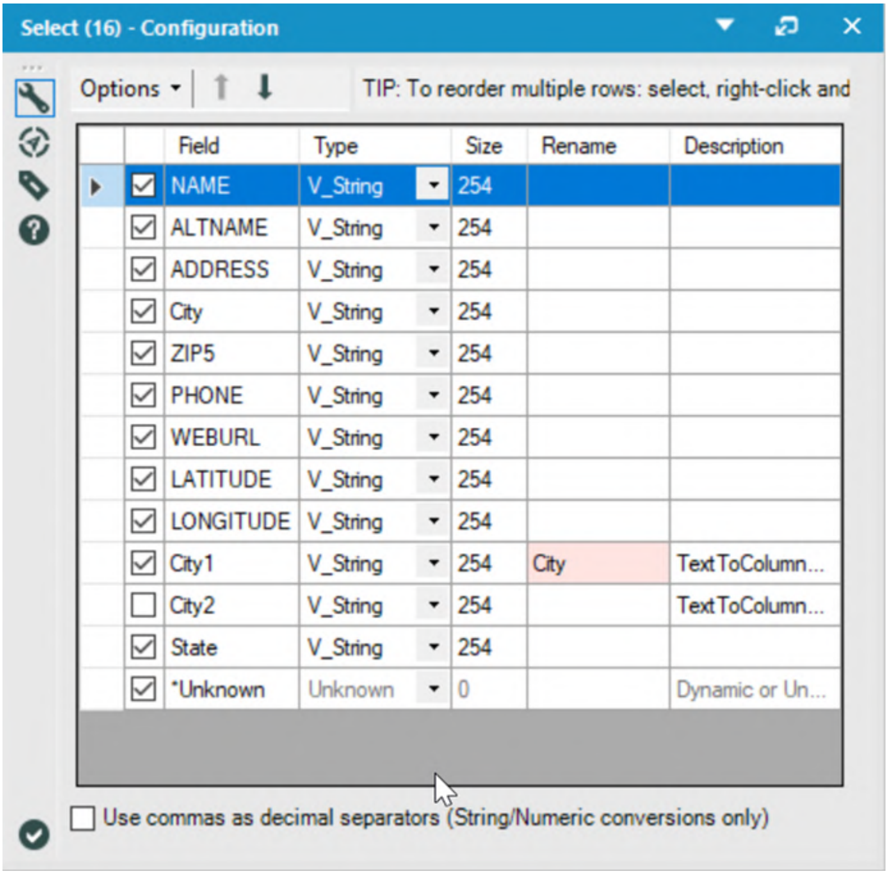


Figure-5-21- Culturally Rich Museum Data Select Configuration after Formula Tool

Now that we are left with clean two-character *State* abbreviations, we can map them to the *GDP State* names. Once again, we will bring a *Find Replace* tool onto the canvas and connect the mapping data to the *Replace* input.

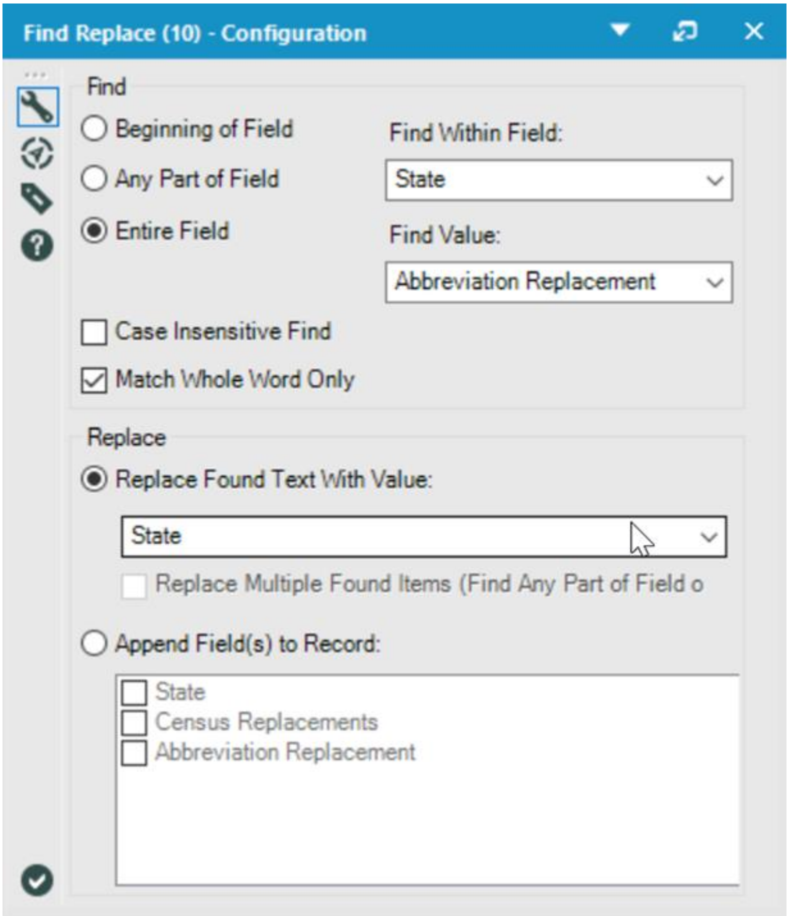


Figure-5-22- Culturally Rich Second Find and Replace Configuration

Now that we have prepared all the contributing data streams, this is what the data stream should look like.

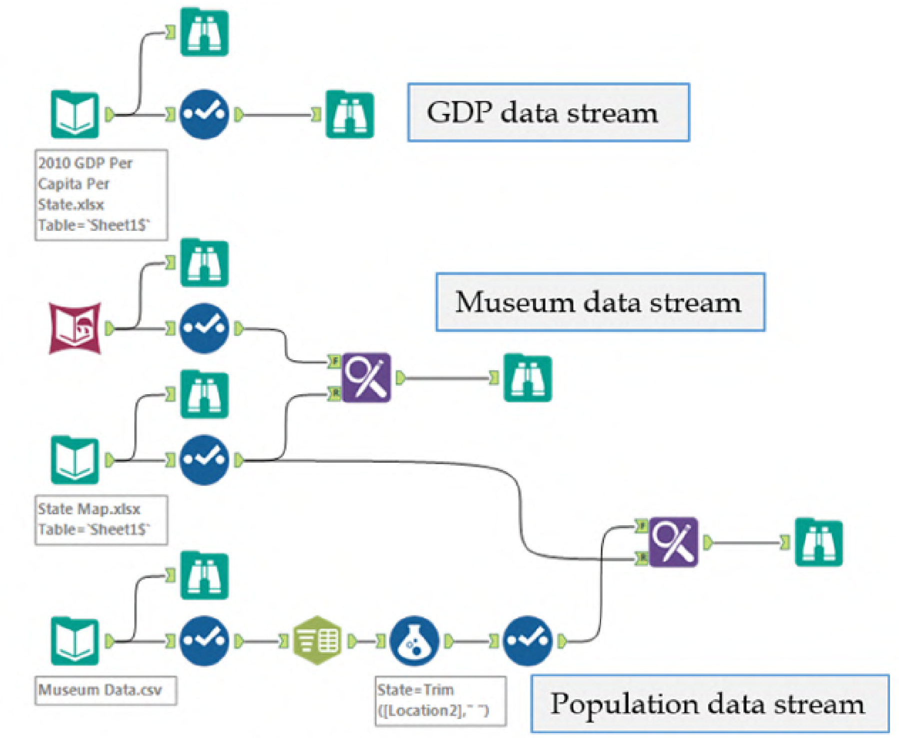


Figure 5-25 – Culturally Rich Consolidated Modified data streams

Notice that we have crossed the data streams. From a technical perspective, it does not matter if they are crossed. However, practically, the workflow becomes harder to read when the data streams are crossed frequently.

Best practice is to create workflows with as few crossovers as possible. In this case, we leave it as-is because there will invariably be at least one cross over in this module.

We could use a couple of *Join* tools now, but since all data streams share the same key (*State* name), we can use a *Join Multiple* tool to join all three data streams at once. We will connect the *GDP Per Capita*, *Museum*, and *Population* data streams to a *Join Multiple* inputs in the same order. We can then rename the connectors similarly and configure it as shown in the image.

One thing to note is that Alteryx renames conflicting fields with the data stream name and an "_". For example, since there are two fields that are named *State*, the second occurrence is renamed with the data stream as an identifier: *Museum_State*.

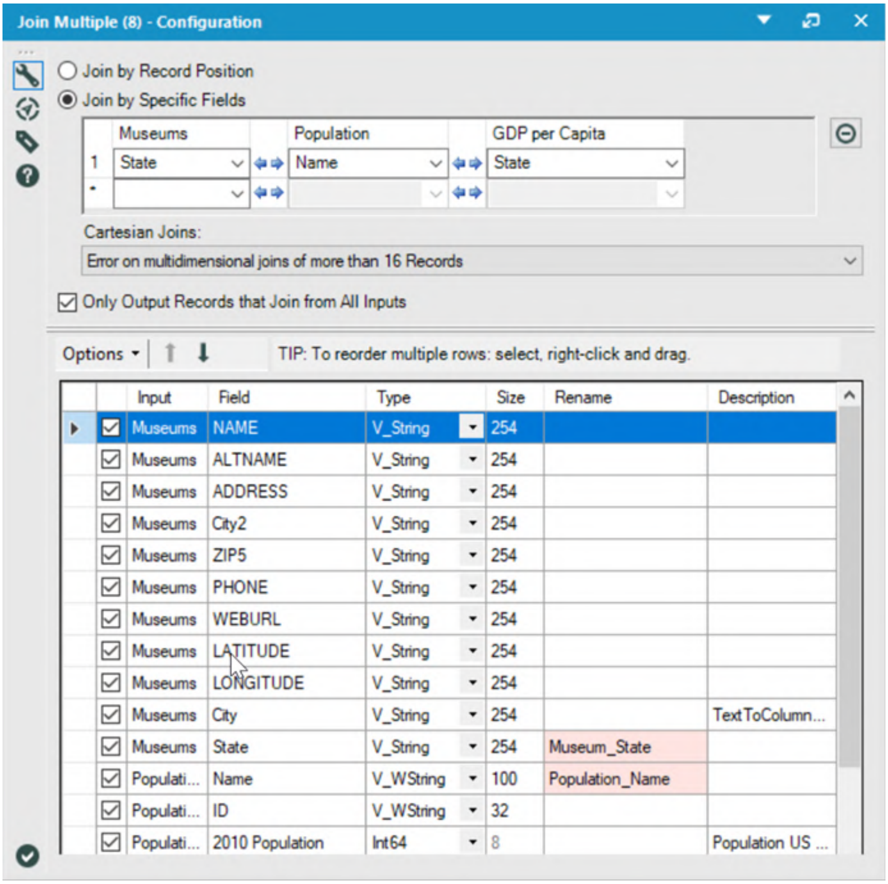
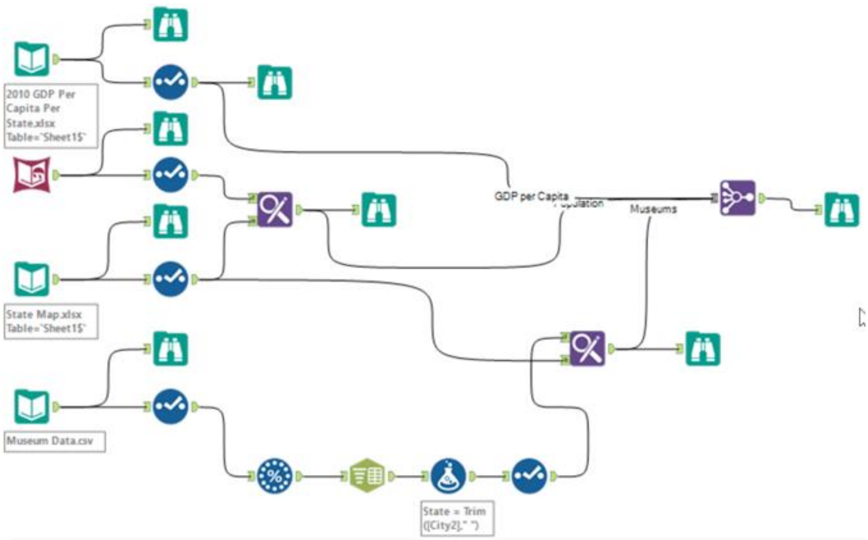


Figure-5-24- Culturally Rich Second Join Multiple Configuration

This is what the data stream would look like.



Fiigure-5-25 – Culturally Rich Unified Data Stream

Once we perform the following steps, our data preparation will be complete.

- Add a field called *Total GDP* (product of GDP Per Capita and Population) using a Formula tool.

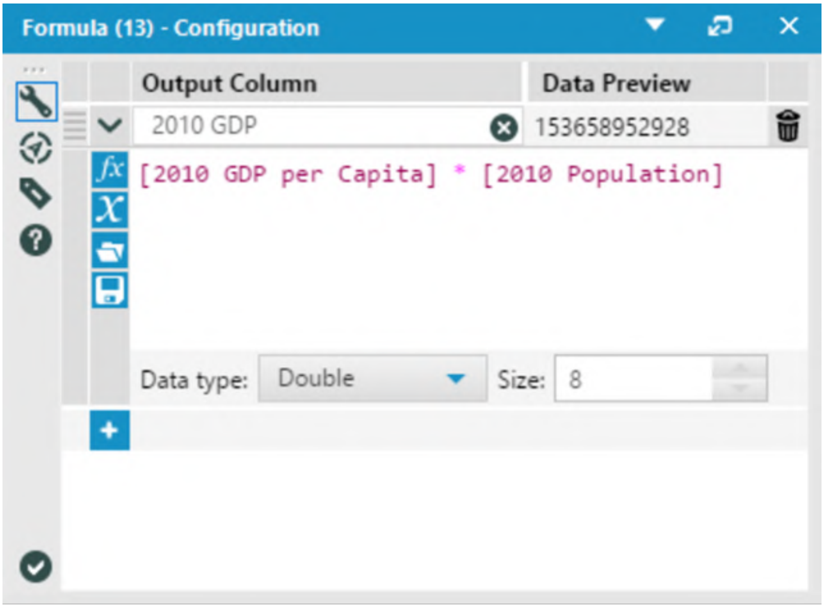


Figure-5-26 – Culturally Rich Total GDP Calculation

- Export the data to a file called *Culturally Rich* using the Output

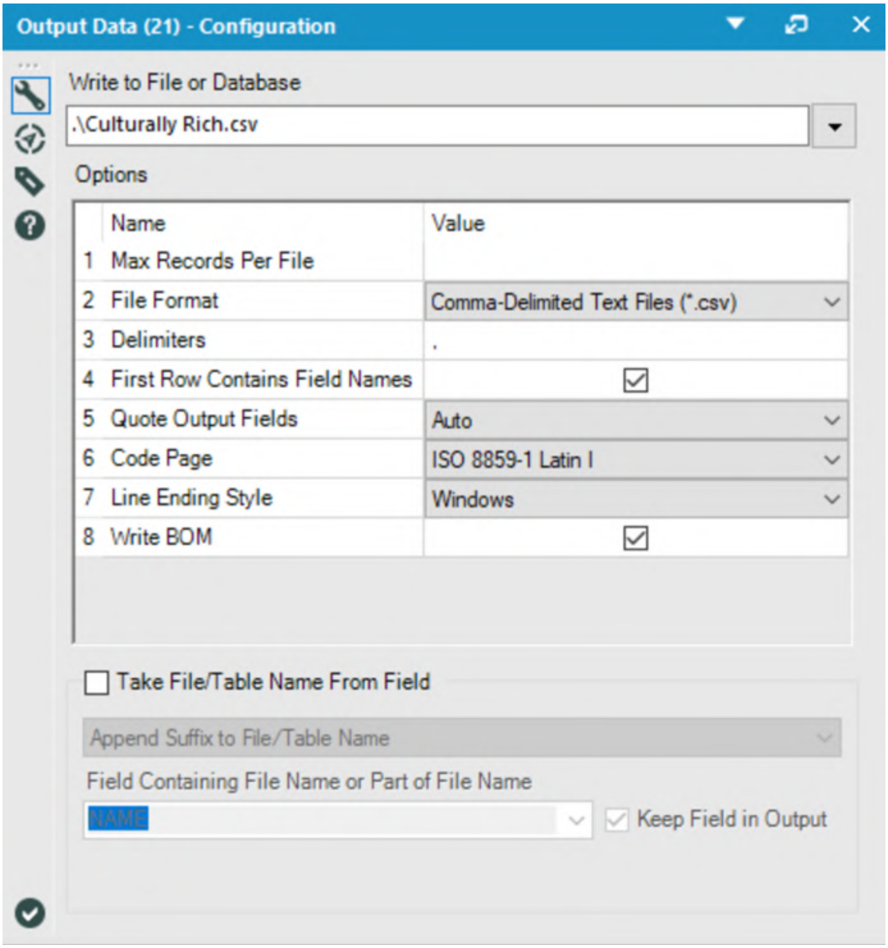


Figure-5-27 – Culturally Rich Output Data Configuration

The *Culturally Rich* data stream should look like this when it is complete.

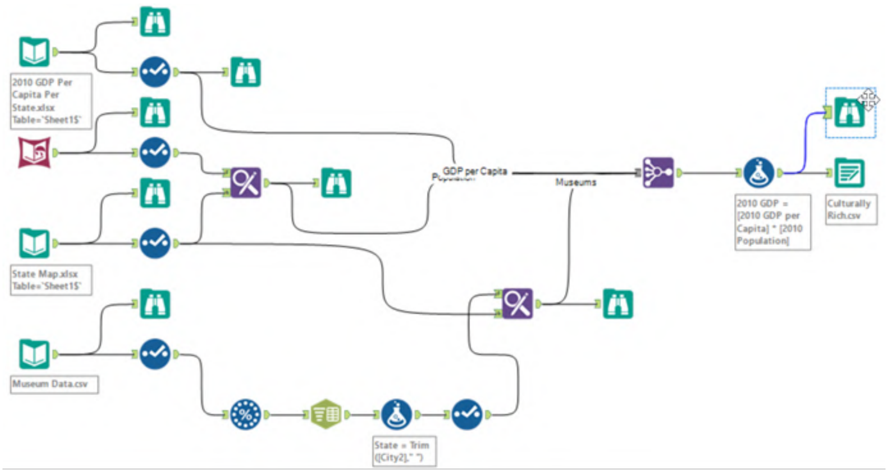


Figure-5-28 – Culturally Rich Complete Workflow

5.3 Culturally Divided

To... Alteryx Consultants

Subject Culturally Divided

Hey,

I want to create a map in another tool that will approximate the cultural appreciation of each state by identifying what the museum's per capita rates are. This may not be an appropriate measure, but I'm curious. Note that this time, we will need to take a count of the museums per state, not just apply the state data to each museum.

It would also be nice if I had a flag for the comparison to the average museum's per capita. (Above average, average, and below average).

I need to run to a meeting; do you think you can get this done by the time I get back?

Thanks.

Chapter 6
The Sport Report

To... Alteryx Consultants
Subject The Sport Report
Attached Major Sport Teams.xlsx

Hey,

I got a call from a sports bar chain. They want to see a rough report that could show them exactly where the major professional baseball, basketball, football, and soccer teams in the United States and Canada are so they can target new locations.

I want you to help me build a report with the following properties.

- Every sport should start on a new page.
- There should be a generic company header and footer (we can use the default for now).
- It should say the name of the sport.
- There should be a map of all of the stadium locations for that sport.
- There should be a bar graph that has the total count of teams by the governing association.
- There should be a table that has the name of the association, the league, or conference, the division, the team, the city, the home park, the street address, and the zip code, for each team. (The table should be allowed to take in as many pages as necessary.)

Thanks.

To the reader,

We will be covering the basics of report building in Alteryx. However, since building reports in Alteryx is an advanced technique, we will not be exploring the Report tools as completely as the tools in other sections.

USEReady

6.1 Tools & Concepts

6.1.1 Charting

Concept – Report Building

<table>
<tr><td rowspan="2">
Figure-6-1-Charting</td><td>**Tool Palette: Reporting**</td></tr>
<tr><td>Creates a chart object for reporting.
For more details use the link below.
bit.ly/2KgZl9U</td></tr>
</table>

6.1.2 Create Points

Concept – Spatial Objects

<table>
<tr><td rowspan="2">
Figure-6-2-Create Points</td><td>**Tool Palette: Spatial**</td></tr>
<tr><td>Converts decimal latitudes and longitude columns into a new field called Centroid with a point object for each record.
For more details use the link below.
bit.ly/2HupHTY</td></tr>
</table>

6.1.3 Layout

Concept – Maps

 **Figure-6-3-Layout**	**Tool Palette: Reporting** Creates a data stream with reporting objects and allows us to structure their layout. For more details use the link below. bit.ly/2Kadsxv

6.1.4 Render

Concept – Report Transformation

 **Figure-6-4-Render**	**Tool Palette: Reporting** Transforms report snippets into presentation-quality reports into a variety of formats. For more details use the link below. bit.ly/2HRB6k2

6.1.5 Report Footer

Concept – Report Transformation

	Tool Palette: Reporting

<table>
<tr><td>
Figure-6-5-Report Footer</td><td>Allows a user to easily setup and put a footer onto their report.
For more details use the link below.
bit.ly/2HZooNH</td></tr>
</table>

6.1.6 Report Map

Concept – Maps

<table>
<tr><td rowspan="2">
Figure-6-6-Report Map</td><td>Tool Palette: Reporting</td></tr>
<tr><td>Enables the user to create a map image from the Alteryx workflow.
For more details use the link below.
bit.ly/2I0kbts</td></tr>
</table>

6.1.7 Report Header

Concept – Report Transformation

<table>
<tr><td rowspan="2">
Figure-6-7-Report Header</td><td>Tool Palette: Reporting</td></tr>
<tr><td>Allows a user to easily setup and put a header onto their report.
For more details use the link below.
bit.ly/2KgpiXa</td></tr>
</table>

6.1.8 Table

Concept – Report Transformation

 **Figure-6-8-Table**	**Tool Palette: Reporting**
	Creates a data or pivot table to output in a report via the Render tool. For more details use the link below. bit.ly/2Fget3H

6.1.9 Email

Concept – Share Reports

 **Figure-6-9-Email**	**Tool Palette: Reporting**
	Allows you to use a data stream to send emails for each record. For more details use the link below. bit.ly/2HwYK28

6.1.10 Image

Concept – Report Formatting

 **Figure-6-10-Image**	**Tool Palette: Reporting**
	Makes it possible for an image to be output in a report via the Render tool. For more details use the link below. bit.ly/2HqW8qn

6.1.11 Report Text

Concept – Report Transformation

 **Figure-6-11-Report Text**	**Tool Palette: Reporting**
	Creates a text element to output in a report via the Render tool. For more details use the link below. bit.ly/2FgHwnm

6.2 Major Sports Teams

To... Alteryx Consultants

Subject The Sport Report

Attached Major Sport Teams.xlsx

Hey,

I got a call from a sports bar chain. They want to see a rough report that could show them exactly where the major professional baseball, basketball, football, and soccer teams in the United States and Canada are so they can target new locations.

I want you to help me build a report with the following properties.

- Every sport should start on a new page.
- There should be a generic company header and footer (we can use the default for now).
- It should say the name of the sport.
- There should be a map of all of the stadium locations for that sport.
- There should be a bar graph that has the total count of teams by the governing association.
- There should be a table that has the name of the association, the league, or conference, the division, the team, the city, the home park, the street address, and the zip code, for each team. (The table should be allowed to take in as many pages as necessary.)

Thanks.

When building reports in Alteryx, we need to remember the following key points:

- Most report objects should be created in individual data streams and then brought together. The exceptions are *Layout*, *Header*, and *Footer* objects.
- Report objects should be in the same record for each grouping.

Let us start by connecting to the data *Chapter 6 –Major Sports Teams.xlsx*.

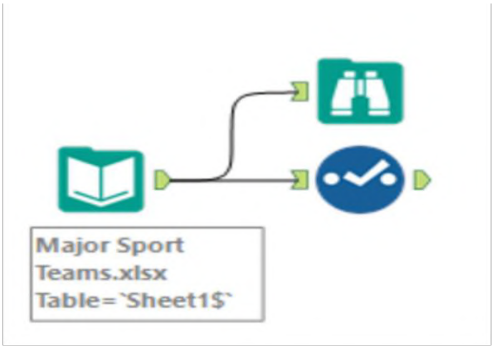


Figure-6-12- Opening Sport Report

As in any other workflow, we will bring in the data and ensure that it is in the correct format.

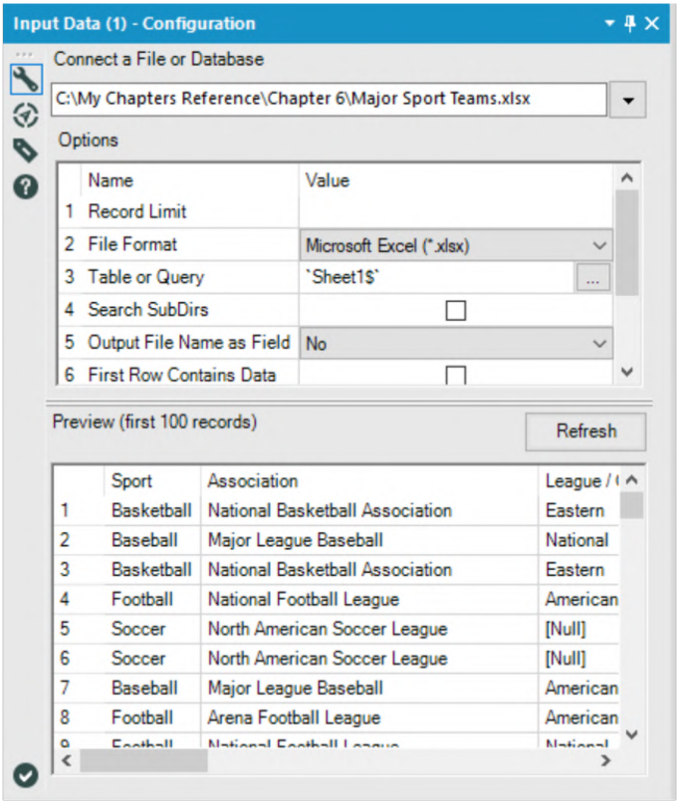


Figure-6-13-Sports Report – Data formatting

Since we have *Latitude* and *Longitude* fields, we can easily start tackling the mapping aspect of our report. We start by adding a *Create Points* tool to the end of the workflow and using the accompanying configuration.

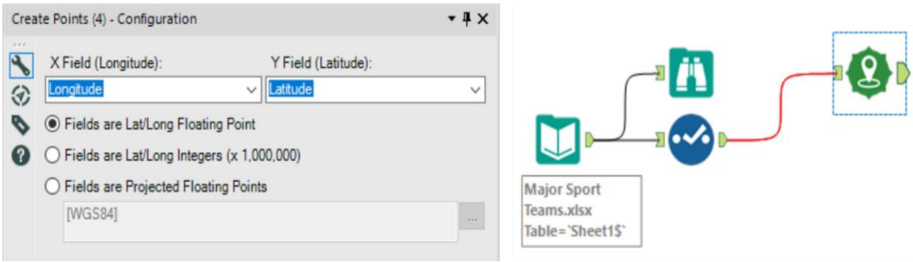


Figure-6-14-Sports Report-Mapping and Creating Points

Then, when we add a *Browse* tool and run the workflow, we will see that we have a field called *Centroid* in the last column with a green point object in each record. This is a special spatial field, and we will use this field in order to plot the locations of each team's home field.

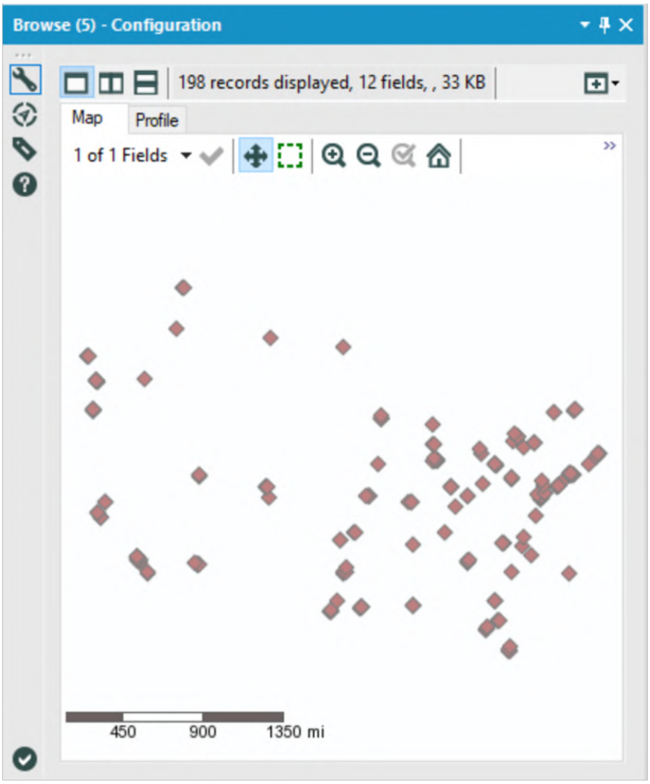


Figure-6-15-Sports Report – Run Module Using Browse

The resulting data stream will look like below.

Figure-6-16-Sports Report Data Stream

Since we are only going to use the one input for this *Report Map,* we can leave the connection labeled *#1*. We need to think about the way the report is going to look, so we should consider the overall layout. It will likely flow better if the map and chart are placed next to each other. This way, we know that image will not take more than half of the page.

In the following pages, the map configuration will allow us to create the desired maps.

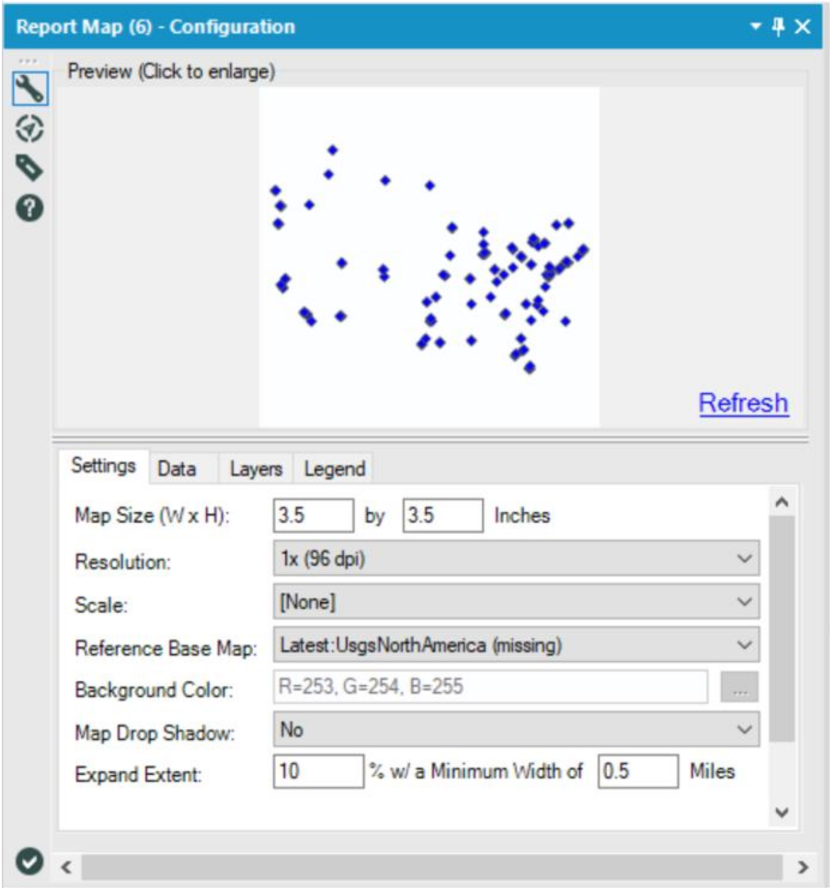


Figure-6-17-Report Map Setting Configuration

The following are the Report Map Data and Layers settings.

Note that we want to group by sport to make one map for each.

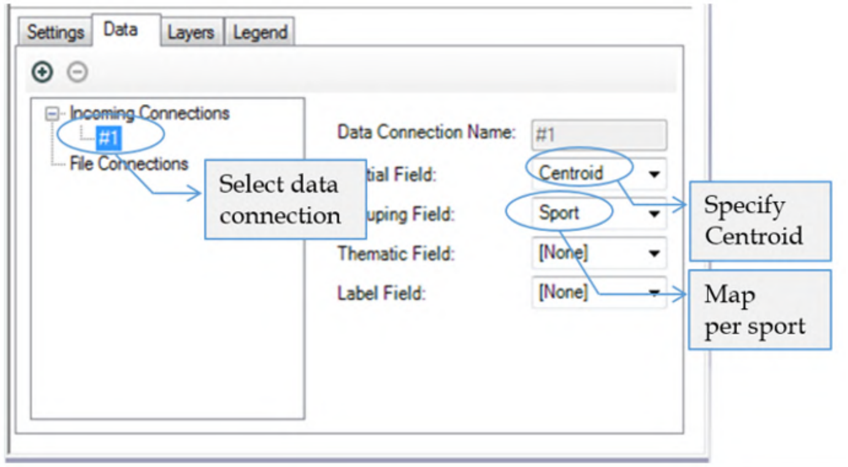


Figure-6-18-Sport Report – Report Map Data Configuration

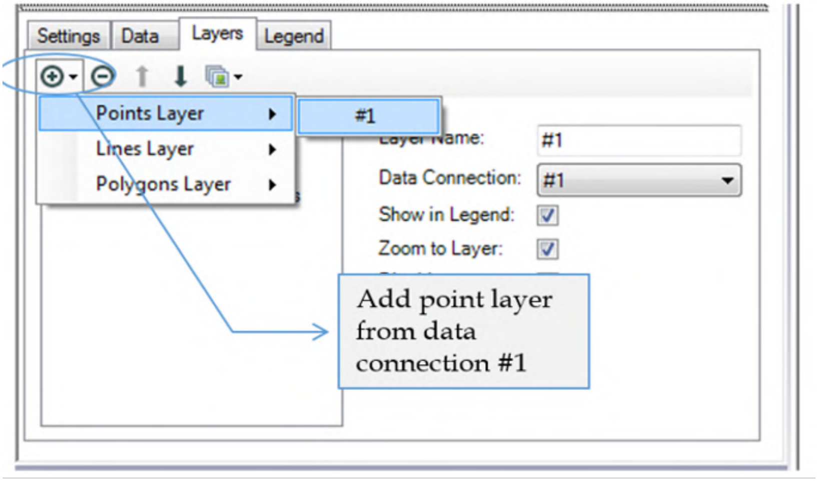


Figure -6-19-Sport Report-Report Map Layers Configuration

Normally, we would not put a *Browse* tool after *Report* tools, but it will help us understand the output of the *Report* tools.

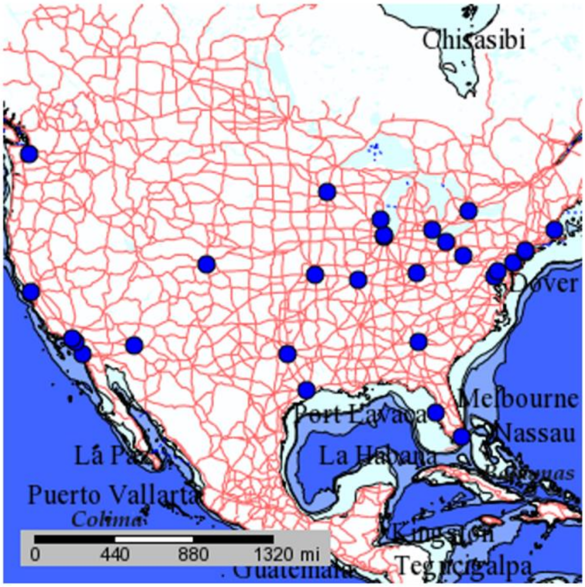


Figure-6-20-Sport Report- Browse

If we run the module, we should see the output in the *Browse* tool at the end of the data stream.

We see that we have four records, one for each of the four sports we grouped by, identified by the *Group* column at the right. A *Map* field contains the *Report* object map for each sport. The *BoundingRect* spatial field can be used to create new maps that are zoomed to the same area as these. Since we will not be creating any more maps, we can ignore this field.

Also, we should note that we can preview the map and report it if we want by selecting the tabs along the top.

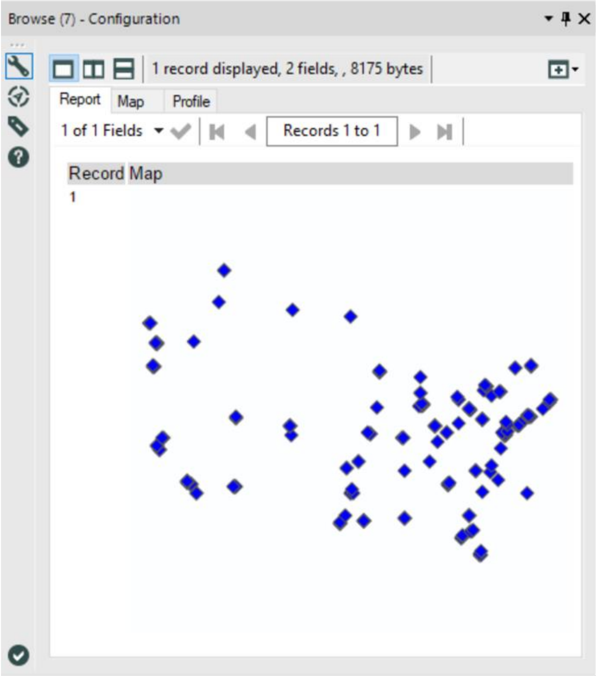


Figure-6-21- Sport Report – Chart Creation using Map

Now that we have our maps, let us create our charts. We will be making a bar graph for each sport, showing the number of teams in each association. This means that the first thing we should do is summarize the data by sport and association, then take the count of the team names.

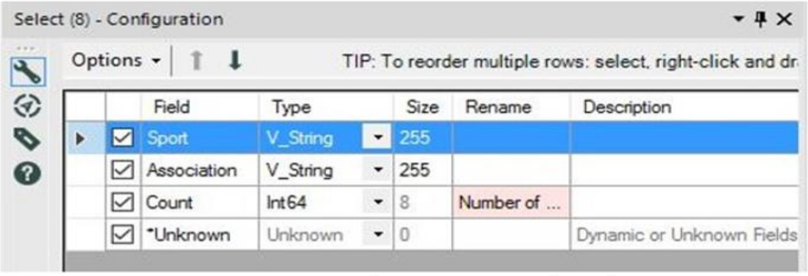


Figure-6-22-Sport Report – Chart creation using Select

Since we want to have clean field names while reporting, we will change the name of the *Count* column to *Number of Teams*.

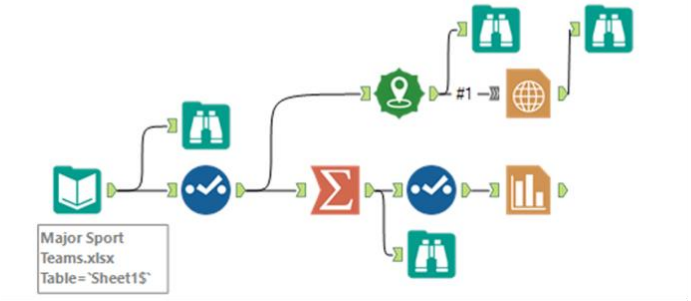


Figure-6-23-Sport Report –Adding Charting tool to end of data stream

Now that we have the data prepared we can add a *Charting* tool to the end of this data stream and apply the settings as shown in the image.

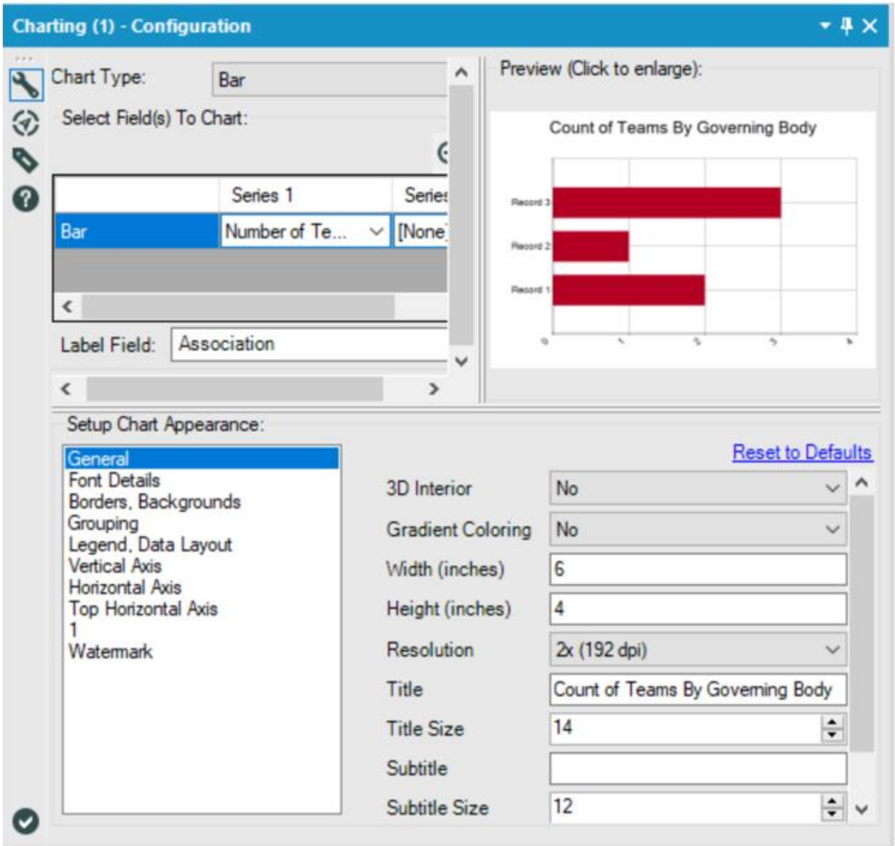


Figure-6-24-Sport Report – Charting Tool

The images below show the grouping, legend, and data layout settings made to the *Charting* tool.

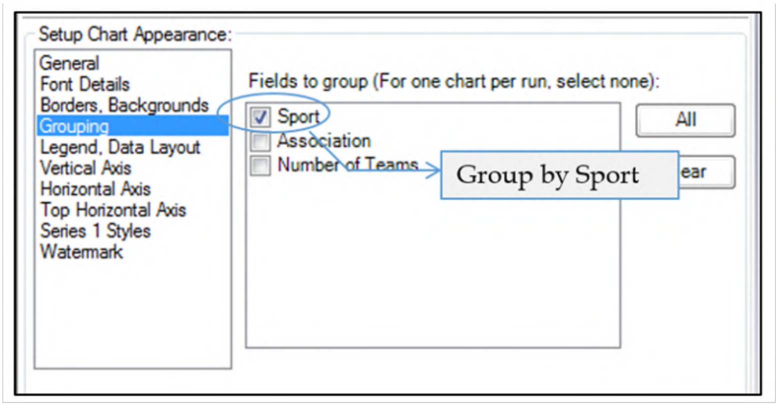


Figure-6-25-Sport Report-Setup Chart Appearance, Grouping set to Sport

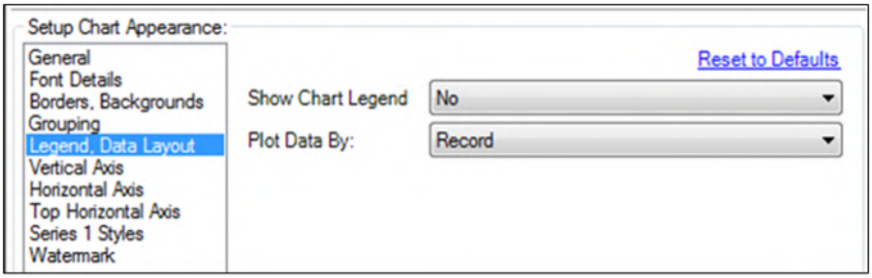


Figure-6-26-Sport Report-Setup Chart Appearance – Setting Legend, Data Layout

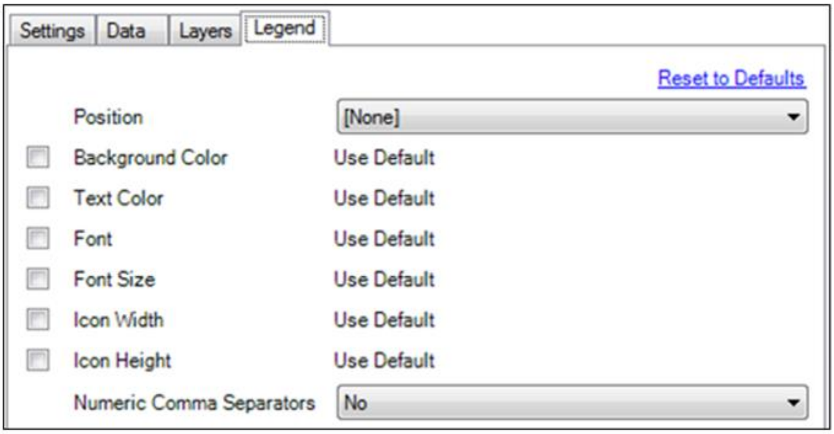


Figure-6-27- Sport Report – Setting Legend Settings(refer Report Map tool)

Let us understand the settings we have changed. The high-level settings are easy to understand. Since we wanted to create a bar chart, we selected *Bar*. Since we wanted to compare the number of teams, we selected that field for our series, and as we wanted to know the governing body related to each bar, we chose *Association*.

Under the *General* settings, we set the width and height to the same dimensions as the maps because we want them to fit side by side. We increased the resolution to have a cleaner image. We provided a descriptive title for the chart with a reduced text size so it fits on one line and we have more room for the chart. We grouped by *Sport* to make one chart for each. We removed the chart legend because we do not need it in this case.

Now that we have covered our maps and charts, we should set a title for each section. We have made a map and chart for each of the sports, but we haven't mentioned anywhere what the sport actually is. We need an object that we can put at the top of the page, with the sport in it.

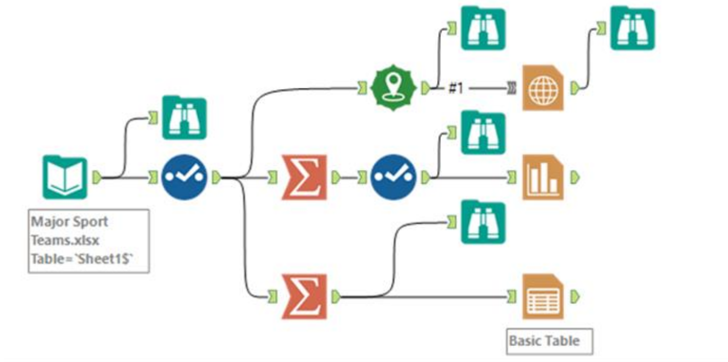


Figure-6-28-Sport Report – Data stream branching

In order to create this object, we will branch off into another data stream, as shown in the previous image. We will use the *Summarize* tool to group by *Sport* so that we have a single record for each. We are then going to add a *Table* tool to create our *Report* objects.

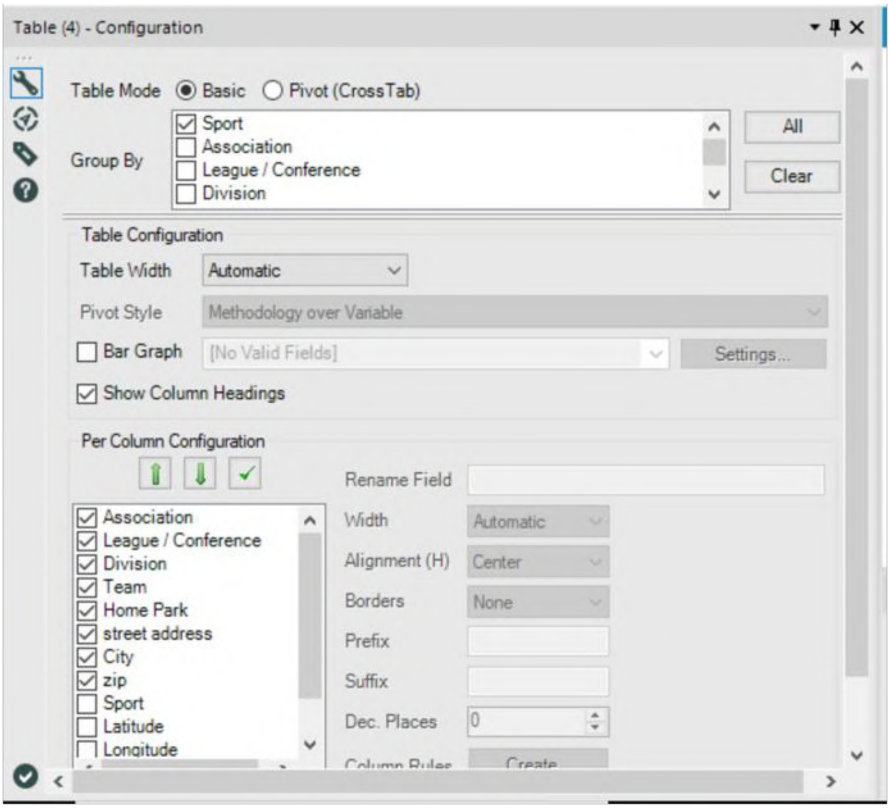


Figure-6-29-Sport Report – Grouping by Sports using Table

Since we want a *Table* object for each sport, we will group by Sport. Since we are using *Sport* as a section title, we will be able to identify the sections we are interested in. Additionally, since we want to ensure that only *Sport* makes it into this table, we can deselect the *Dynamic or Unknown Fields* option.

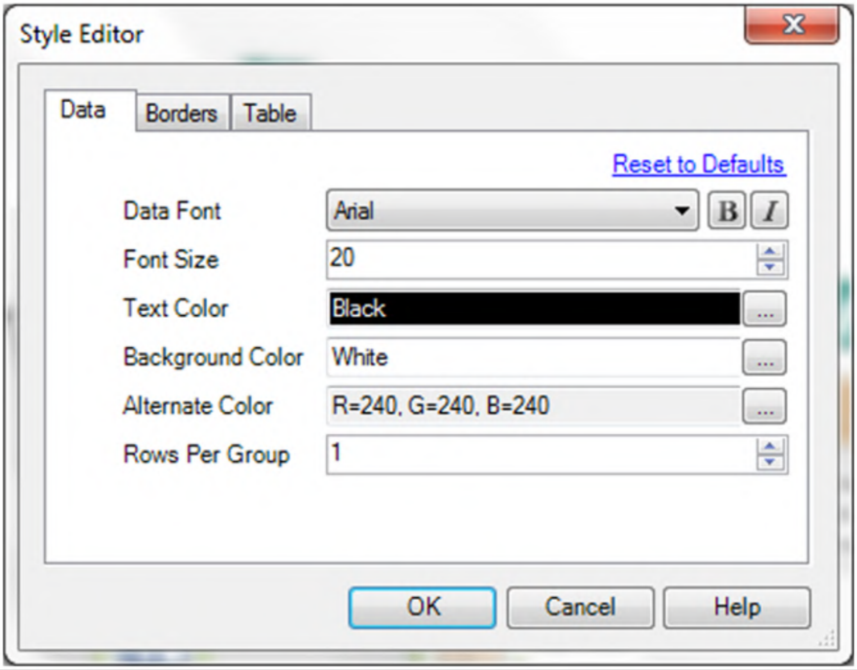


Figure-6-30-Sport Report – Setting Title Style using Style Editor

Since this will be the title in each section, we should increase its font size.

At this point, the following image is what our data stream will look like.

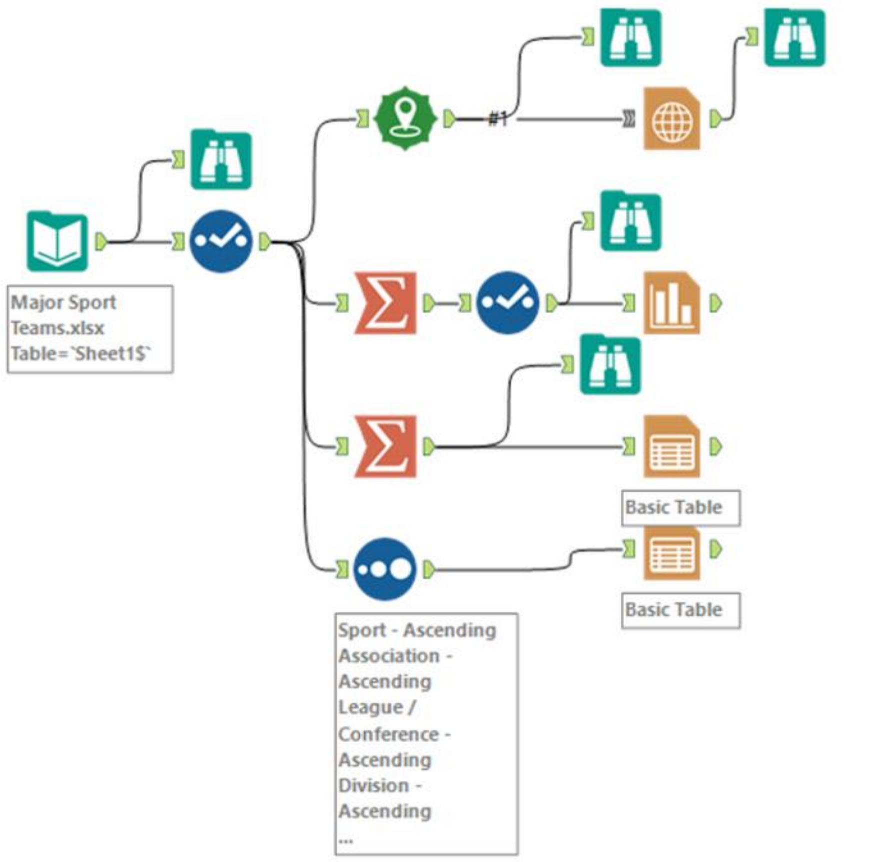


Figure-6-31-Sport Report – Data stream grouped by each report

We now need to create a table for the demographic information about the teams. Since this table will be in the report, it should be sorted so the consumer can easily see what's going on.

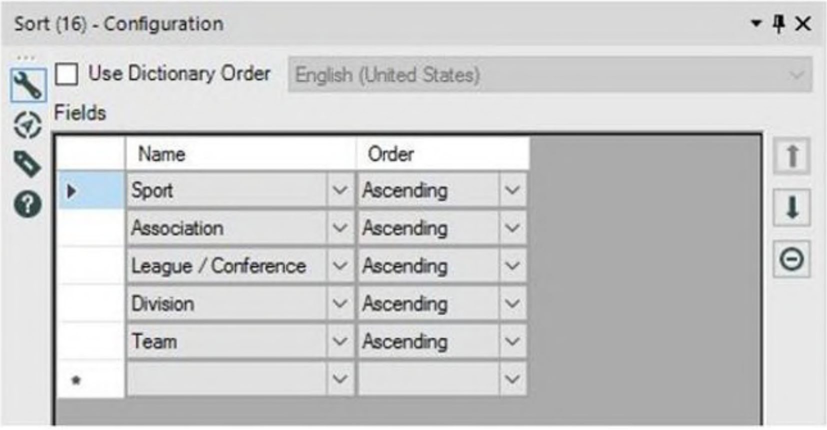


Figure-6-32- Sport Report – Applying Sort to Data

Let us sort the data like we see in the *Sort Configuration* window. This will allow the report consumer to easily scan the list for the team they are interested in.

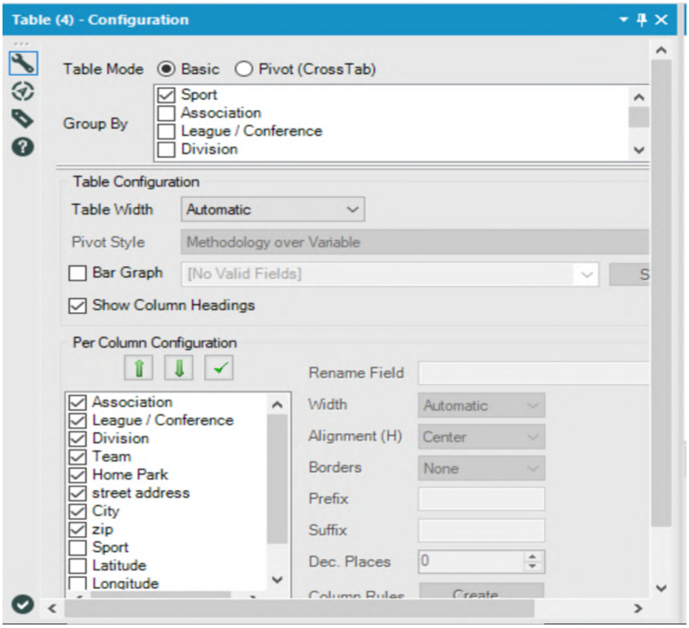


Figure-6-33-Sport Report – Team Information using tables

Using these settings, we can create a table with the desired useful information about each team.

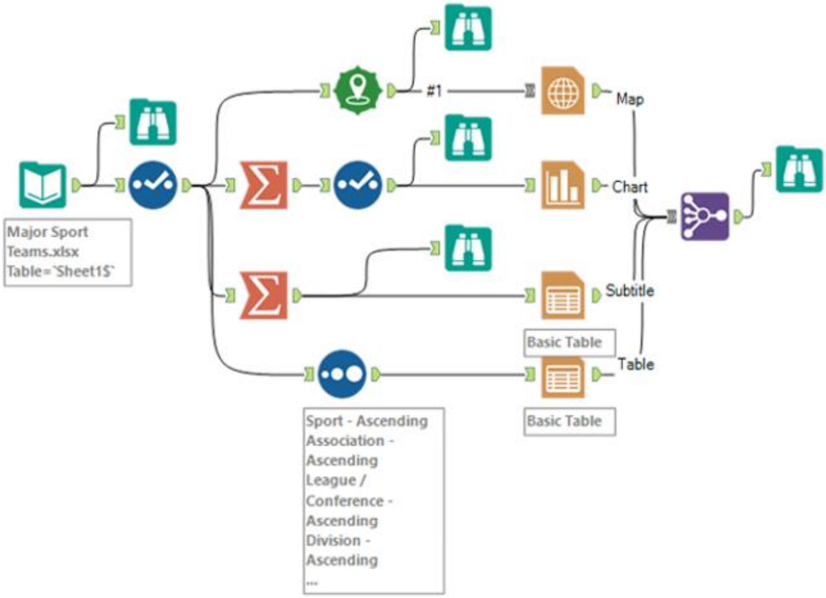


Figure-6-34-Sport Report – with separate data streams

Now that we have created each of the objects for the body of our report, we need to bring them into a single data stream so we can format the layout, header, and footer.

In the *Browse* tool following the map we created, we saw a single record for each sport because that's the field we grouped by. Adding a *Browse* tool after each of the report objects would also result in the same, except that instead of the field being called *Group*, it would retain the title *Sport*. Since we know that the sport name makes a unique key in all of our data streams, we can use the *Multiple Join* tool to bring them together.

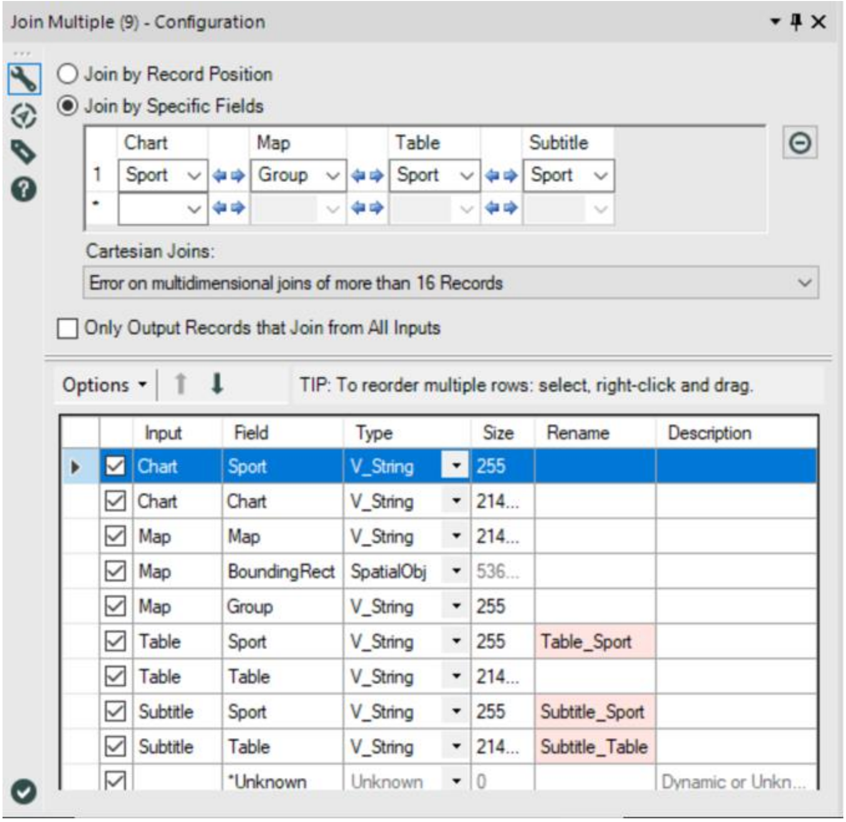


Figure-6-35-Sport Report – Multiple Joins using Sports name

If we had named the incoming connections the way we see them in the previous image, we would see something similar to the above *Join Multiple Properties* window. Let us use the same setup as above to keep the information that will help us downstream.

Figure-6-36-Sport Report – using Browse to get data after joining

At this point, we should have five fields: The *Sport* field for our understanding and the four report object fields use in the workflow.

Now that we have the objects for the body of our report, we need to incorporate the following into the layout:

- The sport should be at the top of each section.
- We designed the map and chart object to fit side by side, but there is no rule for where they should go.
- We have a table that can vary in length because it has one row for each team. In such a situation, the table is typically put at the bottom of the report body.

- Based on what we see here, we need two *Layout* tools. The first is a horizontal *layout* tool that will align the map and chart objects. The second *Layout* tool is to align the section title, map-chart layout, and team information table.

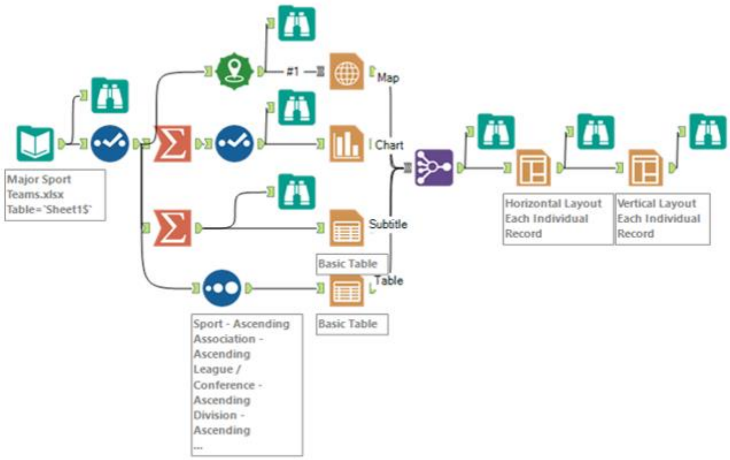


Figure-6-37-Sport Report – Updated Data Stream

Here are the settings for the first *Layout* tool.

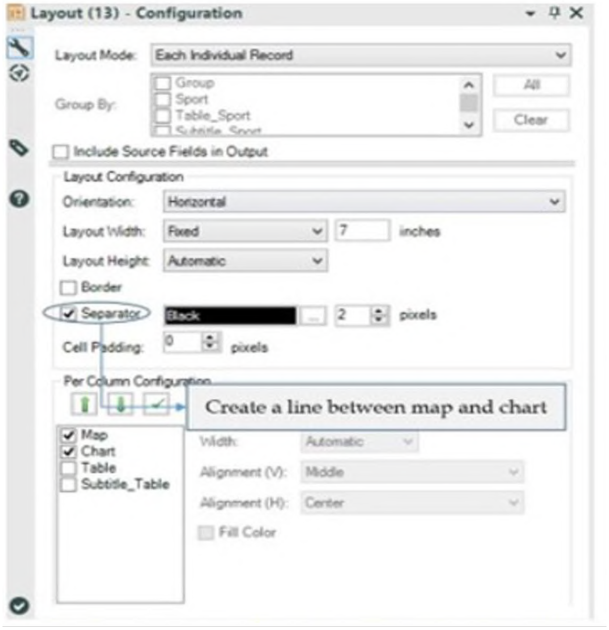


Figure-6-38-Sport Report –Using Layouts to align maps, team info table etc.

A *Browse* will show us the four columns for the first *Layout* tool.

Record #	Sport	Table	Subtitle_Table	Layout
1	Baseball	Table - View Browse Tool Report Tab	Table - View Browse Tool Report Tab	Layout - View Browse Tool Report Tab
2	Basketball	Table - View Browse Tool Report Tab	Table - View Browse Tool Report Tab	Layout - View Browse Tool Report Tab
3	Football	Table - View Browse Tool Report Tab	Table - View Browse Tool Report Tab	Layout - View Browse Tool Report Tab
4	Soccer	Table - View Browse Tool Report Tab	Table - View Browse Tool Report Tab	Layout - View Browse Tool Report Tab

Figure-6-39-Sport Report – Data after applying layouts

The image below shows the settings for the second *Layout* tool.

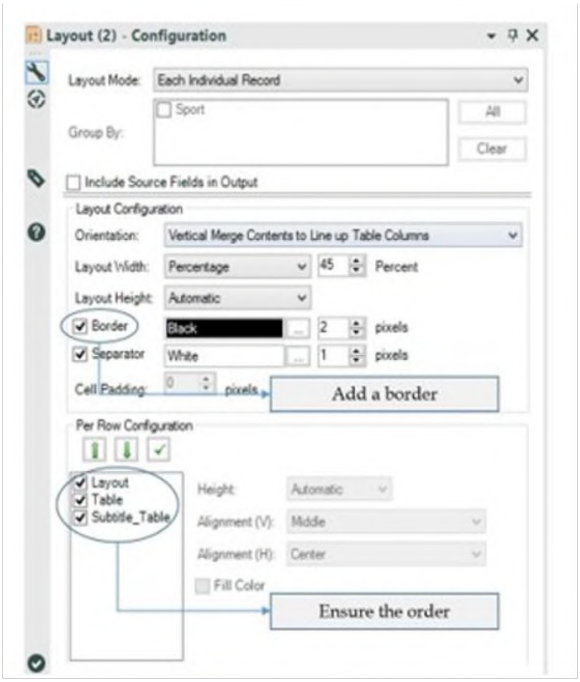
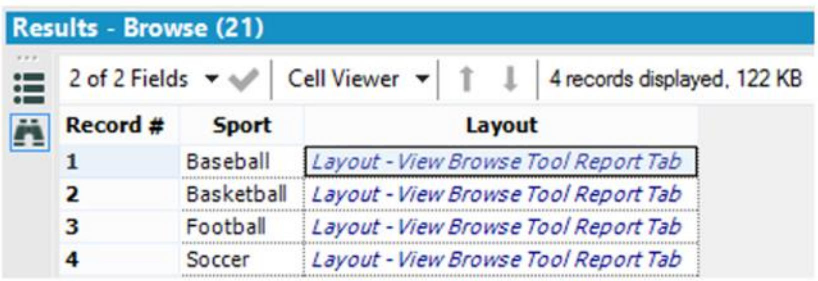


Figure-6-40-Sport Report – Applying Layout Configuration

Rerunning the flow now would show the two columns in the second *Layout* tool.



Record #	Sport	Layout
1	Baseball	*Layout - View Browse Tool Report Tab*
2	Basketball	*Layout - View Browse Tool Report Tab*
3	Football	*Layout - View Browse Tool Report Tab*
4	Soccer	*Layout - View Browse Tool Report Tab*

Figure-6-41-Sport Report-Selected Data after applying Layout

The data stream after having the Layouts in place would look like the below.

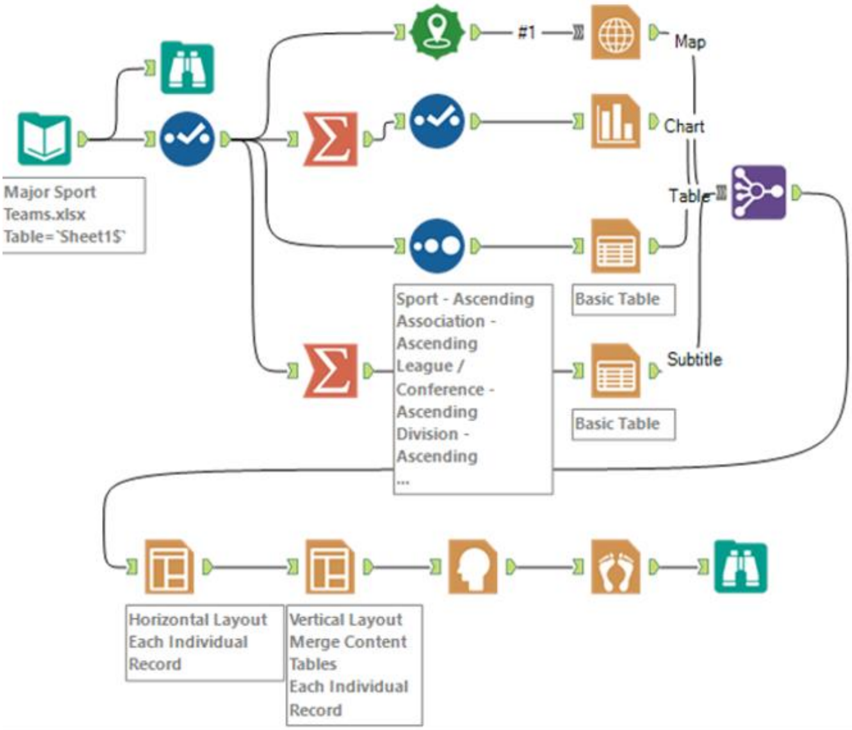


Figure-6-42-Sport Report-Data stream with Layouts in place

Now that we have combined all of the elements of the body of the report into a single report object, we can add a *Report Header* and *Report Footer* to the report. We will use the default settings for both because the client wants to see what would typically be there. We are ready to render the report, but before we do that, let us see what the data we are feeding into the *Render* tool looks like.

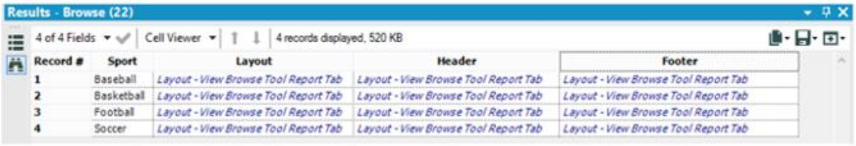


Figure-6-43-Sport Report – Data selected after addition of Report Header and Footer

As we can see, there is no special format for the *Header* and *Footer* fields. This means we could put any object into the *Header* and *Footer* options when we render the report. However, the *Report Header* and *Report Footer* tools create appropriately sized objects that are designed for standard report information.

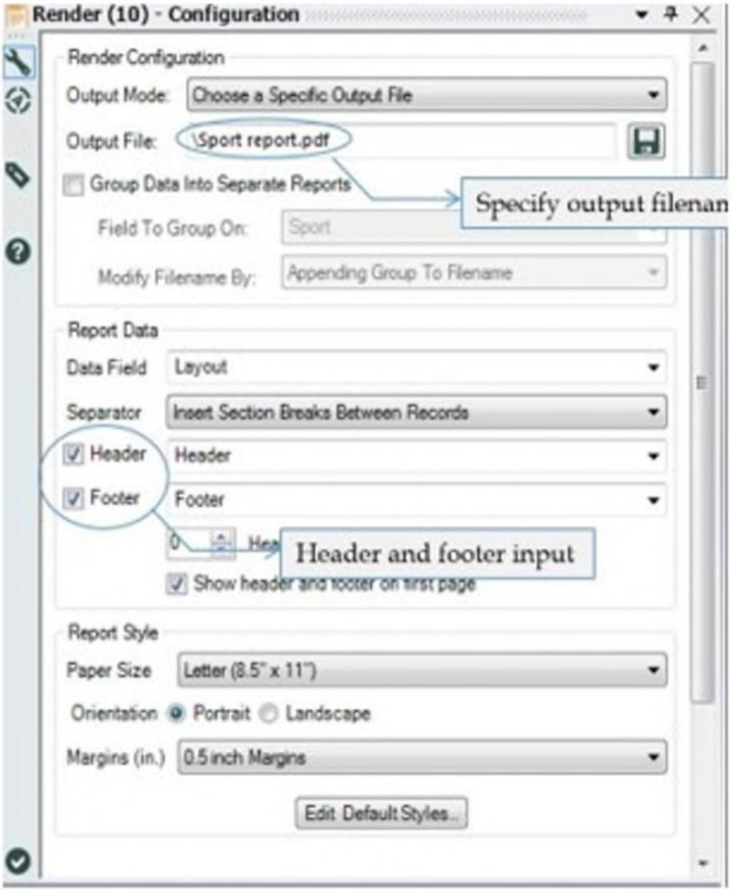


Figure-6-44-Sport Report-Applying Render to specify path

In the *Render Properties* window, we will specify that we are creating an output file and specify a path for it. Since we want to start each sport on a new page, we will use the section break option. Finally, we will turn the Header and Footer fields on and define them. The Major Sports Teams data stream should look like the following when it's complete.

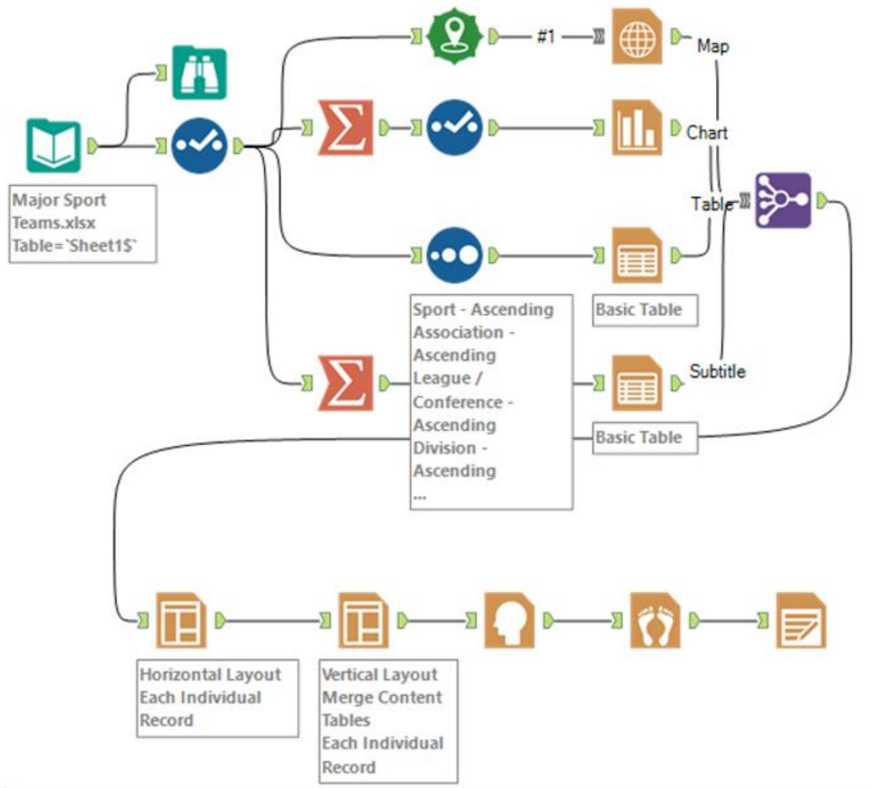
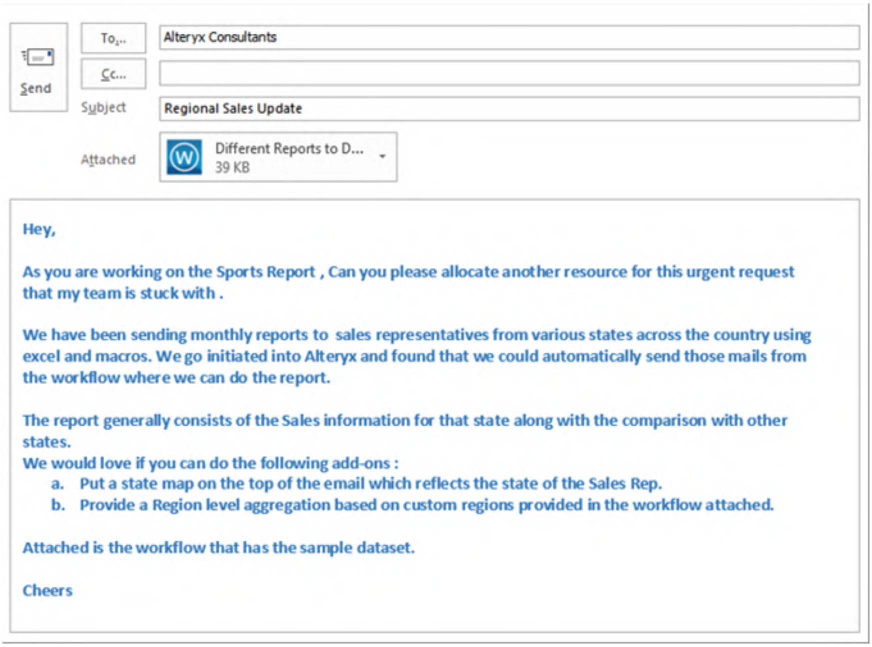


Figure-6-45-Sport Report Data stream on completion

6.3 Regional Sales Monthly Update



Send

To... Alteryx Consultants

Cc...

Subject Regional Sales Update

Attached Different Reports to D... 39 KB

Hey,

As you are working on the Sports Report , Can you please allocate another resource for this urgent request that my team is stuck with .

We have been sending monthly reports to sales representatives from various states across the country using excel and macros. We go initiated into Alteryx and found that we could automatically send those mails from the workflow where we can do the report.

The report generally consists of the Sales information for that state along with the comparison with other states.

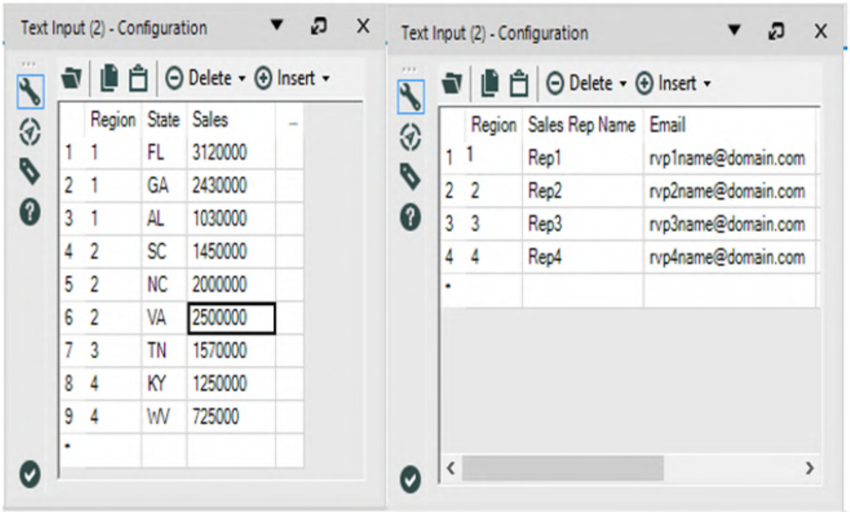
We would love if you can do the following add-ons :

a. Put a state map on the top of the email which reflects the state of the Sales Rep.
b. Provide a Region level aggregation based on custom regions provided in the workflow attached.

Attached is the workflow that has the sample dataset.

Cheers

This sounds like an interesting assignment. Let us start by checking the dataset.



	Region	State	Sales
1	1	FL	3120000
2	1	GA	2430000
3	1	AL	1030000
4	2	SC	1450000
5	2	NC	2000000
6	2	VA	2500000
7	3	TN	1570000
8	4	KY	1250000
9	4	WV	725000

	Region	Sales Rep Name	Email
1	1	Rep1	rvp1name@domain.com
2	2	Rep2	rvp2name@domain.com
3	3	Rep3	rvp3name@domain.com
4	4	Rep4	rvp4name@domain.com

Figure-6-46-Region Sales Data

The next step that we have is to create a *Table* that formulates the sales numbers and join the email information of the different representatives.

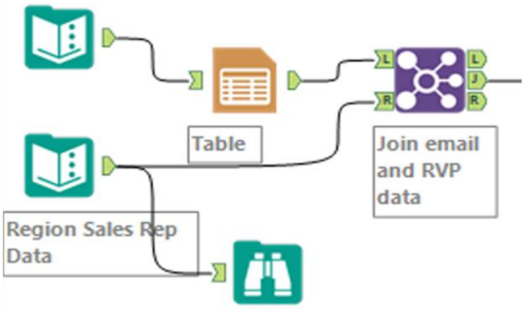


Figure-6-47-Sales Data joined with Email

The next step is to create the *Title* using *Report Text* of the report, add the *Image* of the states and create the *Report Text* and *Layout* of the E-mail that we need to send out.

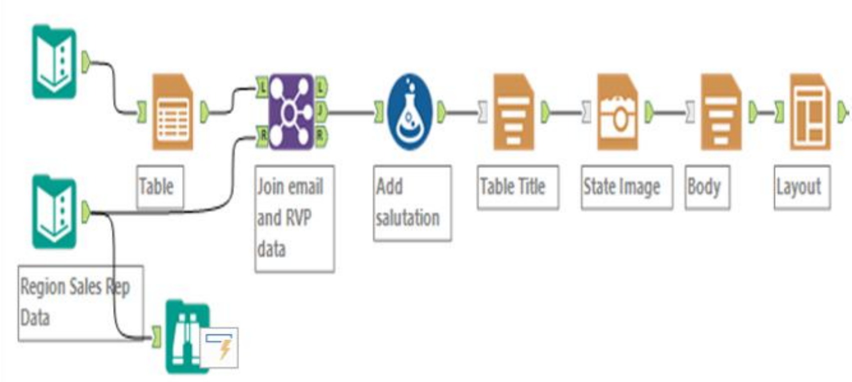


Figure-6-48-Components of the Email

In this step, we added an additional component where we added a salutation based on the time of the day that the *Email* is being sent to add to the aesthetics and added a title that would change as per Region.

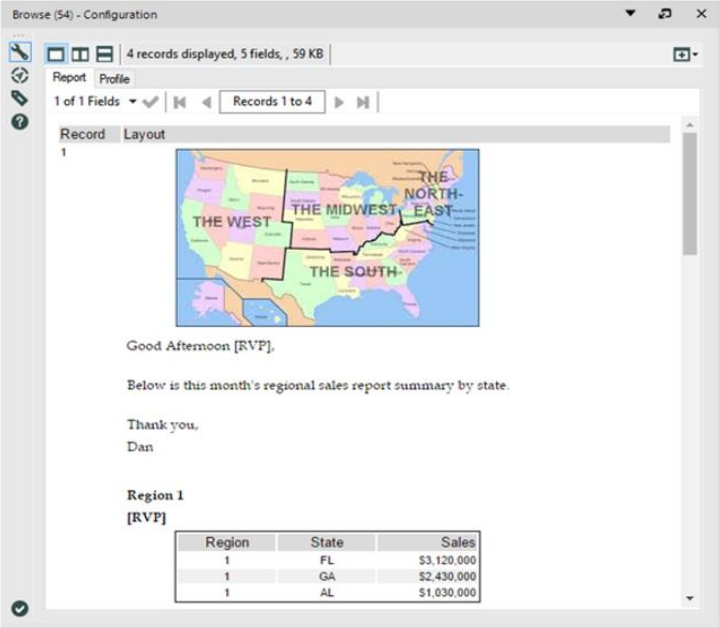


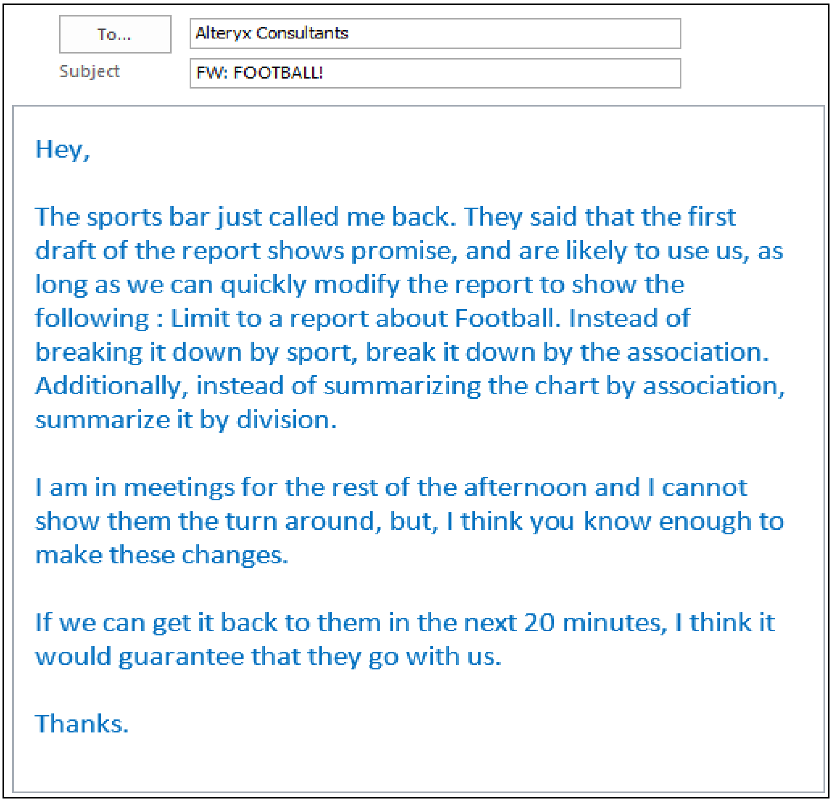
Figure-6-49-Email Preview

The preview of the email shows us the complete map of the United States and the state wise split of the region's sales numbers for the current month.

Now all that is required to provide the necessary email id information and configure the email tool and the output would be sent to multiple Sales representatives across the country.

Figure-6-50-Regional sales monthly report data stream on completion

6.4 Football!



To...
Alteryx Consultants

Subject
FW: FOOTBALL!

Hey,

The sports bar just called me back. They said that the first draft of the report shows promise, and are likely to use us, as long as we can quickly modify the report to show the following : Limit to a report about Football. Instead of breaking it down by sport, break it down by the association. Additionally, instead of summarizing the chart by association, summarize it by division.

I am in meetings for the rest of the afternoon and I cannot show them the turn around, but, I think you know enough to make these changes.

If we can get it back to them in the next 20 minutes, I think it would guarantee that they go with us.

Thanks.

CHAPTER 7

Expensive Beauty Product

To... Alteryx Consultants
Subject Expensive Beauty Products
Attached Body Care Co.csv

Hey,

See the email below from Jan at Body Care Co.

Body Care Co. is a company that I work with pretty regularly. They have Alteryx and tend to ask us to work on projects when they don't have the bandwidth.

I think working together on this would be a good way to introduce you to a few new topics.

Thanks.

I've got something fairly straightforward for you this time.

I am using data that looks like the attached data set in order to run a lot of analyses. I'm swamped, so if you could build me an Alteryx flow that I can use going forward, that would be great!

I am using a dashboarding tool that needs to be highly responsive. Unfortunately, it is running too slow because I don't have an easy way to identify the selected data.

Below is the logic to create a flag for a number of different things I need to look at. Would you be able to create a process so the data can be modified to have these flags?

1) Lotions ("Lotion" is in the description)
2) Moisturizing Products ("Moisturizing" is in the description)
3) Baby Products ("Baby" is in the description)
4) Acne Products ("Anti-Acne" or "Anti-Oil" is in the description)
5) Create groups for all products that are named 90% identically by word (we are looking into similarly named products to see how their sales correlate).

Thanks in advance,
Jan

7.1 Tools & Concepts

7.1.1 Fuzzy Match

Concept- Fuzzy Logic

<table>
<tr><td rowspan="2">
Figure 7-1 - Fuzzy Match</td><td>**Tool Palette: Join**</td></tr>
<tr><td>The *Fuzzy Match* tool allows us to apply fuzzy logic to a data stream to match similar items.
For more details use the link below.
bit.ly/2K8fvCq</td></tr>
</table>

7.1.2 Record ID

Concept- Indexing

<table>
<tr><td rowspan="2">
Figure 7-2 – Record ID</td><td>**Tool Palette: Preparation**</td></tr>
<tr><td>The *Record ID* tool creates an index column in our data stream.
For more details use the link below.
bit.ly/2vyDlDX</td></tr>
</table>

7.1.3 RegEx

Concept- Regular Expressions, Pre-Calculation

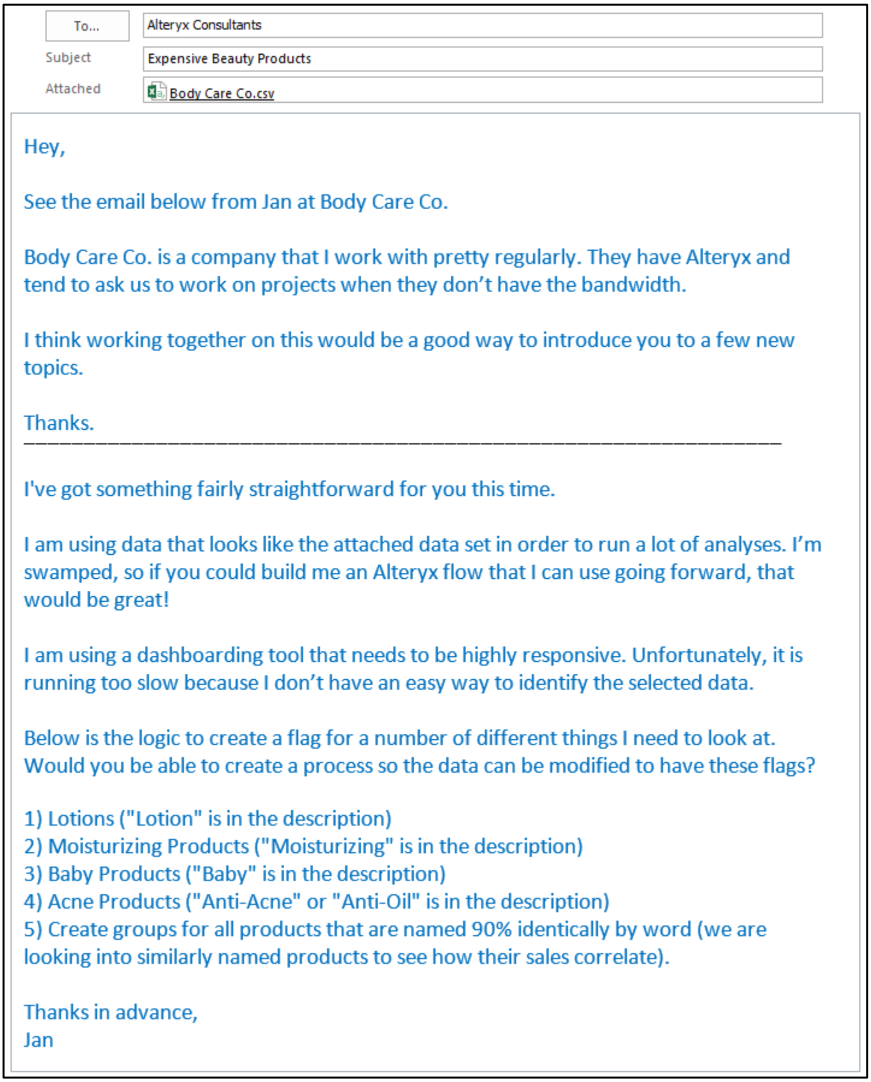
 **Figure 7-3 – RegEx**	**Tool Palette: Parse**
	The *RegEx* tool allows us to run regular expressions on a field in our data stream. For more details use the link below. bit.ly/2qNN3Oe

7.1.4 Unique

Concept- Removing Duplicates

 **Figure 7-4 – Unique**	**Tool Palette: Preparation**
	The *Unique* tool allows us to separate unique or duplicate records based on the field(s) we select. For more details use the link below. bit.ly/2HiRLcY

7.2 Expensive Beauty Products



To... Alteryx Consultants
Subject Expensive Beauty Products
Attached Body Care Co.csv

Hey,

See the email below from Jan at Body Care Co.

Body Care Co. is a company that I work with pretty regularly. They have Alteryx and tend to ask us to work on projects when they don't have the bandwidth.

I think working together on this would be a good way to introduce you to a few new topics.

Thanks.

I've got something fairly straightforward for you this time.

I am using data that looks like the attached data set in order to run a lot of analyses. I'm swamped, so if you could build me an Alteryx flow that I can use going forward, that would be great!

I am using a dashboarding tool that needs to be highly responsive. Unfortunately, it is running too slow because I don't have an easy way to identify the selected data.

Below is the logic to create a flag for a number of different things I need to look at. Would you be able to create a process so the data can be modified to have these flags?

1) Lotions ("Lotion" is in the description)
2) Moisturizing Products ("Moisturizing" is in the description)
3) Baby Products ("Baby" is in the description)
4) Acne Products ("Anti-Acne" or "Anti-Oil" is in the description)
5) Create groups for all products that are named 90% identically by word (we are looking into similarly named products to see how their sales correlate).

Thanks in advance,
Jan

The problem that Jan is describing is important to be aware of when we create data for dashboard and reporting tools. These tools are expected to create visuals in real time based on dynamic user inputs. Ideally, the performance is fast enough that the dashboard consumers do not observe a lag in the data load.

In order to achieve this goal, dashboard creators need well-designed and usually tidy data. This means that as much should be built into the data preparation process as possible. In Jan's case, they are doing some complex string calculations in order to filter the data appropriately.

Best practice is that any calculation the dashboard creator needs to run at a record level should be done in the data preparation stage.

To improve performance, pre-calculation steps are performed and it is a fundamental part of good data preparation.

Even when we are not preparing the data for analytics tools, it is important to pre-calculate fields the end user may need. This is because building and vetting a calculation once in a workflow means there will be a significant reduction in risk caused by human error downstream.

Let us imagine that we work for a large retailer that wants to know the profit ratios for each transaction. We may have 20 different analysts working on this project and reproducing the results every month. If these analysts are each rewriting the same formula *(Profit/Sales)* every month, it means we need to make sure they do not accidently type *(Sales/Profit)* in 240 different formulas per year. Instead, if we know they need transaction level profit ratio, we can create a standard formula in the data preparation process that writes this value into the source data that they all use.

Besides the pre-calculation, we can see from this email that Jan is only interested in a module. This is the case because they work with Alteryx and will do some modifications, like replacing the data connection and outputs. We will often come across things like this when we are supporting other Alteryx users because they either do not know how or do not have time to build a portion of a module.

Considering the information in the email, it seems Jan was talking about flagging item names that fall into each of those buckets. We will use two techniques to achieve this. For requirements one through four, we will write regular expressions that create flags for matches and mismatches. And for the fifth, we will use fuzzy logic to define grouping for each of the tools.

Regular expressions are pattern-matching formulae that allow us to define a pattern to look for in a string. Alteryx uses the Perl programming language syntax for its regular expressions, so any documentation related to Perl regular expressions will help us get up to speed. A cheat sheet is provided at the end. (See Appendix I.)

Fuzzy logic is essentially a logical process that allows computers to say, "Eh, close enough!" As far as Alteryx is concerned, this is limited to string parsing; however, more generally this extends to programming a definition of *tall*, *heavy*, and *blue*. There are various methodologies of matching strings. Alteryx uses the *Levenshtein Distance* and *Jaro Distance* to achieve this. The nuances of the differences are well documented and it is suggested that the reader learns more about them if they plan to use Alteryx's *Fuzzy Match* tool.

Figure 7-6 – Expensive Beauty Products Body Care Company Data

If we connect to the file in *Chapter 7 - Body Care Co.csv*, we see that Alteryx is reading all of the fields in this file as strings.

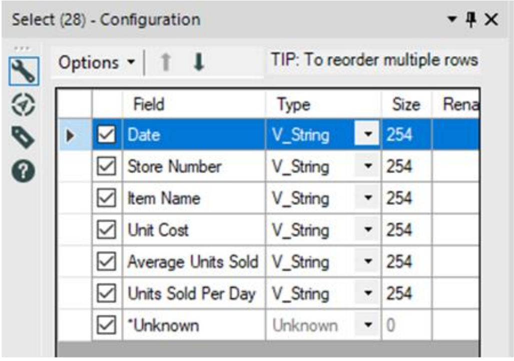


Figure 7-7 – Expensive Beauty Products Selection Configuration

Ideally, we should change the last three fields to numeric. But since we are only going to use *Item Name* and we don't know what the rest of Jan's process looks like, we should leave the fields the way they are.

Since we need to make sure Jan knows what each piece of this module is doing, we have to make sure that it is well documented. The first thing we will do is put our initial connection into a tool container that indicates that the entire set of tools should be replaced when the connection is changed.

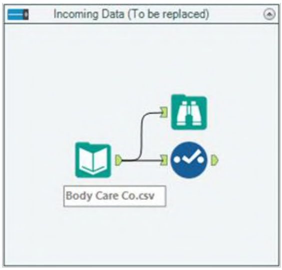
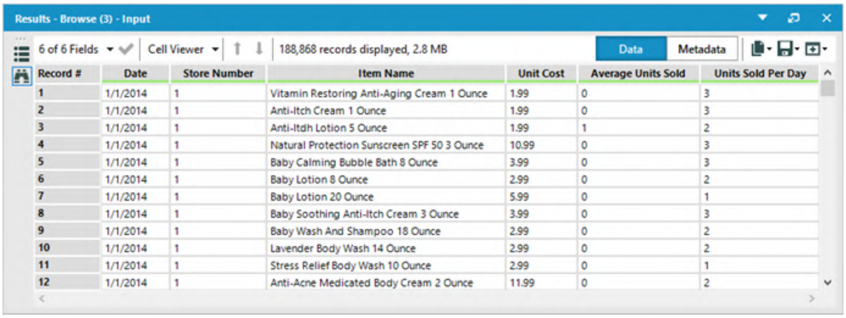


Figure 7-8 – Expensive Beauty Products Input Data – to be replaced in a container

Let us take a look at the contents of this file.



Record #	Date	Store Number	Item Name	Unit Cost	Average Units Sold	Units Sold Per Day
1	1/1/2014	1	Vitamin Restoring Anti-Aging Cream 1 Ounce	1.99	0	3
2	1/1/2014	1	Anti-Itch Cream 1 Ounce	1.99	0	3
3	1/1/2014	1	Anti-Itdh Lotion 5 Ounce	1.99	1	2
4	1/1/2014	1	Natural Protection Sunscreen SPF 50 3 Ounce	10.99	0	3
5	1/1/2014	1	Baby Calming Bubble Bath 8 Ounce	3.99	0	3
6	1/1/2014	1	Baby Lotion 8 Ounce	2.99	0	2
7	1/1/2014	1	Baby Lotion 20 Ounce	5.99	0	1
8	1/1/2014	1	Baby Soothing Anti-Itch Cream 3 Ounce	3.99	0	3
9	1/1/2014	1	Baby Wash And Shampoo 18 Ounce	2.99	0	2
10	1/1/2014	1	Lavender Body Wash 14 Ounce	2.99	0	2
11	1/1/2014	1	Stress Relief Body Wash 10 Ounce	2.99	0	1
12	1/1/2014	1	Anti-Acne Medicated Body Cream 2 Ounce	11.99	0	2

Figure 7-9 – Expensive Beauty Products Input Data – Browse Configuration

Next, let us create flags using regular expressions. We will create a new *Tool Container* called Regular Expressions with a *RegEx* tool in it.

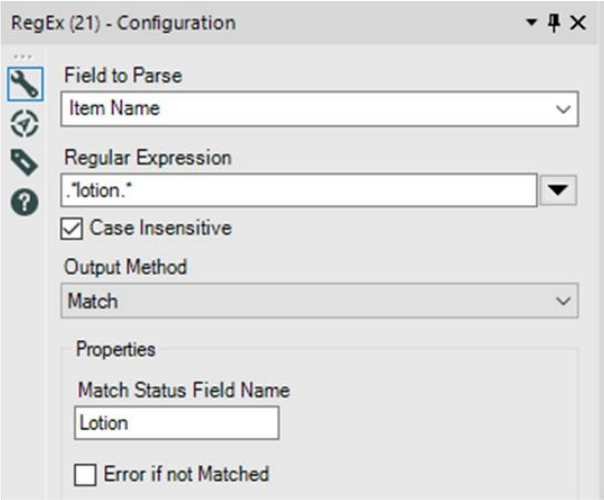


Figure 7-10 – Expensive Beauty Products Regex – "lotion" Flag Configuration

<u>Properties Window</u>:
The *RegEx Configuration* window has five basic components:

- *Field to Parse* is the text field we want to manipulate.

- *Regular Expression* is the expression we use in order to identify sections of the string.
- *Case Insensitive* allows us to specify if the case is important to our search.
- *Output Method* allows us to select what the regular expression will result in.
- *Properties* define specific options depending on which *Output Method* is selected.

The first requirement we want is a flag to identify if *lotion* is in the *Item Name* field. We can use the expression. *lotion. * to identify any field that has the string "lotion" in it.

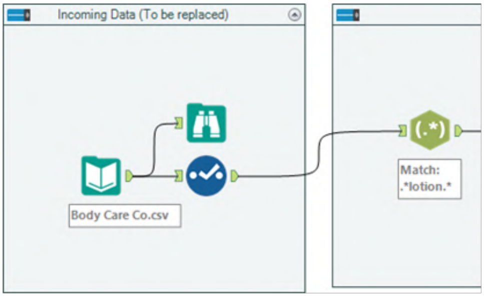


Figure 7-11 – Expensive Beauty Products Data Stream

This regular expression checks to see if the literal pattern "0 or more characters followed by 'lotion' followed by 0 or more characters" is in each record and returning true or false.

A familiarity with Body Care Co's data lets us know their item names always have spaces between words. This means we can actually make our expression more specific as shown in the image, with. *\<moisturizing\>. *

This regular expression reads "0 or more characters followed by the word 'moisturizing' followed by 0 or more characters."

The special characters \< and \> in regular expressions take into account specific characters preceding or ending a word to identify if the entire word is matched.

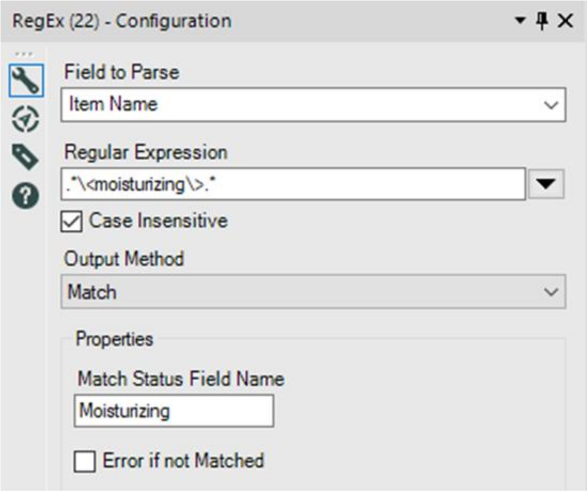


Figure 7-12 – Expensive Beauty Products Regex – "moisturising" Flag Configuration

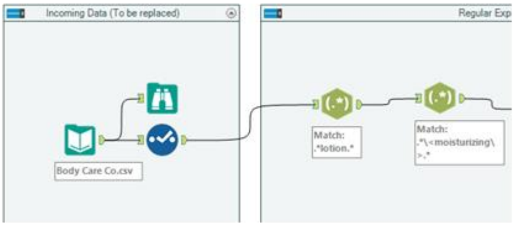


Figure 7-13 – Expensive Beauty Products Data Stream

Following the same pattern, we can create a field called Baby Product.

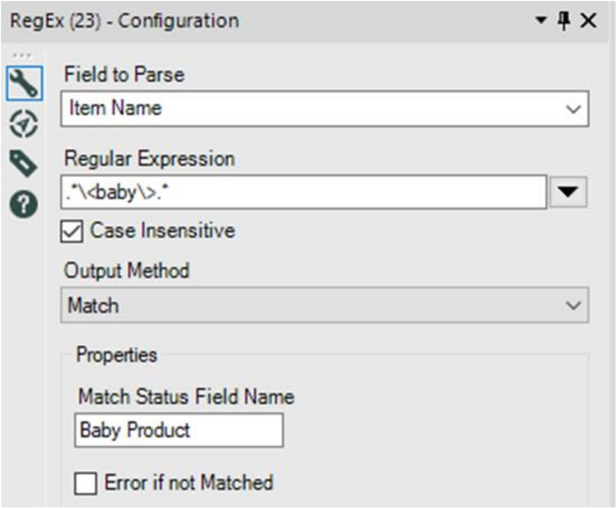


Figure 7-14 – Expensive Beauty Products Regex – "baby" Flag Configuration

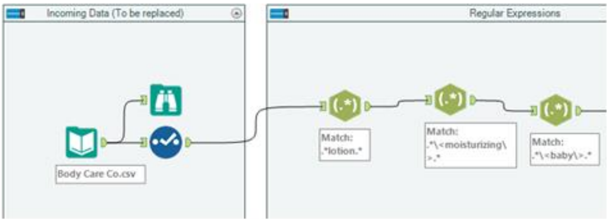


Figure 7-15 – Expensive Beauty Products Data Stream

We then make another for acne products.

Our last flag is more complicated because we have two different things that could define acne products.

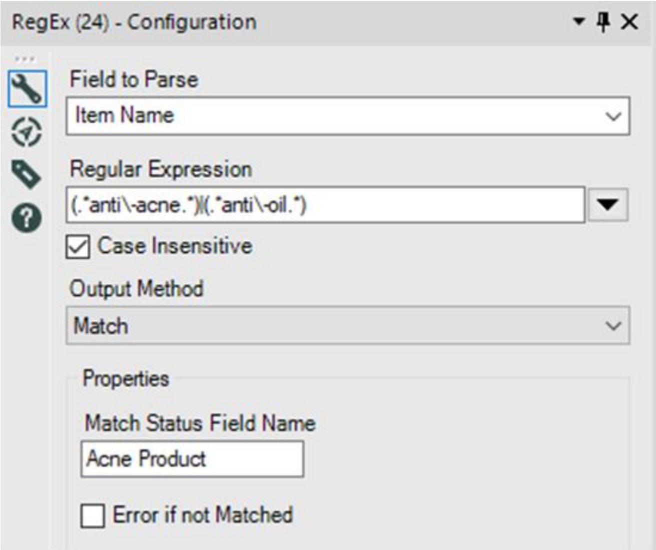


Figure 7-16 – Expensive Beauty Products Regex – a Complex Flag - Configuration

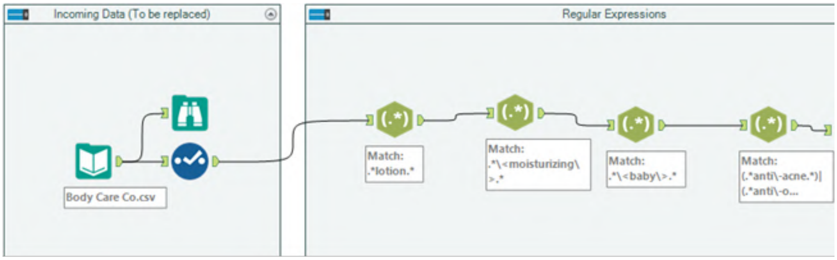


Figure 7-17 – Expensive Beauty Products Data Stream

We will combine what we know about Boolean statements and regular expressions to create the expression

(.*anti\-acne.*) | (.*anti\-oil.*)

This regular expression is checking to see if the pattern "0 or more characters followed by 'anti-acne' followed by 0 or more characters, or

0 or more characters followed by 'anti-oil' followed by 0 or more characters' exists.

As we can see, regular expressions can get complex, but there are always multiple ways to create them. In this case, we could have also used *.*anti\\-(acne|oil).**, where it would have looked for the pattern "0 or more characters followed by 'anti-' followed by 'acne' or 'oil' followed by 0 or more characters".

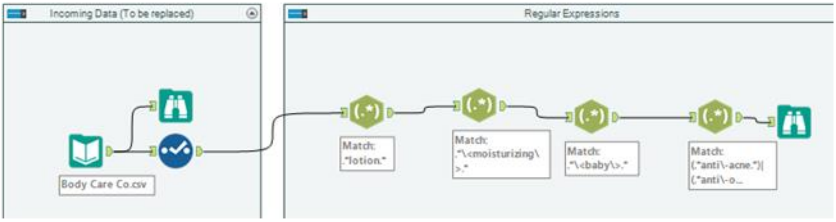
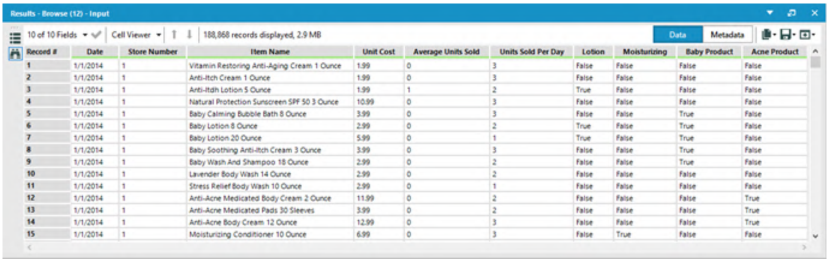


Figure 7-18 – Expensive Beauty Products Data Stream



Record #	Date	Store Number	Item Name	Unit Cost	Average Units Sold	Units Sold Per Day	Lotion	Moisturizing	Baby Product	Acne Product
1	1/1/2014	1	Vitamin Restoring Anti-Aging Cream 1 Ounce	1.99	0	3	False	False	False	False
2	1/1/2014	1	Anti-Itch Cream 1 Ounce	1.99	0	3	False	False	False	False
3	1/1/2014	1	Anti-Itdh Lotion 5 Ounce	1.99	1	2	True	False	False	False
4	1/1/2014	1	Natural Protection Sunscreen SPF 50 3 Ounce	10.99	0	3	False	False	False	False
5	1/1/2014	1	Baby Calming Bubble Bath 8 Ounce	3.99	0	3	False	False	True	False
6	1/1/2014	1	Baby Lotion 8 Ounce	2.99	0	2	True	False	True	False
7	1/1/2014	1	Baby Lotion 20 Ounce	5.99	0	1	True	False	True	False
8	1/1/2014	1	Baby Soothing Anti-Itch Cream 3 Ounce	3.99	0	3	False	False	True	False
9	1/1/2014	1	Baby Wash And Shampoo 18 Ounce	2.99	0	2	False	False	True	False
10	1/1/2014	1	Lavender Body Wash 14 Ounce	2.99	0	2	False	False	False	False
11	1/1/2014	1	Stress Relief Body Wash 10 Ounce	2.99	0	1	False	False	False	False
12	1/1/2014	1	Anti-Acne Medicated Body Cream 2 Ounce	11.99	0	2	False	False	False	True
13	1/1/2014	1	Anti-Acne Medicated Pads 30 Sleeves	3.99	0	2	False	False	False	True
14	1/1/2014	1	Anti-Acne Body Cream 12 Ounce	12.99	0	3	False	False	False	True
15	1/1/2014	1	Moisturizing Conditioner 10 Ounce	6.99	0	3	False	True	False	False

Figure 7-19 – Expensive Beauty Products Boolean Filed (Flags) for RegEx Match

We can see that we have the four Boolean fields to identify each of the four sets of products we have defined.

We can now add a *Comment* tool that describes what is happening in the *Tool Container* and moves onto the fuzzy logic.

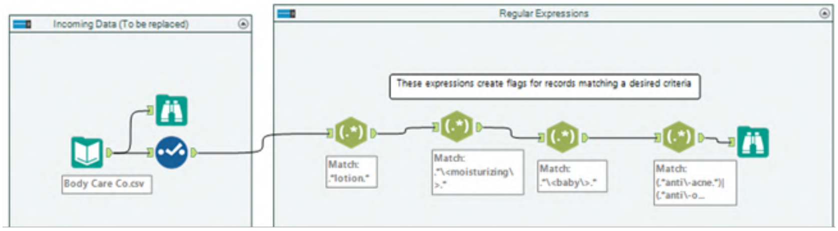


Figure 7-20 – Adding Comments to indicate contents in Tool Container and Updated Data Stream

We need to define the groups of products with similar names. The first thing we should do is to create a unique list of product names. Let us introduce a *Summarize* tool within a container and connect it to data input stream.

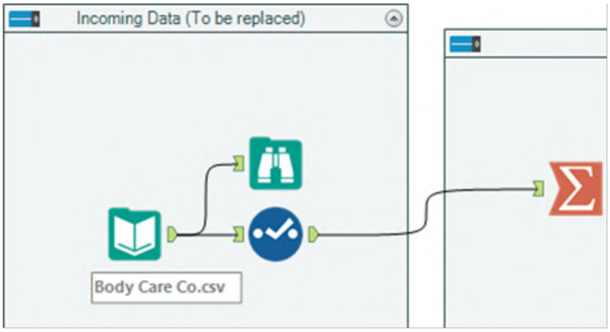


Figure 7-21 – Expensive Beauty Products second Data Stream for fuzzy Match

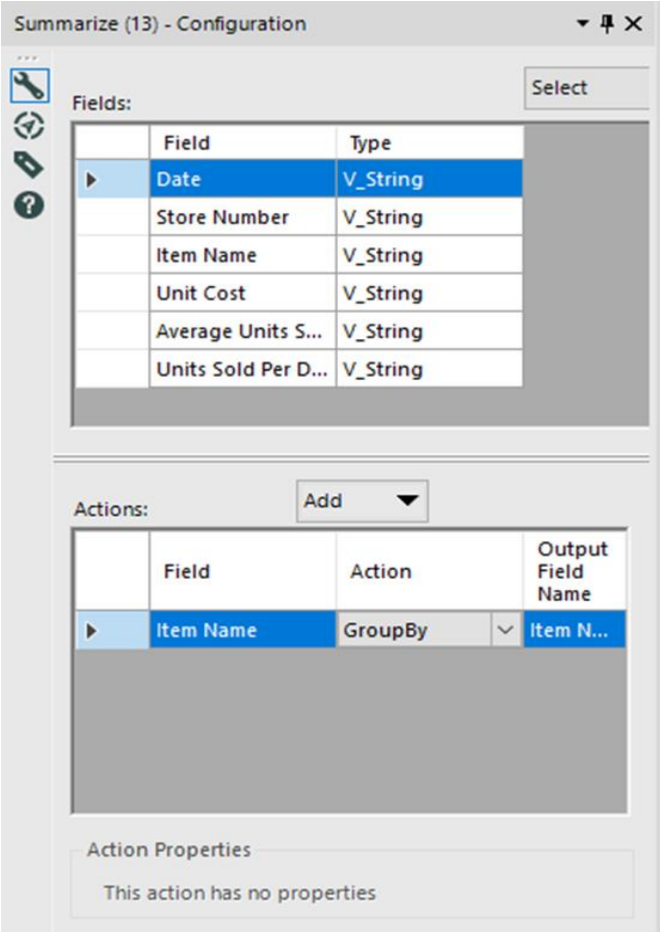
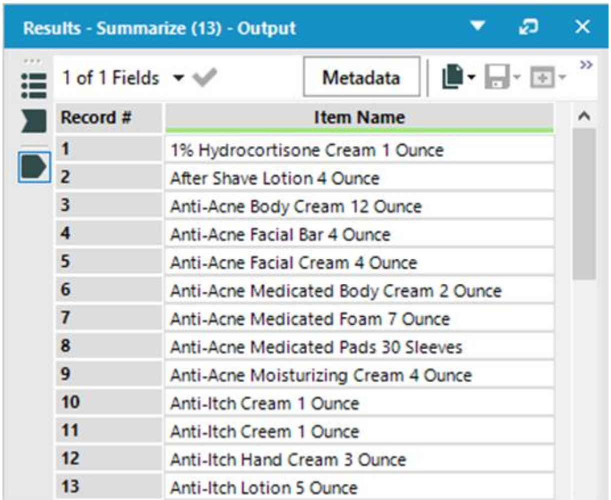


Figure 7-22 – Expensive Beauty Products Summarize tool configuration

Let us run the workflow and take a look at the unique list.



Results - Summarize (13) - Output

1 of 1 Fields Metadata

Record #	Item Name
1	1% Hydrocortisone Cream 1 Ounce
2	After Shave Lotion 4 Ounce
3	Anti-Acne Body Cream 12 Ounce
4	Anti-Acne Facial Bar 4 Ounce
5	Anti-Acne Facial Cream 4 Ounce
6	Anti-Acne Medicated Body Cream 2 Ounce
7	Anti-Acne Medicated Foam 7 Ounce
8	Anti-Acne Medicated Pads 30 Sleeves
9	Anti-Acne Moisturizing Cream 4 Ounce
10	Anti-Itch Cream 1 Ounce
11	Anti-Itch Creem 1 Ounce
12	Anti-Itch Hand Cream 3 Ounce
13	Anti-Itch Lotion 5 Ounce

Figure 7-23 – Expensive Beauty Products Browse Configuration after Summarize tool.

Now that we have a unique list of products, we can determine which of those products have similar names.

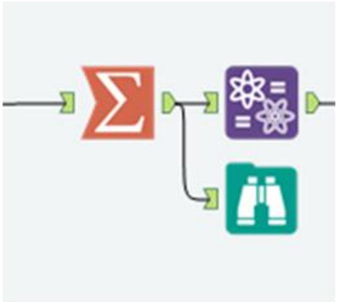


Figure 7-24 – Including Fuzzy Match Tool in Data Stream

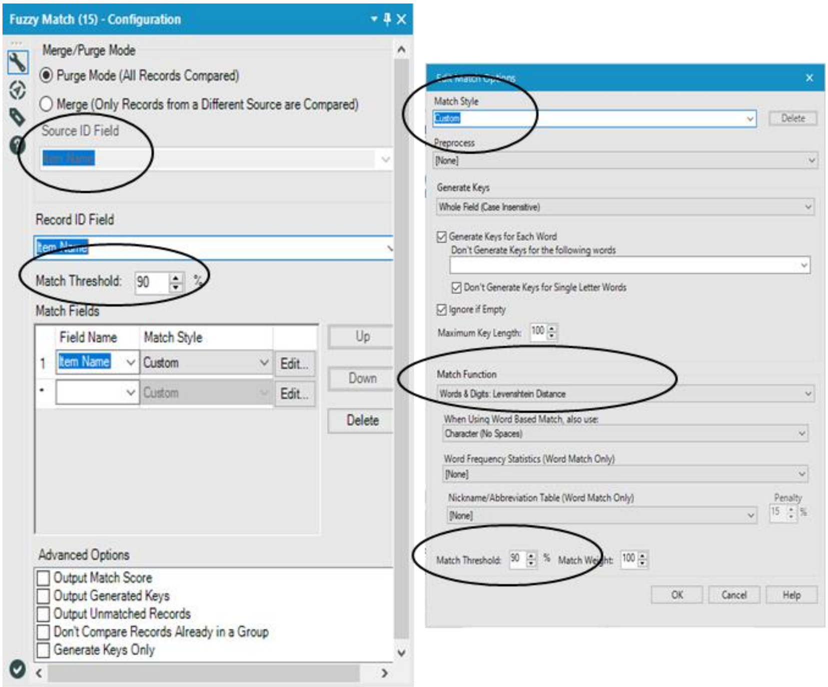


Figure 7-25 – Expensive Beauty Products Fuzzy Match Configuration

<u>Properties Window</u>:
The *Fuzzy Match Configuration* window has the following five core components.

- *Merge/Purge Mode allows us to decide if we need to compare all of the records in the list, or to compare records that have different sources. This means if we have data that is broken into logical buckets, we can compare across those buckets without comparing them.*
- *Record ID Field is the field we will match on.*
- *Match Threshold is the minimum match percentage to have a positive result.*
- *Match Fields allows us to select fields and methodologies to run the analysis for matches.*
- *Advanced Options allows us to modify the output of the Fuzzy Match.*

If we add a Fuzzy Match tool with the above settings, we know we are matching *Item Name* at a minimum 90% threshold. The settings here define a matching algorithm using a word and digit-based *Levenshtein* *distance method*. This algorithm is looking at the whole field by word (except for single-character word; up to 100 words per field) and keeping anything that it finds with at least a 90% match.

Because of the nature of this methodology, we will introduce an issue in the output data stream. That will result in duplicate records for the fields that match based on multiple keys. In order to correct this, we will isolate the unique records to be kept.

If we add a *Unique* tool after the *Fuzzy Match,* we can see a new field has been created called *Item Name2*. The new field allows us to see which entries match.

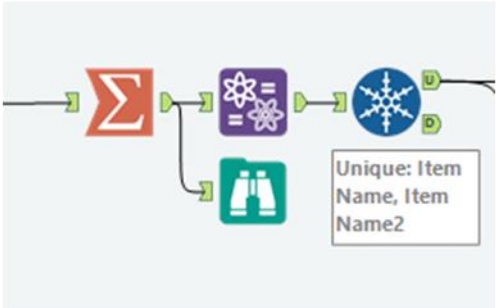


Figure 7-26 – Addition of Unique tool to fuzzy logic data stream

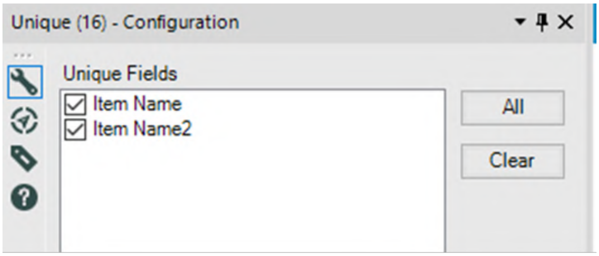


Figure 7-27 – Expensive Beauty Products Unique tool configuration

<u>Properties Window</u>:
The *Unique Configuration* window is a list of each of the fields in the dataset.
We need to select all those we want to use to identify unique records.

A quick run using the *Browse* tool shows us the matched items.

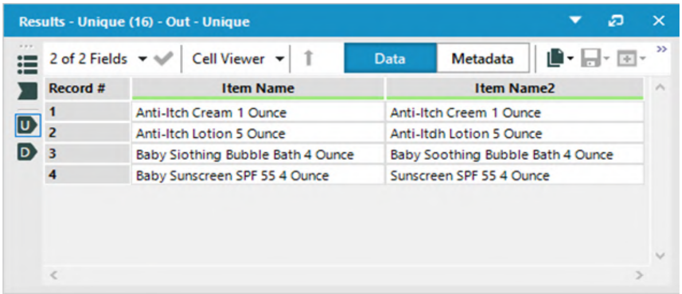


Figure 7-28 – Expensive Beauty Products viewing matching items

We can see by looking at the resulting data stream that we have four groups of items, and three of those are a result of misspellings in the data.

What we can do is provide a numeric grouping for each of these matches, which will solve Jan's problem of data comparison. Since each of the groups is unique, we can use the *Record ID* field to identify them.

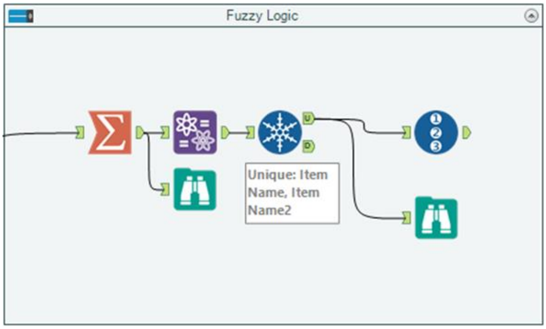


Figure 7-29 – Addition of Record ID tool to fuzzy logic data stream

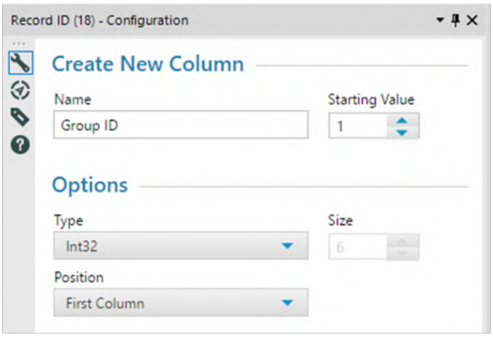


Figure 7-30 – Expensive Beauty Products Record ID Configuration

<u>Properties Window</u>:
The *Record ID Configuration* window has four basic components:

- *Starting Value* is the number we want to assign to the first record (which will be incremented following that record).
- *Field Name* is the resulting field name for our index.
- *Field Type* allows us to select the type and size of the resulting field.
- *Field Position* allows us to set the new field to the beginning or end of the dataset.

Now we can add a comment that will allow Jan to understand what we did, and then we will be ready to combine the data streams.

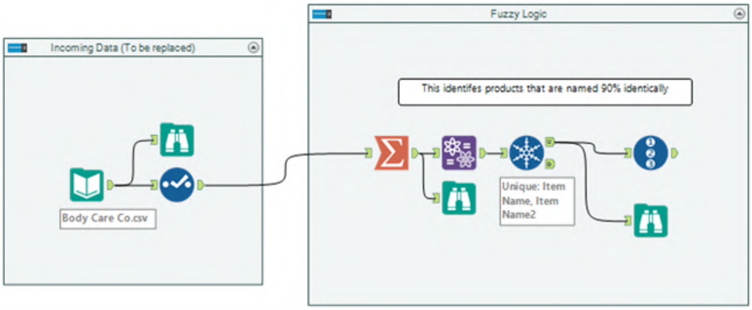


Figure 7-31 – Adding Comments to indicate contents in Tool Container and Updated Fuzzy Logic Stream

In order to combine the data streams, we need to join the data twice so that each part of the group can be flagged with the *Group ID*.

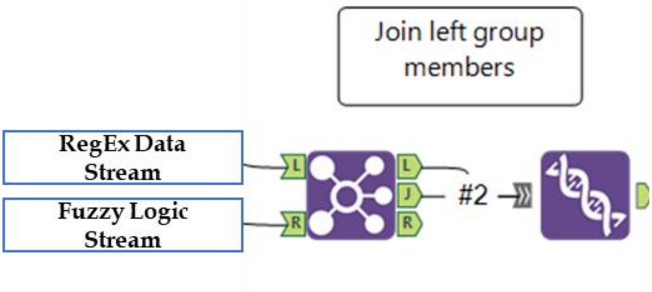


Figure 7-32 – Expensive Beauty Products First Join for Left Group Members

The *Join* tools settings are shown below.

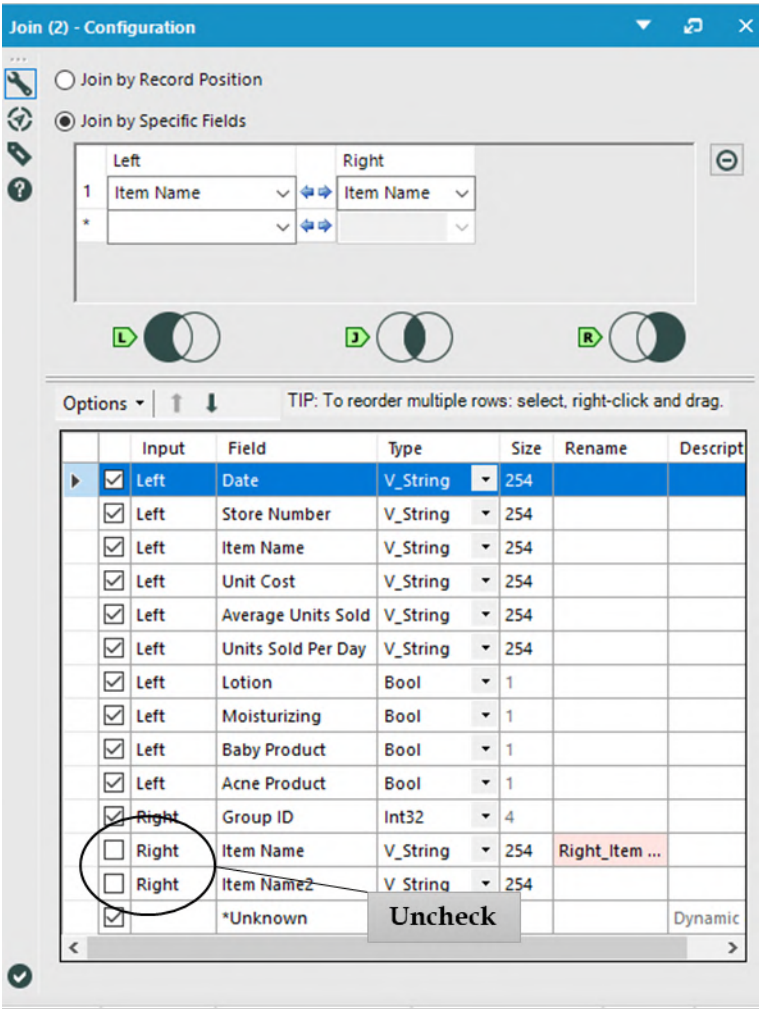


Figure 7-33 – Expensive Beauty Products Join Configuration for "Item Name"

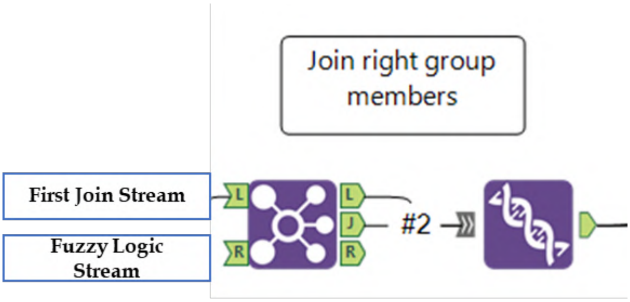


Figure 7-34 – Expensive Beauty Products Second Join for Right Group Members

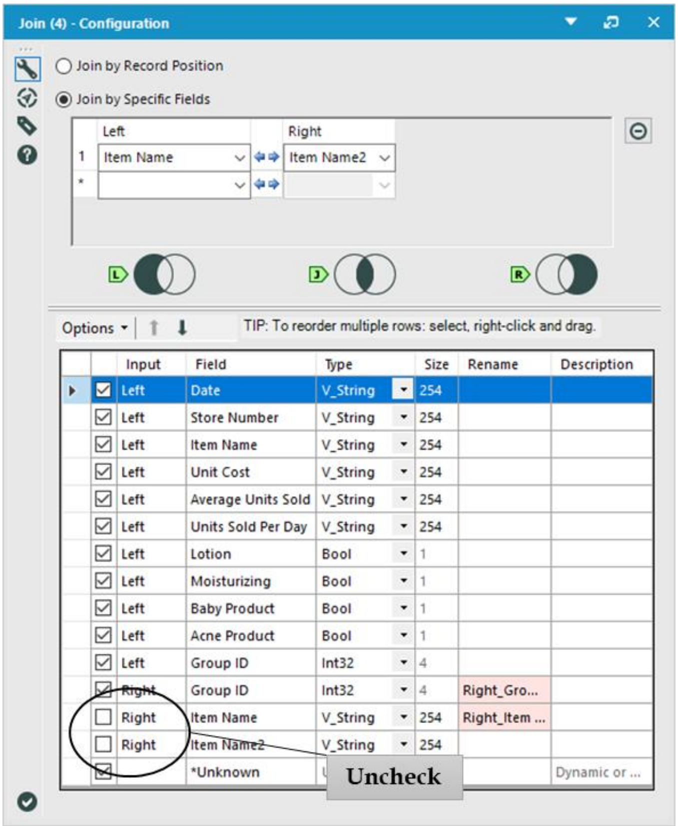
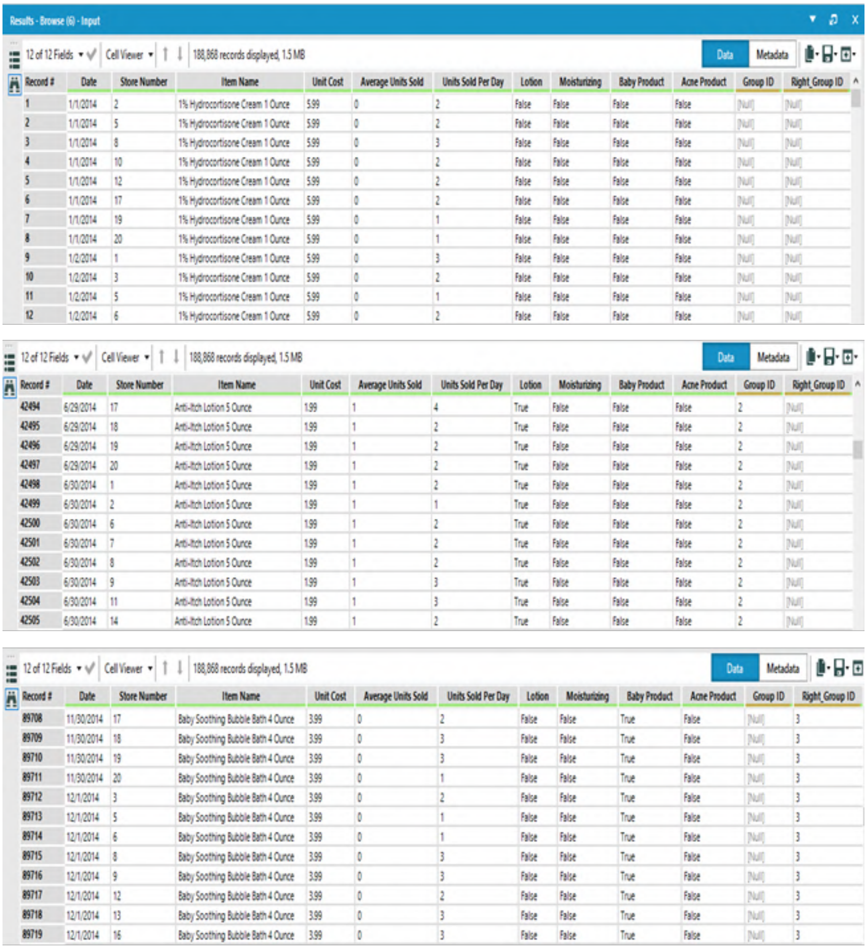


Figure 7-35 – Expensive Beauty Products Join Configuration for "Item Name 2"



Results - Browse (6) - Input

12 of 12 Fields | Cell Viewer | 188,868 records displayed, 1.5 MB | Data | Metadata

Record #	Date	Store Number	Item Name	Unit Cost	Average Units Sold	Units Sold Per Day	Lotion	Moisturizing	Baby Product	Acne Product	Group ID	Right_Group ID
1	1/1/2014	2	1% Hydrocortisone Cream 1 Ounce	5.99	0	2	False	False	False	False	[Null]	[Null]
2	1/1/2014	5	1% Hydrocortisone Cream 1 Ounce	5.99	0	2	False	False	False	False	[Null]	[Null]
3	1/1/2014	8	1% Hydrocortisone Cream 1 Ounce	5.99	0	3	False	False	False	False	[Null]	[Null]
4	1/1/2014	10	1% Hydrocortisone Cream 1 Ounce	5.99	0	2	False	False	False	False	[Null]	[Null]
5	1/1/2014	12	1% Hydrocortisone Cream 1 Ounce	5.99	0	2	False	False	False	False	[Null]	[Null]
6	1/1/2014	17	1% Hydrocortisone Cream 1 Ounce	5.99	0	2	False	False	False	False	[Null]	[Null]
7	1/1/2014	19	1% Hydrocortisone Cream 1 Ounce	5.99	0	1	False	False	False	False	[Null]	[Null]
8	1/1/2014	20	1% Hydrocortisone Cream 1 Ounce	5.99	0	1	False	False	False	False	[Null]	[Null]
9	1/2/2014	1	1% Hydrocortisone Cream 1 Ounce	5.99	0	3	False	False	False	False	[Null]	[Null]
10	1/2/2014	3	1% Hydrocortisone Cream 1 Ounce	5.99	0	2	False	False	False	False	[Null]	[Null]
11	1/2/2014	5	1% Hydrocortisone Cream 1 Ounce	5.99	0	1	False	False	False	False	[Null]	[Null]
12	1/2/2014	6	1% Hydrocortisone Cream 1 Ounce	5.99	0	2	False	False	False	False	[Null]	[Null]

12 of 12 Fields | Cell Viewer | 188,868 records displayed, 1.5 MB | Data | Metadata

Record #	Date	Store Number	Item Name	Unit Cost	Average Units Sold	Units Sold Per Day	Lotion	Moisturizing	Baby Product	Acne Product	Group ID	Right_Group ID
42494	6/29/2014	17	Anti-Itch Lotion 5 Ounce	1.99	1	4	True	False	False	False	2	[Null]
42495	6/29/2014	18	Anti-Itch Lotion 5 Ounce	1.99	1	2	True	False	False	False	2	[Null]
42496	6/29/2014	19	Anti-Itch Lotion 5 Ounce	1.99	1	2	True	False	False	False	2	[Null]
42497	6/29/2014	20	Anti-Itch Lotion 5 Ounce	1.99	1	2	True	False	False	False	2	[Null]
42498	6/30/2014	1	Anti-Itch Lotion 5 Ounce	1.99	1	2	True	False	False	False	2	[Null]
42499	6/30/2014	2	Anti-Itch Lotion 5 Ounce	1.99	1	1	True	False	False	False	2	[Null]
42500	6/30/2014	6	Anti-Itch Lotion 5 Ounce	1.99	1	2	True	False	False	False	2	[Null]
42501	6/30/2014	7	Anti-Itch Lotion 5 Ounce	1.99	1	2	True	False	False	False	2	[Null]
42502	6/30/2014	8	Anti-Itch Lotion 5 Ounce	1.99	1	2	True	False	False	False	2	[Null]
42503	6/30/2014	9	Anti-Itch Lotion 5 Ounce	1.99	1	3	True	False	False	False	2	[Null]
42504	6/30/2014	11	Anti-Itch Lotion 5 Ounce	1.99	1	3	True	False	False	False	2	[Null]
42505	6/30/2014	14	Anti-Itch Lotion 5 Ounce	1.99	1	2	True	False	False	False	2	[Null]

12 of 12 Fields | Cell Viewer | 188,868 records displayed, 1.5 MB | Data | Metadata

Record #	Date	Store Number	Item Name	Unit Cost	Average Units Sold	Units Sold Per Day	Lotion	Moisturizing	Baby Product	Acne Product	Group ID	Right_Group ID
89708	11/30/2014	17	Baby Soothing Bubble Bath 4 Ounce	3.99	0	2	False	False	True	False	[Null]	3
89709	11/30/2014	18	Baby Soothing Bubble Bath 4 Ounce	3.99	0	3	False	False	True	False	[Null]	3
89710	11/30/2014	19	Baby Soothing Bubble Bath 4 Ounce	3.99	0	3	False	False	True	False	[Null]	3
89711	11/30/2014	20	Baby Soothing Bubble Bath 4 Ounce	3.99	0	1	False	False	True	False	[Null]	3
89712	12/1/2014	3	Baby Soothing Bubble Bath 4 Ounce	3.99	0	2	False	False	True	False	[Null]	3
89713	12/1/2014	5	Baby Soothing Bubble Bath 4 Ounce	3.99	0	1	False	False	True	False	[Null]	3
89714	12/1/2014	6	Baby Soothing Bubble Bath 4 Ounce	3.99	0	1	False	False	True	False	[Null]	3
89715	12/1/2014	8	Baby Soothing Bubble Bath 4 Ounce	3.99	0	3	False	False	True	False	[Null]	3
89716	12/1/2014	9	Baby Soothing Bubble Bath 4 Ounce	3.99	0	3	False	False	True	False	[Null]	3
89717	12/1/2014	12	Baby Soothing Bubble Bath 4 Ounce	3.99	0	2	False	False	True	False	[Null]	3
89718	12/1/2014	13	Baby Soothing Bubble Bath 4 Ounce	3.99	0	3	False	False	True	False	[Null]	3
89719	12/1/2014	16	Baby Soothing Bubble Bath 4 Ounce	3.99	0	3	False	False	True	False	[Null]	3

Figure 7-36 – Expensive Beauty Products Browse Configuration after Joins

If we applied the previous settings along with the default *Union* settings, we should see that we have two new fields at the end of the data stream. We need to combine these fields in order to give Jan a simple dataset to work from.

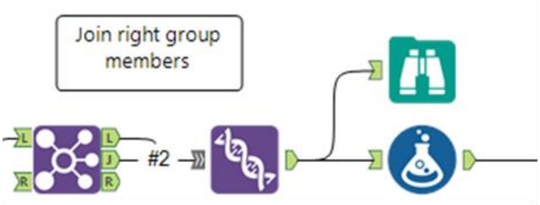


Figure 7-37 – Inclusion of Formula tool in data stream

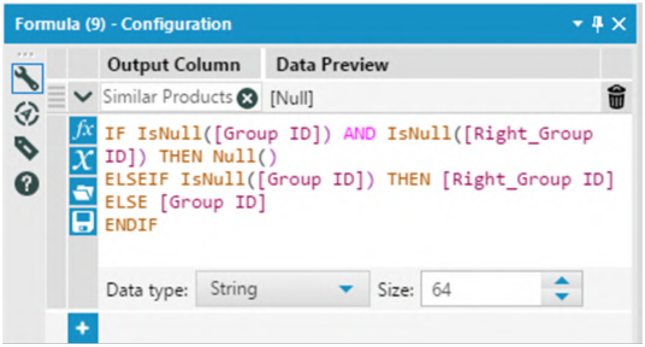


Figure 7-38 – Expensive Beauty Products Formula tool configuration

We can create a conditional formula called *Similar Products* that will bring the fields together:

IF IsNull([Group ID]) AND IsNull([Right_Group ID]) THEN Null()

ELSEIF IsNull([Group ID]) THEN [Right_Group ID]

ELSE [Group ID]

ENDIF

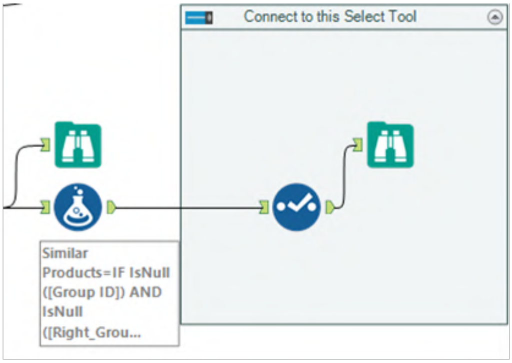


Figure 7-39 – Select tool included for final output

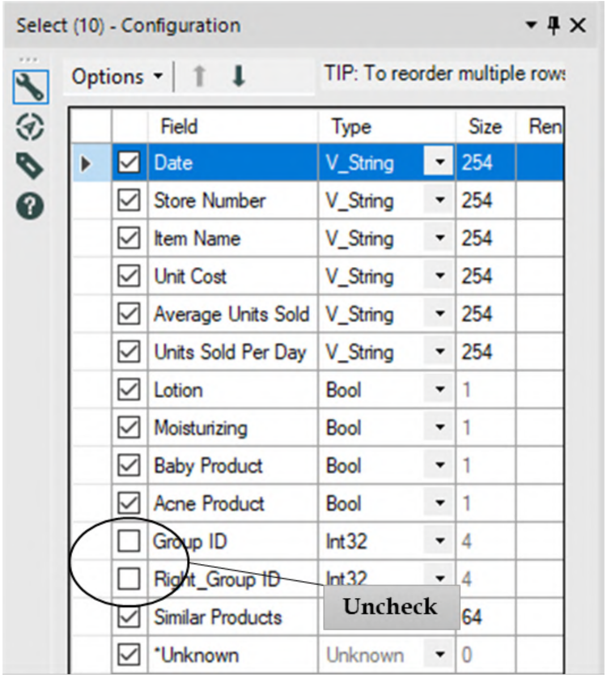


Figure 7-40 – Expensive Beauty Product output Select Configuration

In order to finish this data stream, we can add a select statement that has the two *Group ID* fields removed. If we then use a *Tool* container, we can very clearly show where any downstream tools should be connected.

Record #	Date	St...	Item Name	Unit...	A...	Un...	Lotion	Mo...	Baby...	Acn...	Similar Products
1	1/1/2014	2	1% Hydrocortisone Cream 1 Ounce	5.99	0	2	False	False	False	False	[Null]
2	1/1/2014	5	1% Hydrocortisone Cream 1 Ounce	5.99	0	2	False	False	False	False	[Null]
3	1/1/2014	8	1% Hydrocortisone Cream 1 Ounce	5.99	0	3	False	False	False	False	[Null]
4	1/1/2014	10	1% Hydrocortisone Cream 1 Ounce	5.99	0	2	False	False	False	False	[Null]
5	1/1/2014	12	1% Hydrocortisone Cream 1 Ounce	5.99	0	2	False	False	False	False	[Null]
6	1/1/2014	17	1% Hydrocortisone Cream 1 Ounce	5.99	0	2	False	False	False	False	[Null]

Figure-7-41-Expensive Beauty Product Output

The *Expensive Beauty Products* data stream should look like the following image when it is complete.

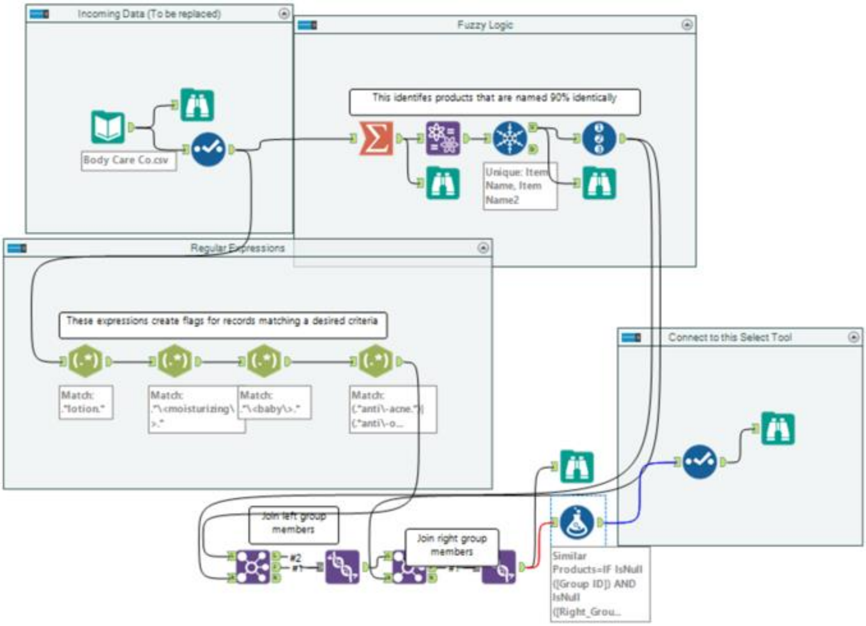


Figure-7-42-Expensive Beauty Product Stream on Completion

7.3 More Flags

To... Alteryx Consultants

Subject More Flags

Hey,

Check out the response we got!

Think you can handle it?

____

Hey,

That last dataset you created was great. I have a few more items that I need to add to the analysis.

Can you add a flag for the following?

1) All sun protective products (includes "SPF" in the description)
2) All itch relief (includes "anti-itch", "anti-itdh" (I noticed an issue with the data), or "hydrocortisone" in the description)
3) All shampoos (includes "shampoo" in the description)
4) All conditioners (includes "conditioner" in the description)

Thanks again,
Jan

CHAPTER 8

Applications Wanted

To... Alteryx Consultants

Subject Let's Build An App For That!

Attached All Recorded Traffic Tickets.csv

Hey,

I need to pull you in on a project with the Baltimore Ticket Team.

They have asked us if we can build an application that will allow them to do the following:

1. select a file with violations
2. select a date range to limit the data
3. have it create output file(s) of their choice
    a. in-state plates
    b. out-of-state plates
    c. in-state and out-of-state plates (as a single or multiple files).

I think we will hear back from them soon if we build it to these exact standards. We should design this analytic application such that the default is to select Maryland, yet it also gives the flexibility to select any state. Also include a text box so they can limit to a specific location of interest.

Should be fun.

8.1 Tools & Concepts

8.1.1 Action

Concept- Updates values of development tools with the values from the interface questions at runtime

 **Figure-8-1-Action**	**Tool Palette: Interface**
	Modifies the values in other tools. For more details use the link below. bit.ly/2KiRhWf

8.1.2 Condition

Concept- Tests for the presence of user selections

 **Figure-8-2-Condition**	**Tool Palette: Interface**
	Allows us to trigger different results depending on the condition in the tool. For more details use the link below. bit.ly/2I2yn56

8.1.3 Control Parameter

Concept- A Control Parameter input for a batch macro

 **Figure-8-3-Control Parameter**	**Tool Palette: Interface**
	Is the input for each iteration of a Batch Macro. For more details use the link below. bit.ly/2ra6tNd

8.1.4 Date

Concept- A calendar in app

 **Figure-8-4-Date**	**Tool Palette: Interface**
	Allows for date selection from a calendar menu. For more details use the link below. bit.ly/2JFndmo

8.1.5 Drop Down

Concept- A single selection list in an app

8.1.5 Drop Down

Concept- A single selection list in an app

 Figure-8-5-Drop Down	**Tool Palette: Interface**
	Allows for the selection of a single value from a provided list. For more details use the link below. bit.ly/2vZ3Dj1

8.1.6 Error Message

Concept- Displays an error message

 Figure-8-6-Error Message	**Tool Palette: Interface**
	Throws an Error message to the end user of an app or macro. Once the error message is thrown, all downstream processing stops. For more details use the link below. bit.ly/2JETLgo

8.1.7 File Browse

Concept- File browse control in an app

 Figure-8-7-File Browse	**Tool Palette: Interface**
	Allows connecting to a file of choice instead of a predesignated file. For more details use the link below. bit.ly/2rcxnUL

8.1.8 Folder Browse

Concept- Folder browse control in an app

 Figure-8-8-Folder Browse	**Tool Palette: Interface**
	Displays a folder browse control in an app or macro. The directory path specified by the user is passed to downstream tools. For more details use the link below. bit.ly/2FsO56J

8.1.9 List Box

Concept- A multi-selection check box list

<table>
<tr><td rowspan="2">
Figure-8-9-List Box</td><td>Tool Palette: Interface</td></tr>
<tr><td>Displays a multi-selection check box list in an app or macro. The selections specified by the user are passed as values to downstream tools.
For more details use the link below.
bit.ly/2raCWSJ</td></tr>
</table>

8.1.10 Radio Button

Concept- A mutually exclusive option in an app

<table>
<tr><td rowspan="2">
Figure-8-10-Radio Button</td><td>Tool Palette: Interface</td></tr>
<tr><td>Creates a single select option for use in the Interface designer.
For more details use the link below.
bit.ly/2w1OKwt</td></tr>
</table>

8.1.11 Text Box

Concept- A free form text box in an app

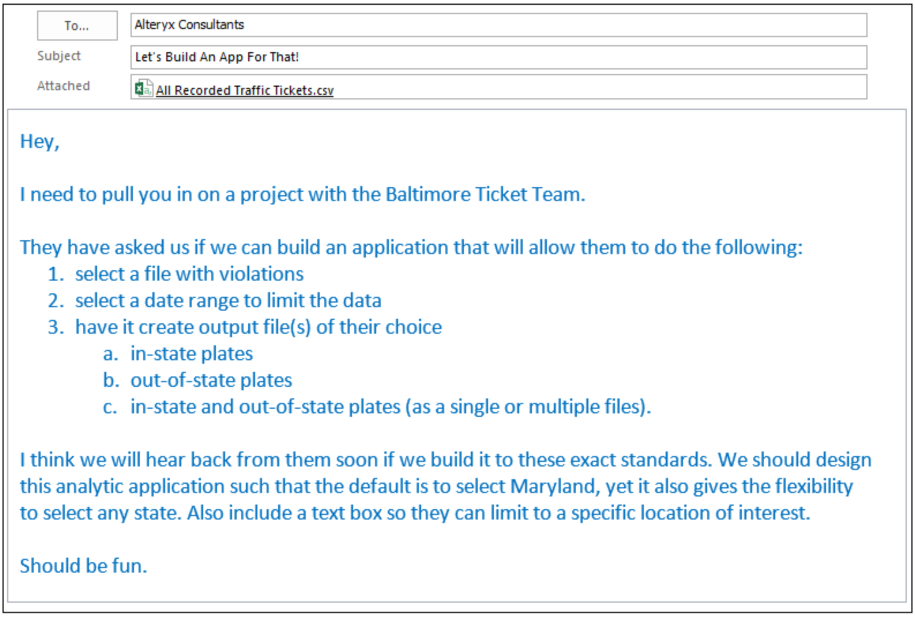
 **Figure-8-11-Text Box**	**Tool Palette: Interface**
	Allows the user to enter a string of choice. For more details use the link below. For more details use the link below. bit.ly/2vZHwcp

8.1.12 Tree

Concept- An organized, hierarchical data structure in an app

 **Figure-8-12-Tree**	**Tool Palette: Interface**
	Displays an organized, hierarchical data structure in an app or macro. The selections made by the user are passed as values to downstream tools. The values returned from trees are separated by a new line character (\n). For more details use the link below. bit.ly/2JEvB5r

8.2 Let's Build an App For That!



To... Alteryx Consultants
Subject Let's Build An App For That!
Attached All Recorded Traffic Tickets.csv

Hey,

I need to pull you in on a project with the Baltimore Ticket Team.

They have asked us if we can build an application that will allow them to do the following:

1. select a file with violations
2. select a date range to limit the data
3. have it create output file(s) of their choice
    a. in-state plates
    b. out-of-state plates
    c. in-state and out-of-state plates (as a single or multiple files).

I think we will hear back from them soon if we build it to these exact standards. We should design this analytic application such that the default is to select Maryland, yet it also gives the flexibility to select any state. Also include a text box so they can limit to a specific location of interest.

Should be fun.

Alteryx allows us to develop Analytic Applications which is a workflow with a user interface. Apps have a couple of special properties that allow us to:

- Use an interface to run the workflow instead of opening the file in Alteryx Designer.
- Publish the workflow to an Alteryx Gallery, giving users without Alteryx Designer the ability to run workflows.

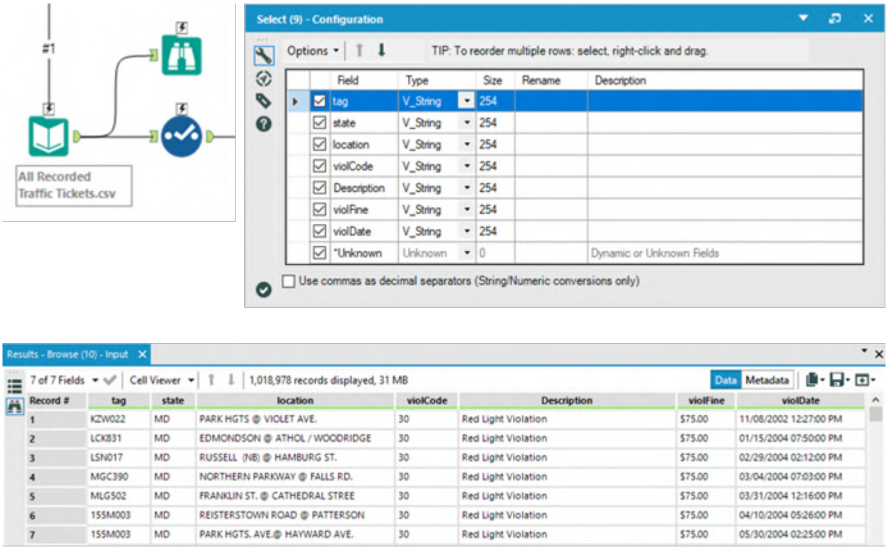


Figure-8-13-Traffic Tickets-Opening, Selecting and Viewing Data

We are going to approach this by laying out the workflow and then adding the interface tools that allow for the modification. So, let us get started by connecting to the file *Chapter 8- All Recorded Traffic Tickets.csv*. Since they haven't asked us to do anything with the Fine amount field, we can leave it in a string format.

However, since we know that we need to limit the data by dates, we need *violDate* in a date format. If we take a look at the format of *violDate*, we see it is a string formatted date with an *AM/PM* flag. If we look at the *DateTime* tool, we can see that we do not have a matching date format, so we need to use a formula tool to manipulate the string first.

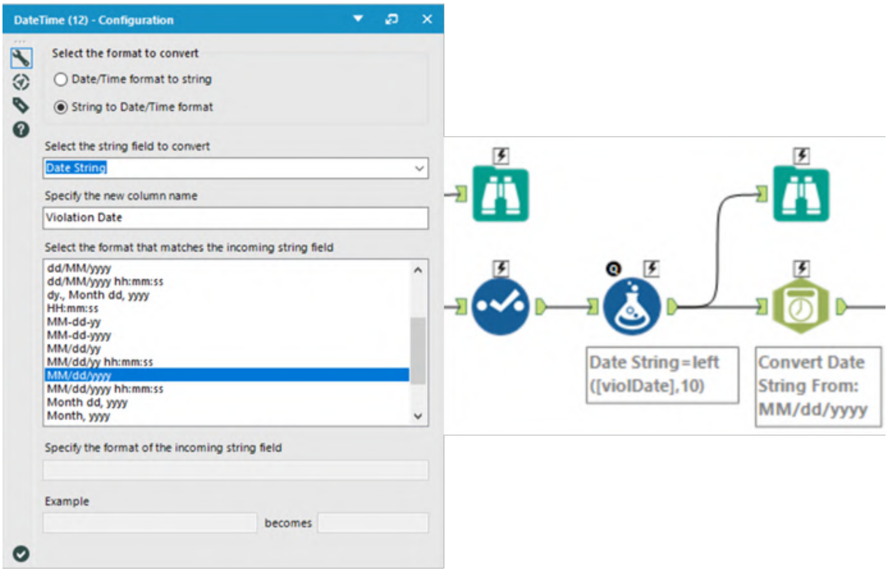


Figure-8-14-Traffic Tickets-Conversion from String to Date

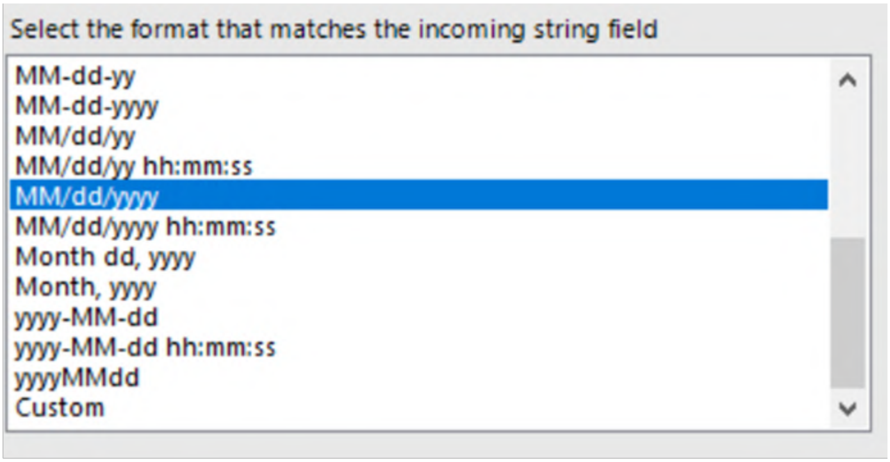


Figure-8-15-Traffic Tickets-Input String Format

Since we are only going to allow the end user to select the dates, we can drop the time portion of the field. This means that we only need the first 10 characters. The formula is: *LEFT([violDate],10)*

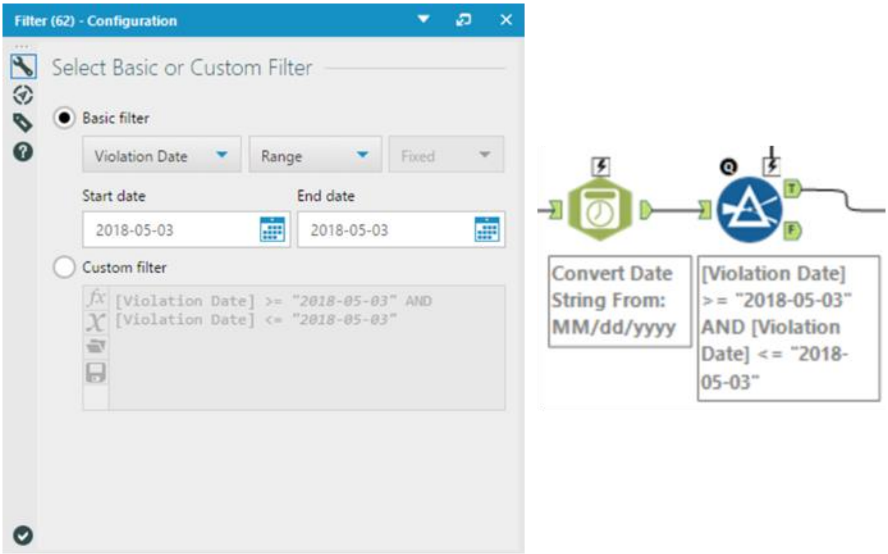


Figure-8-16-Traffic Tickets-Using Filter Tool

Now that we have the field in date format, we will use a *Filter* tool to take in the end user's inputs of start date and end date to get the date range. We select the Basic filter option and select Range to allow for the start date and end date inputs. As we are going to allow the end user the ability to select the beginning and end date, it does not matter that both dates are referring to today because we are going to replace these values.

Note: While we are testing, it may make it hard to see what is going on if we do not select a date range that is reflected in the data.

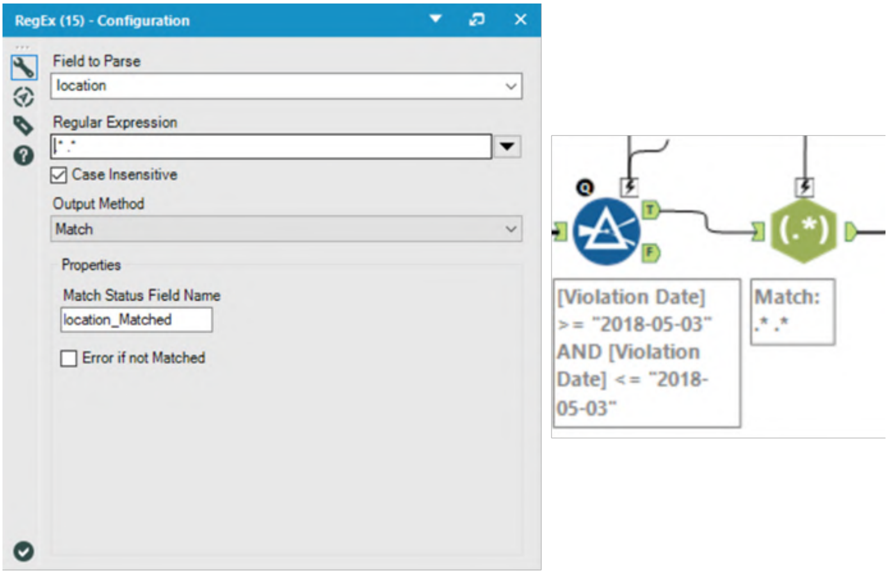


Figure-8-17-Traffic Tickets-Using RegEx to Flag String

Now we can add the feature they didn't ask for. We can write a regular expression that allows us to capture partial or full location names to limit the data. We want to look at the location field and then create a regular expression that will tell us if the field matches the user input. We can use the following expression along with a *text box interface* tool to do this.

*.*something.**

Using the above expression, we will update *something* with the user-specified value so we can flag the appropriate records. (If the user wants everything, they can leave the field blank resulting in the *expression* .*.*, which will mark all fields as true.)

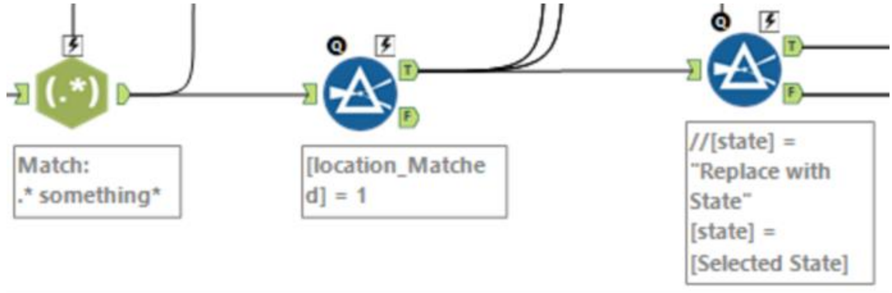


Figure-8-18-Traffic Tickets-Matched with Flagged String

Now that we have flagged which records match the desired location, we can use the *location_Matched* field that was generated in the *RegEx* tool to filter the data. If we use the basic filter for the *location_Matched,* we can see the field is true and the function becomes the expression *location_Matched* = 1.

Now that the data is limited to the appropriate locations, we can flag which state they may want to analyze in detail. For now, we can use a custom filter expression in order to finish building the core data stream.

[state] = "Replace with State"

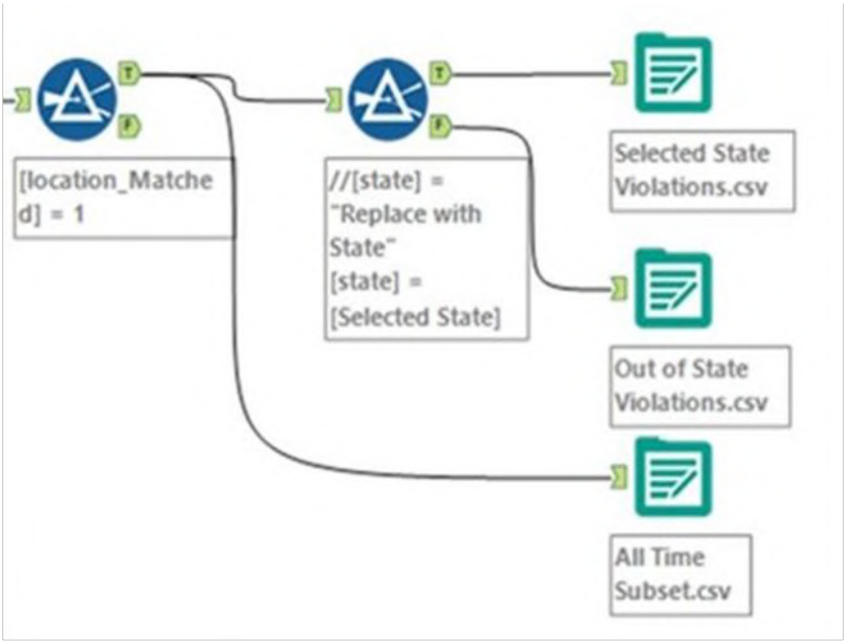


Figure-8-19-Traffic Tickets-Getting Three Possible Outputs

The last step in the workflow is to create the three possible outputs. We can add them to the end of the data stream where appropriate.

- Connect the *Selected State Violations.csv* to the True output on the state *filter*.
- Connect the *Out of State Violations.csv* to the False output on the state *filter*.
- Connect the *All Time Subset.csv* to the True output on the Location Match *filter*.

We can put the three outputs in *Tool Containers*. This is not for organizational purposes like we saw earlier. When we are building applications, we can enable and disable *Tool Containers* based on the user selection. In this case, we are going to disable all three tool containers so the only time the output is created is when we have selected that option.

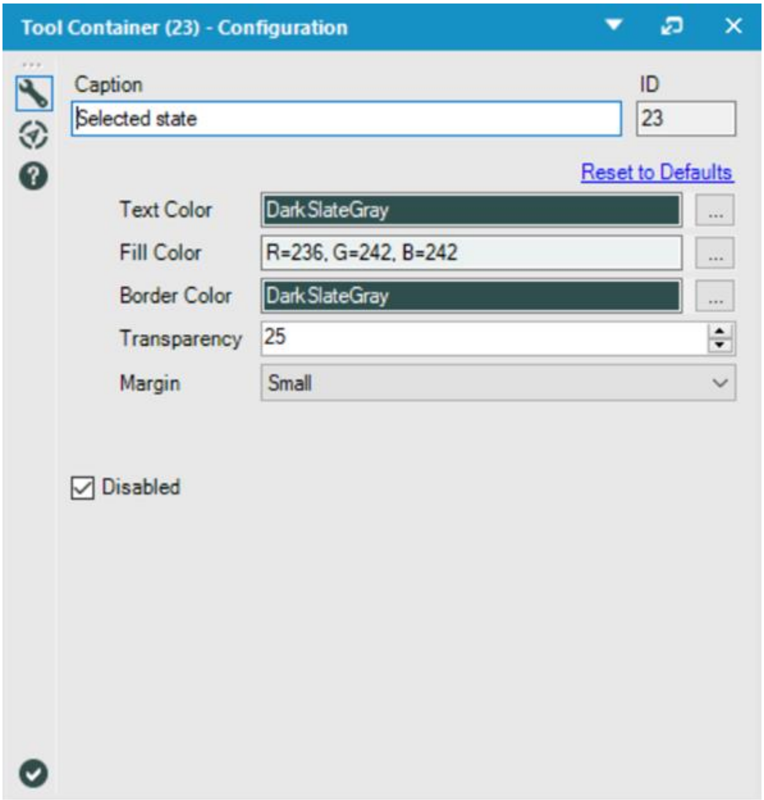


Figure-8-20-Traffic Tickets-Use Tool Container to disable it except when appropriate option is selected

We can disable the containers using the *Disabled* option in the bottom-left corner of the properties windows. This option only shows the container and not the contents and will turn off all the three outputs.

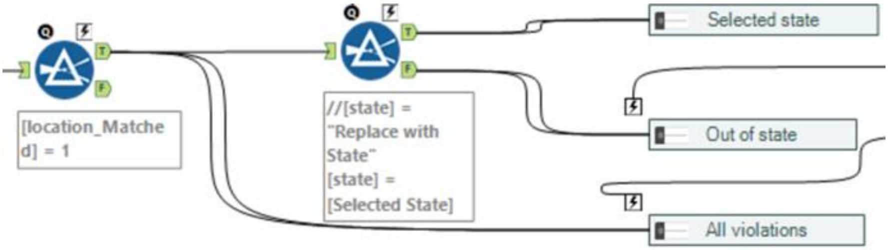


Figure-8-21-Traffic Tickets-Selected Options

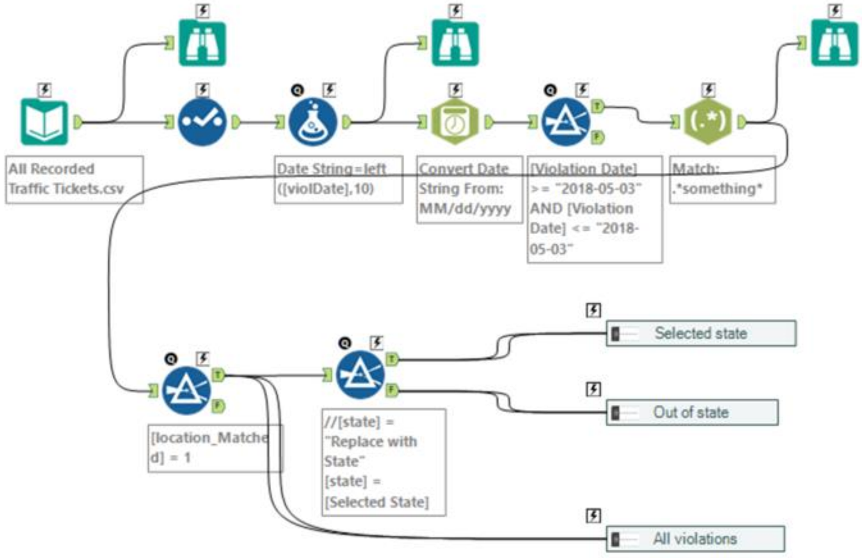


Figure-8-22-Traffic Tickets-Basic Data Stream

We now have the basic data stream designed for the analytic app. We will have to modify some of the tools, but almost everything is ready for the transition. Let us look at the list of interactions we are providing the Baltimore Ticket Team.

1. Select files.

We now have the basic data stream designed for the analytic app. We will have to modify some of the tools, but almost everything is ready for the transition. Let us look at the list of interactions we are providing the Baltimore Ticket Team.

1. Select files.
2. Choose the date range.
3. Name a location (optional).
4. Select a State (Maryland, by default).
5. Select which combination of the three output files is to be returned.

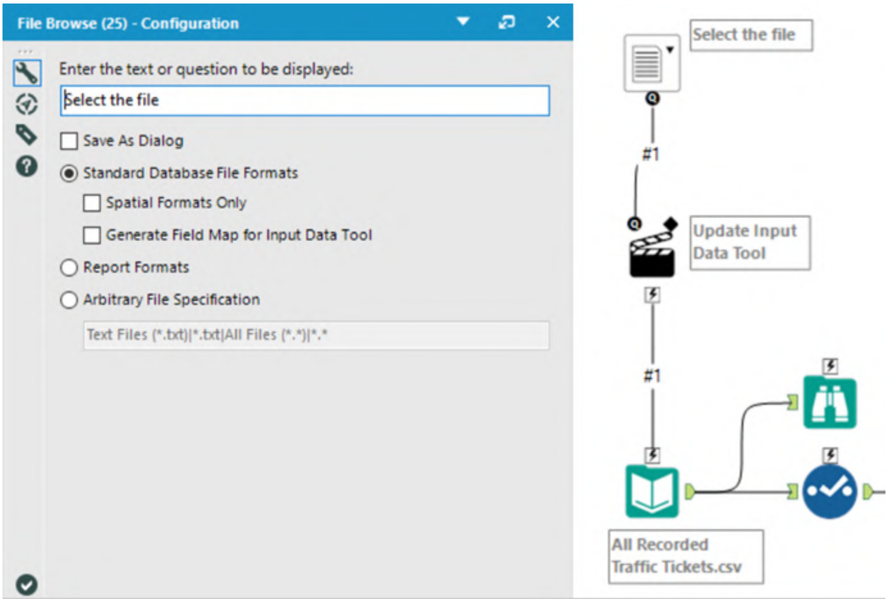


Figure-8-23-Traffic Tickets-File Browse

The first question we can address is which file should be selected. If we drop a *File Browse* tool onto the canvas, we can see that the tools show their *lightning bolt* and *question mark* anchors indicating we are now working with an *Analytic Application or Macro*. We can then change the text so it reads *Select the file* and use the remaining default settings.

We know we want to connect the file selection question to the *Input Data* tool; however, the *lightning bolt anchor* only connects to *Action* tools. We could bring in the tool, but Alteryx has a clever way of handling it: When we connect a question directly to a *lightning bolt anchor,* it brings an *Action* tool onto the canvas with all of the default settings in place. In this case, it will do exactly what we want, and we can move onto the next question. (Click the Action Tool On and Off to confirm the settings.)

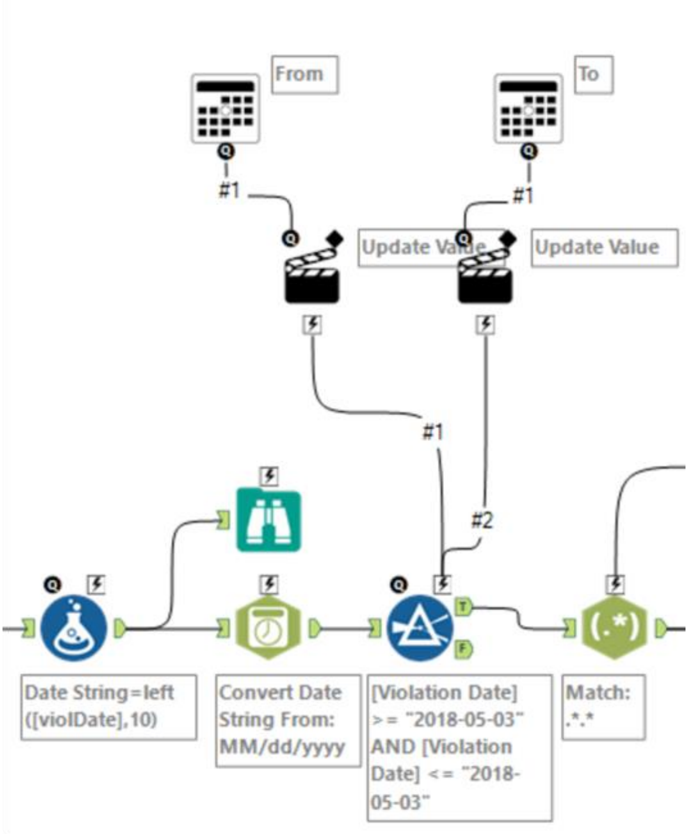


Figure-8-24-Traffic Tickets-Using Action Tool for Date

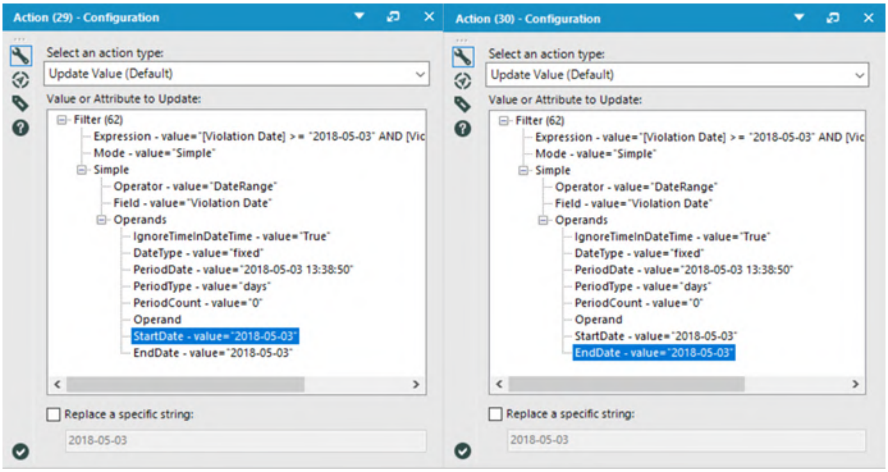


Figure-8-25-Traffic Tickets-Configuration of From and To's Action Tool

The next question is slightly more complicated. We need to set the beginning and end dates for the selected timeframe. We can drag in two *Date* tools and label one *From* and the other *To* so we know which dates refer to the beginning and end of the range. This time, when we connect the questions to the *Filter* tool, there is no indication what value should be updated. What we want to do is update two values in the *Filter*. However, since there are multiple inputs, we need to select which value each action should update.

- From's *Action* tool should update *StartDate – value='2018-05-03'*
- To's *Action* tool should update *EndDate – value='2018-05-03'*

Now that we have this filter getting its values updated, we can move onto the next question.

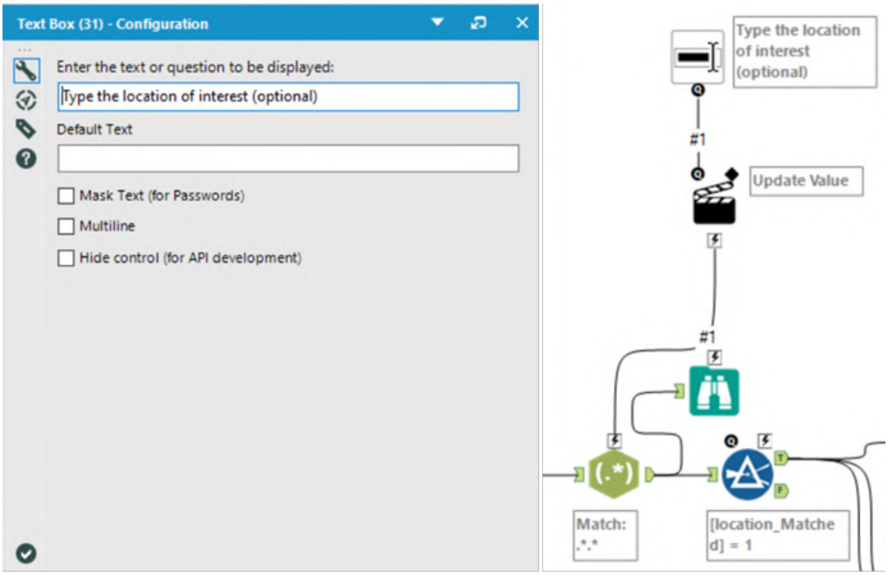


Figure-8-26-Traffic Tickets-Using Text Box to specify Location

This time, we are going to add a *Text Box* tool with the prompt *Type the location of interest (optional)*. This will allow the end user to type anything they want to limit the locations required. We want this tool to modify the regular expression in the *RegEx* tool.

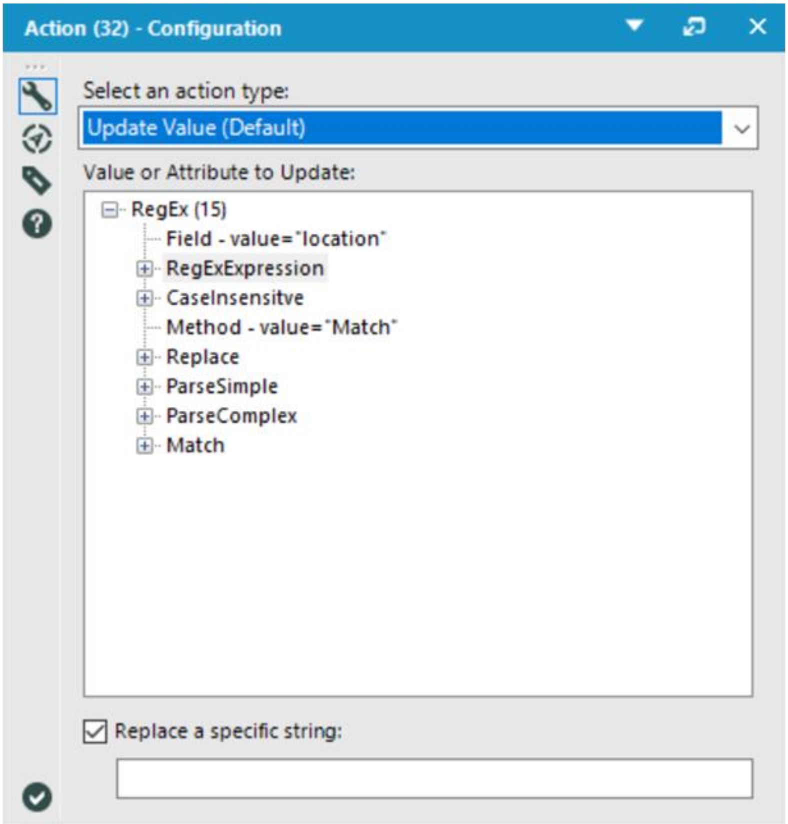


Figure-8-27-Traffic Tickets-Using Action with RegEx to specify String

We can see the *Action* tool is giving us an error, and if we take a look, we need to specify what should be updated. In this case, we know that in the *RegEx* tool, there is the expression string .*.* We want to replace this with the value the end user types in. Thus, we can use the *Replace a specific string:* option to replace the string .*.* within the regular expression. Now let's see how we can modify the state filter.

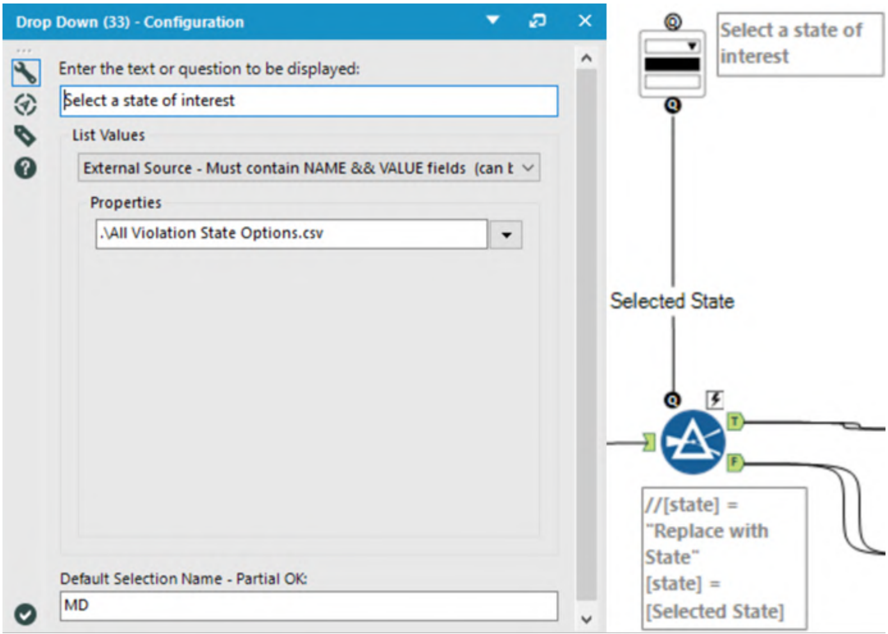


Figure-8-28-Traffic Tickets-Using Drop Down for selecting a State of Interest

This time, we are going to feed the answer of a question directly into a tool for use in the expression. We are going to create a *Drop-Down* question with the prompt *Select a State of Interest*. We are going to use the second file we received from the Baltimore Ticket Team to import the list of possible state codes from a file called *All Violation State Options.csv*. Finally, we are going to set the default value to *MD* so the team can quickly run the in-state analysis.

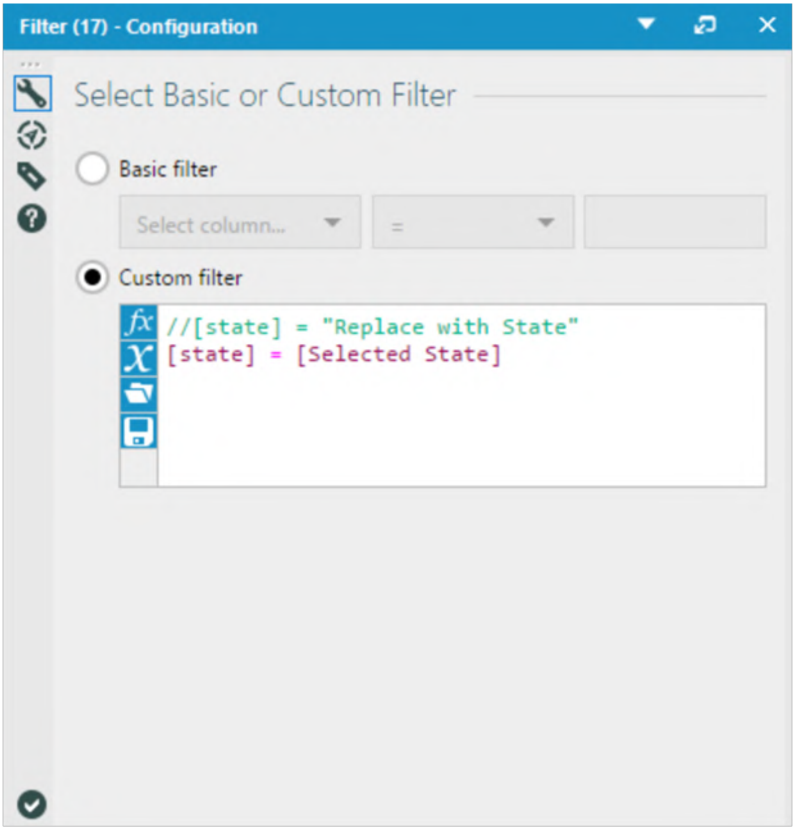


Figure-8-29-Traffic Tickets-Using Formula to filter the Data

We are almost done with this question, except that we need to update the formula used to filter the data. We can replace the original formula with the one in the expression mentioned. (If we do not rename the connection, it should say [#1] instead of [Selected State].)

There is only one more question, but this one is the most complicated. This last question involves selecting which output(s) get created. Since the end user may want to output any combination of three different files (we will force them to select at least one), there are five different options we need to provide.

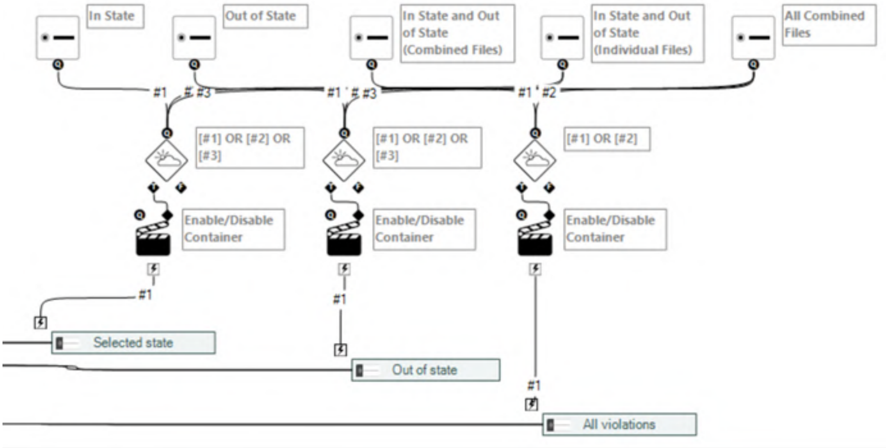


Figure-8-30-Traffic Tickets-Choosing from 5 different options

Create five Radio Button tools with the default settings other than the labels:

1. In State
2. Out of State
3. In State and Out of State (Combined Files)
4. In State and Out of State (Individual Files)
5. All Combined Files

The goal here is to develop a structure where anytime we want a particular file created, we enable the *Tool Container(s)*. In order to activate the tool containers, we need *Action* tools connected to them that enable the container if the incoming connection is true.

Since multiple values can result in each file getting produced, we have complex conditions that need to be met. In order to evaluate these conditions, we need to add a *Condition* tool before each *Action* tool.

Each of these Condition tools should be connected in the following way:

1. Condition connected to the Selected State Output
   - In State File (connection #1)
   - In State and Out of State (Individual Files) (connection #2)
   - All Combined Files (connection #3)
2. Condition connected to the Non-Selected State Output
   - Out of State File (connection #1)
   - In State and Out of State (Individual Files) (connection #2)
   - All Combined Files (connection #3)
3. Condition connected to the All Violations Output
   - In State and Out of State (Combined Files) (connection #1)
   - All Combined Files (connection #2)

The expressions in each of the three conditions should be:

1. [#1] OR [#2] OR [#3]
2. [#1] OR [#2] OR [#3]
3. [#1] OR [#2]

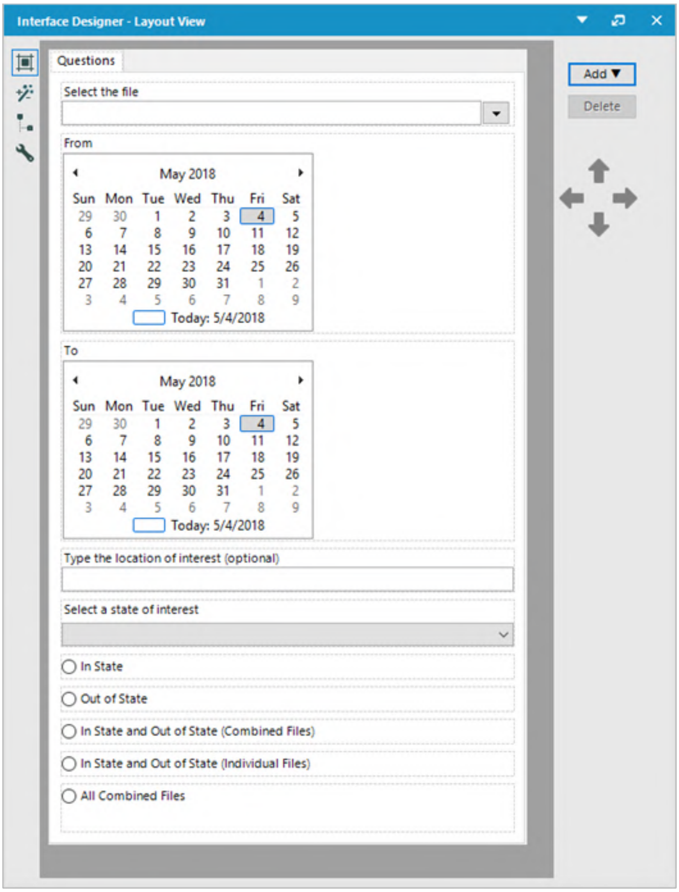


Figure-8-31-Traffic Tickets-Interface Designer

Now that we have the workflow developed, we can see the *Interface Design*. The interface is how the end users are going to interact with the workflow. As we can see from the previous page, the *Interface Designer* shows us the list of questions we created in the workflow. We could leave the questions like this, yet it provides a better user experience if we organize the tools.

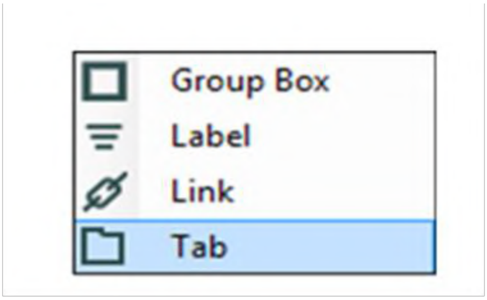


Figure-8-32-Traffic Tickets-Choosing Tab from the Drop Down

We can create a new tab by clicking *add drop down* and selecting *Tab*. We can then click on questions we want to move in order to have related questions together.

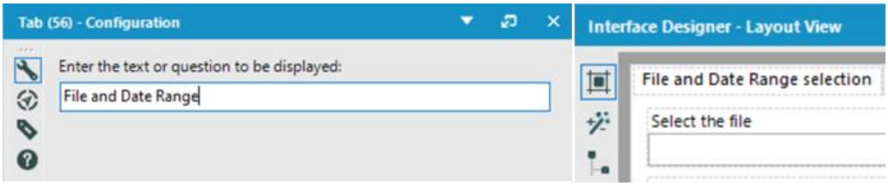


Figure-8-33-Traffic Tickets-Entering Text to be displayed

We can rename this tab by clicking on the tab at the top of the list and changing the text in the *Tab configuration window*.

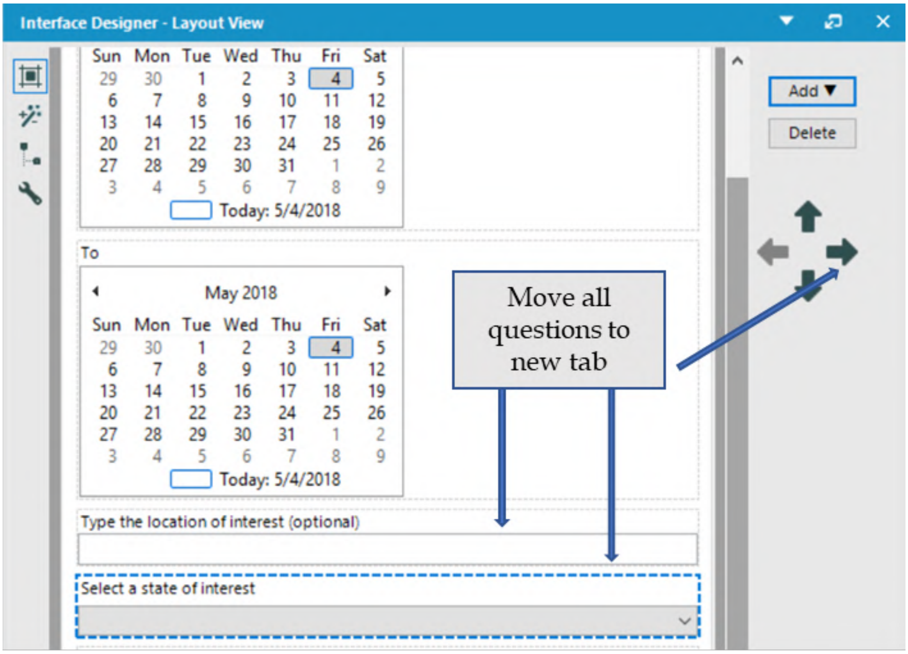


Figure-8-34-Traffic Tickets-Location Information

We can then add a new tab and rename it *Location Information*. Once it is created, we can move the questions in the previous image to that tab that is about Location (of the driver's home state or violation).

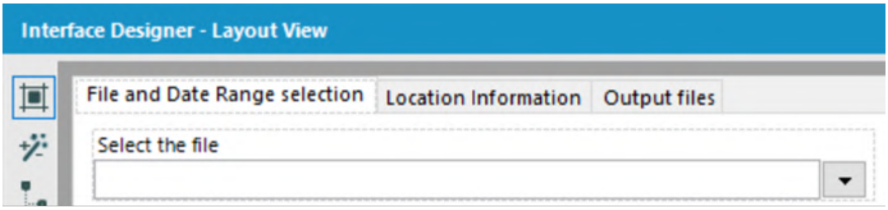


Figure-8-35-Traffic Tickets-Adding a tab called Output

Now we can create one more tab called *Output*. But before we move questions, we need to add a *Group Box* from the *Add* menu.

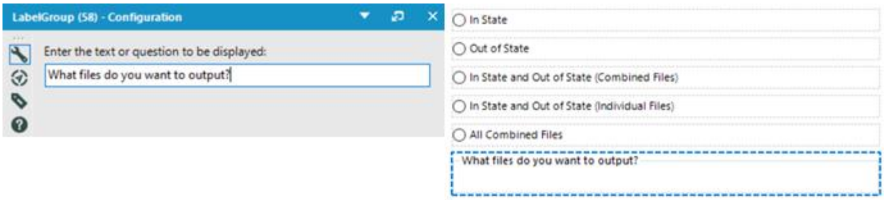


Figure-8-36-Traffic Tickets-Moving all questions into a group using LabelGroup

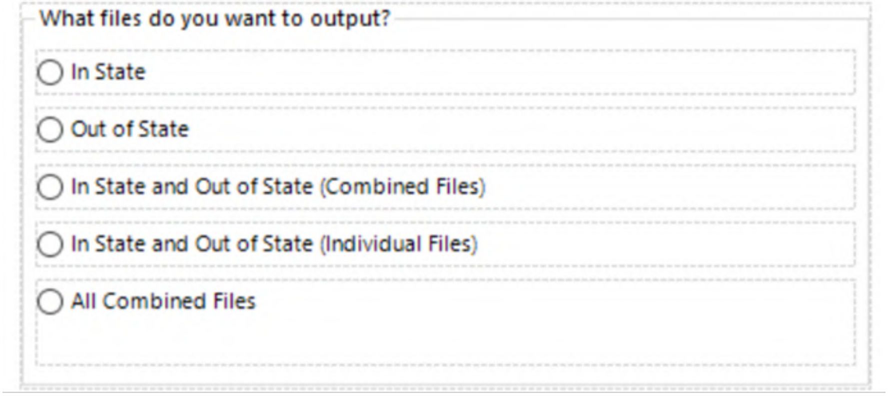


Figure-8-37-Traffic Tickets-Grouped questions moved to Output tab

Now we will move all the required questions down into the *Group Box* by selecting the question and clicking down until our list looks like the image.

Once we have all of our output questions grouped together, it is easy to move them across into the Output tab. Then we can click on the *Group Box* title and move that.

After completion, we should have the three tabs for our questions.

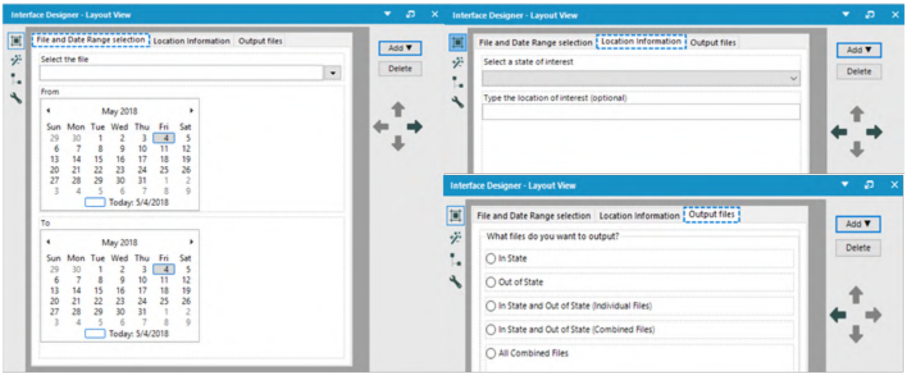


Figure-8-38-Traffic Tickets-All Three Tabs

For better representation purposes, some of the connections are made wireless. The final workflow after adding the Interface tools looks as shown in the image.

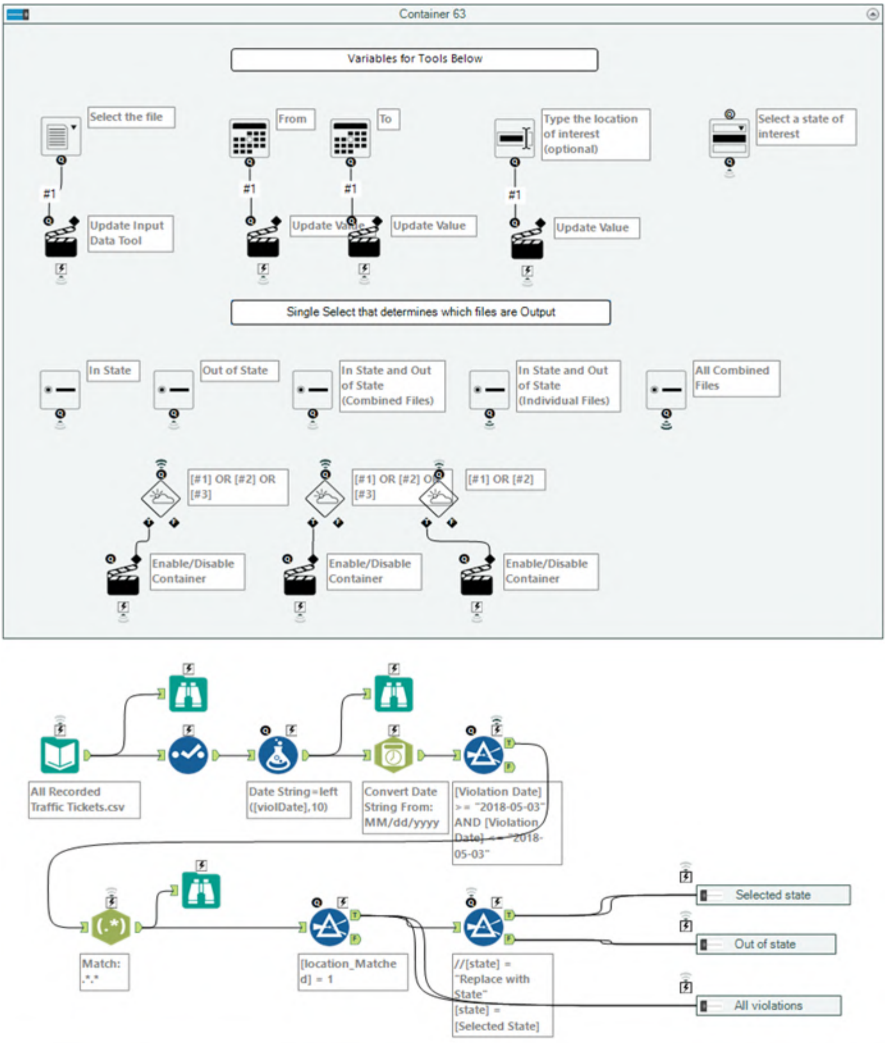


Figure-8-39-Traffic Tickets-Data Stream when complete

8.3 To Summarize or Not to Summarize: That is the Question

To... Alteryx Consultants

Subject To Summarize, or Not to Summarize, That is the Question

Hey,

As it turns out, the Baltimore Ticket Team really likes what we put together but would like to add one more feature.

They want to keep everything that is currently there, plus add an option to summarize the output files by state to see the total count of violations.

Let me know when you've come up with a solution.

Thanks.

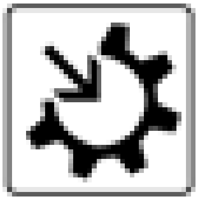
CHAPTER 9

Where’s the Joe

9.1 Tools & Concepts

9.1.1 Macro Input

Concept- Incoming Data connection for a Macro

 **Figure-9-1-Macro Input**	**Tool Palette: Interface**
	Allows for the creation of an incoming data connection for a macro. For more details use the link below. bit.ly/2HmG2d7

9.1.2 Macro Output

Concept- Outgoing Data Connection for a Macro

 **Figure-9-2-Macro Output**	**Tool Palette: Interface**
	Allows us to create an outgoing data connector for a macro. For more details use the link below. bit.ly/2JpapR5

9.1.3 Map

Concept- Working with spatial objects

 **Figure-9-3-Map**	**Tool Palette: Interface**
	Allows us to create a spatial object(s) on a map for use in the data stream. For more details use the link below. bit.ly/2HrPkso

9.1.4 Map Input

Concept- Work with Map as a Source

 **Figure-9-4-Map Input**	**Tool Palette: In/Out**
	Allows us to use the map as a source to create spatial objects. For more details use the link below. bit.ly/2rngVRG

9.1.5 Numeric Up Down

Concept – To input a numeric value

<table>
<tr><td rowspan="2">
**Figure-9-5-
Numeric Up Down**</td><td>**Tool Palette: Interface**</td></tr>
<tr><td>Allows us to input a numeric value to be used in the data stream
For more details use the link below.
bit.ly/2HZSUXK</td></tr>
</table>

9.1.6 Spatial Match

Concept – To filter spatial fields

<table>
<tr><td rowspan="2">
**Figure-9-6-
Spatial Match**</td><td>**Tool Palette: Spatial**</td></tr>
<tr><td>Allows us to use one spatial field to filter the data from another.
For more details use the link below.
bit.ly/2vGwNn1</td></tr>
</table>

9.1.7 Trade Area

Concept – To draw a Polygon

<table>
<tr><td rowspan="2">
**Figure-9-7-
Trade Area**</td><td>**Tool Palette: Spatial**</td></tr>
<tr><td>Draws a polygon centered on a point object that is provided.
For more details use the link below
bit.ly/2JoKHvW</td></tr>
</table>

9.1.8 Create Point

Concept – To create a point type spatial object

 **Figure-9-8-Create Points**	**Tool Palette: Spatial**
	Converts decimal latitudes and longitude columns into a new field called *Centroid* with a point object for each record. For more details use the link below. bit.ly/2vNYf1U

9.1.9 Distance

Concept – calculate drive distance between spatial objects

 **Figure-9-9-Distance**	**Tool Palette: Spatial**
	Allows us to input a numeric value to be used in the data stream For more details use the link below. bit.ly/2ImFmbB

9.1.10 Find Nearest

Concept – To identify shortest distance

 **Figure-9-10-Find Nearest**	**Tool Palette: Spatial**
	Allows us identify the shortest distance between spatial objects in one file and the objects in a second file. For more details use the link below. bit.ly/2JpYx1a

9.1.11 Spatial Info

Concept – To extract tabular information for a spatial object

 **Figure-9-11-Spatial Info**	**Tool Palette: Spatial**
	Extracts tabular information about the spatial object. Attributes such as: area, spatial object, number of parts, number of points, and centroid Latitude /Longitude coordinates can be revealed. For more details use the link below. bit.ly/2FpuJPN

9.1.12 CASS

Concept – cleans up the data by correcting address information

 **Figure-9-12- CASS**	**Tool Palette: Address**
	Takes the input address file and checks it against the USPS Coding Accuracy Support System. It also appends valuable address-specific information to the data. For more details use the link below. bit.ly/2JBMvSa

9.1.13 Parse Address

Concept – Breaks down street address in to components

 **Figure-9-13- Parse Address**	**Tool Palette: Address**
	Breaks down a street address into its component parts, such as street number, directional (S, NW, and so on), street name, and suffix (ST, RD, BLVD). For more details use the link below. bit.ly/2HwgmPs

9.1.14 Street Geocoder

Concept – links geographic coordinates with input addresses

<table>
<tr><td rowspan="2">
Figure-9-14-Sreet Geocoder</td><td>**Tool Palette: Address**</td></tr>
<tr><td>Places a point object based on address interpolation by converting a multi-line address into a normalized form, with latitude and longitude, spatial object, and additional fields specific to the coding process.
For more details use the link below.
bit.ly/2Jx0etu</td></tr>
</table>

9.1.15 US ZIP9 Coder

Concept – Appends latitude, longitude, county and State

<table>
<tr><td rowspan="2">
Figure-9-15-US ZIP9 Coder</td><td>**Tool Palette: Address**</td></tr>
<tr><td>Associates geographic coordinates with input ZIP9 (also known as ZIP+4) codes in an address file, enabling the user to carry out geography-based analyses
For more details use the link below.
bit.ly/2HyAjoW</td></tr>
</table>

9.2 Where's the Joe?

To... Alterxy Consultants

Subject Where's the Joe?

Attached Mermaid Coffee Company Locations.csv

Hey,

We were asked by The Mermaid Coffee company to create a map of all of their locations in the United States for an advertising campaign.

They have provided us a map of all of their global locations to work with.

They want to see the location as well as a set radius around all of their stores, which should be configurable from 1 mile to 10 miles, depending on the user's preferences.

They want a report to be created with a map of all U.S. locations drawn within a selected area, as well as the individual maps for each state for stores in that selected area.

This is a good time to introduce you to a concept that will save you a lot of time down the road. Many requests have similar core elements, and it saves a lot of time when you can reuse them. Thus, Alteryx allows you to make macros.

We should develop a macro that will allow you to pass map inputs as well as a numeric field for the radius before we tackle this project.

Thanks.

Macros are tools we develop so processes we need to reuse in the same workflow or others don't need to be replicated. This is useful for multiple reasons: The first is it will save us time in creating workflows, and the second is it makes maintenance much easier by allowing us to make a change to a single macro instead of each instance in all files that use the macro.

Let's start by building a macro that is given a data stream with latitude and longitude field. *Macros* are a special type of Analytic Application, so we will be taking the same approach in designing it as we do with apps.

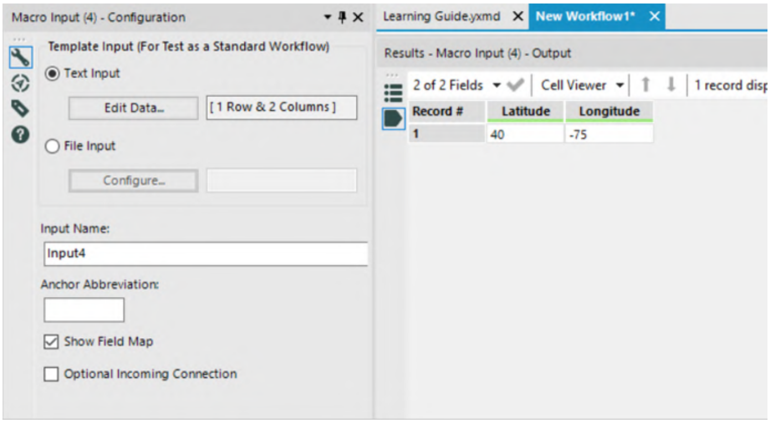


Figure-9-16-Mermaid Coffee Macro Input for location

We are going to build this macro by starting with a *Macro Input* tool. We are going to create a sample data using the *Text Input* option and make sure that *Show Field Map* is checked. We are setting up the data this way because we need to make sure we have latitude and longitude fields in any data that uses the tool. In the event, the incoming data doesn't have fields named *Latitude* and *Longitude* we are giving the person using this macro the ability to map the fields accordingly. First, connect to the dataset *Chapter 9- Mermaid Coffee Company Locations.csv.*

4

Now for the first time, we pay attention to the **Unknown* option in the select tool. Because we are building a macro, we don’t know what data will be coming into this tool; this option allows us to pass through any unknown fields in the data stream without modifying them.

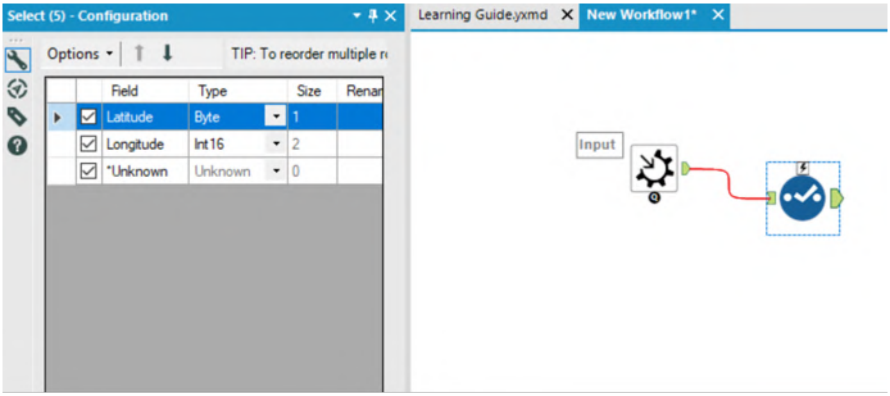


Figure-9-17-Mermaid Coffee using select to get the data

Now for the first time, we pay attention to the **Unknown* option in the select tool. Because we are building a macro, we don’t know what data will be coming into this tool; this option allows us to pass through any unknown fields in the data stream without modifying them

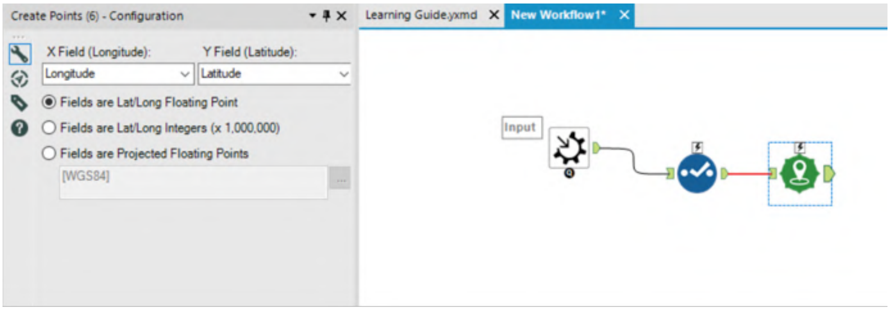


Figure-9-18-Mermaid Coffee creating points using default settings

Since we have named the latitude and longitude fields in a way that the *Create Points* tool recognizes, it will auto populate with those fields in the appropriate places in the settings. We are going to use the default settings to create our points.

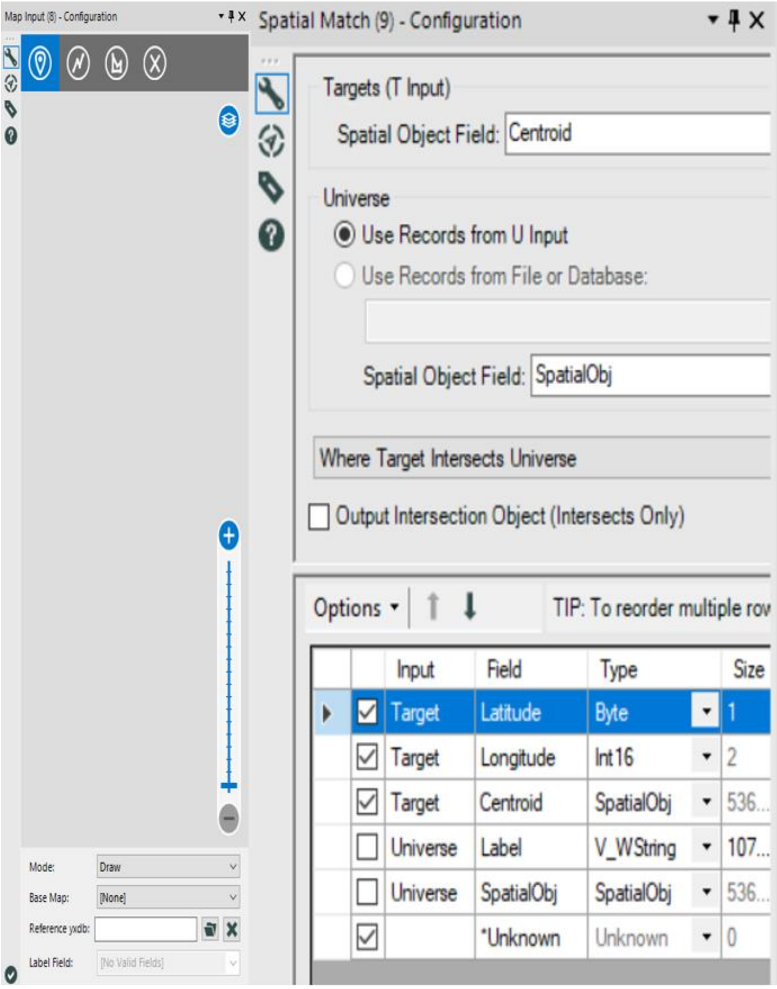


Figure-9-19-Mermaid Coffee using Map Input and Spatial Match to filter incoming data

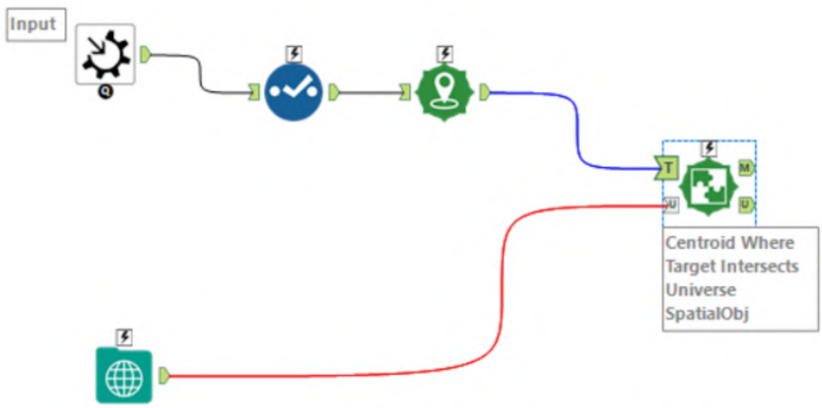


Figure-9-20-Mermaid Coffee incoming Data Stream

In order to limit the incoming data, we are going to use a *Map Input* tool to create a spatial object that will be passed to the *Spatial Match* tool. The *Spatial Match* tool has the ability to compare different spatial objects to determine if they share space.

We are not going to configure the *Map Input* tool because we are going to be giving it spatial objects from an interface tool called *Map* once we have finished the basic workflow.

The *Spatial Match* tool will use the configuration shown in the image to include only locations that are in/or touch the user-defined Map Input region.

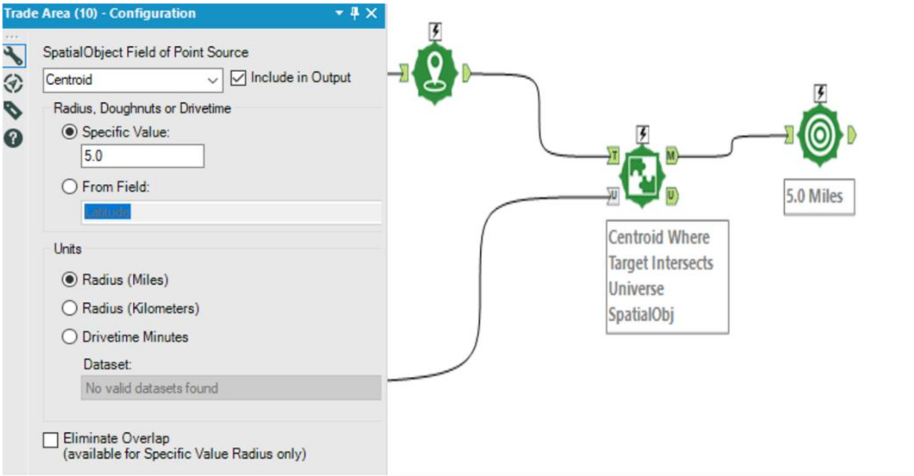


Figure-9-21-Mermaid Coffee with points and limits

Now that we have limited the points to only those in the area of interest, we can create the *Trade Area* polygons for those points. We are going to make sure that the units are miles, but it doesn't matter that the *Specific Value* is 5 because we are going to use a *Numeric Up Down* to set the distance.

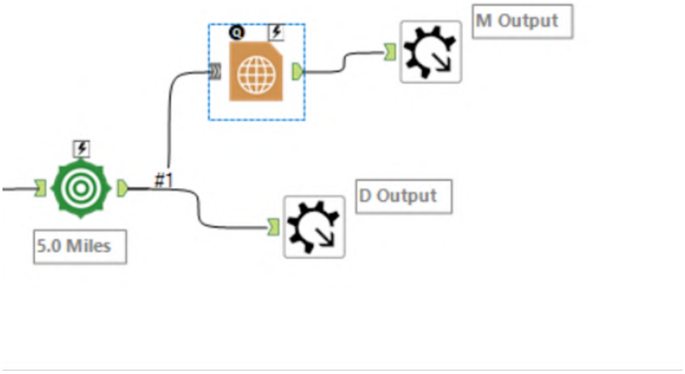


Figure-9-22-Mermaid Coffee Trade Area Polygons

Now that we have created our trade areas, we need to create outputs.

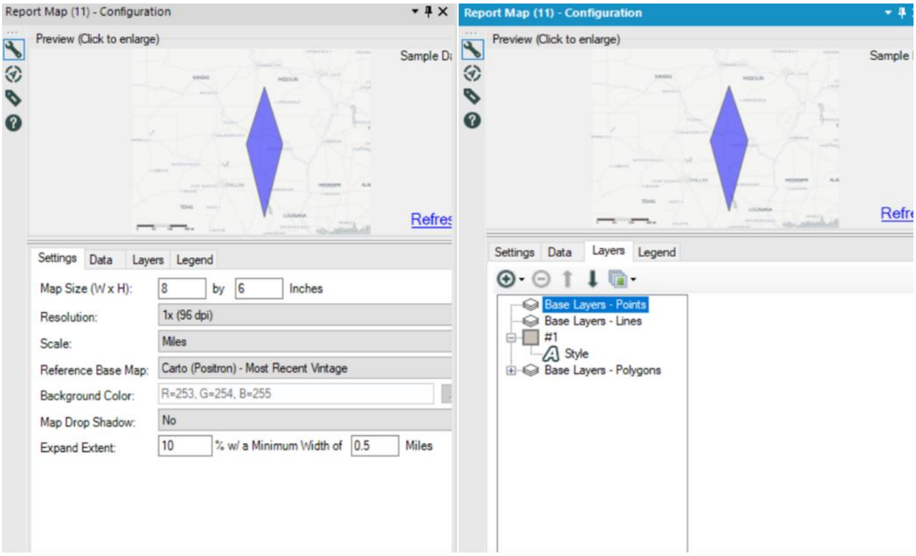


Figure-9-23-Mermaid Coffee- Using Report Map to create object

For the map output, we need to create a map report object. Use any setting of choice; just make sure to include a base layer and the polygon layer from the data stream.

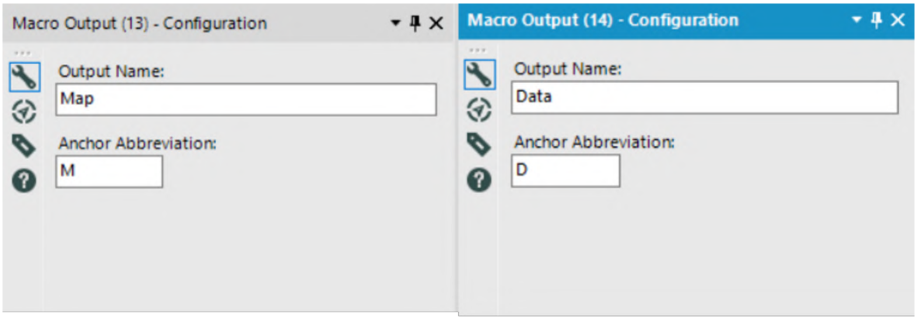


Figure-9-24-Mermaid Coffee Macro Output

Now all that is left for the outputs is to name them and provide them with an identifying anchor character.

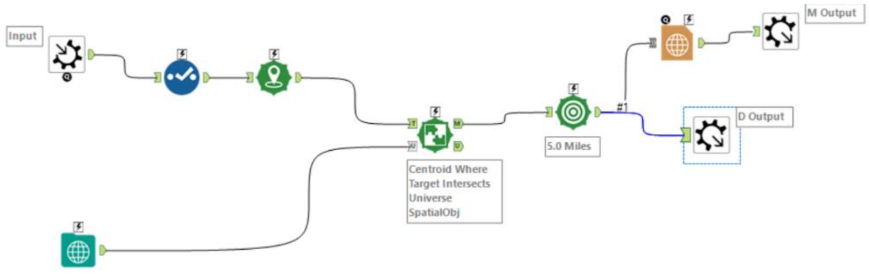


Figure-9-25-Mermaid Coffee data stream after Macro Output

This macro now only needs the user settings so that the controls can be defined for the two tools we want to be able to change.

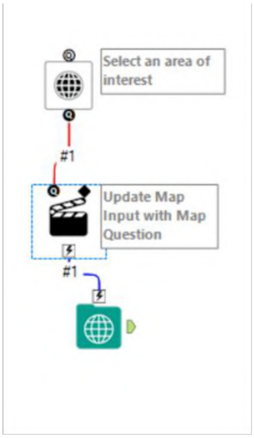


Figure-9-26-Mermaid Coffee using Map tool

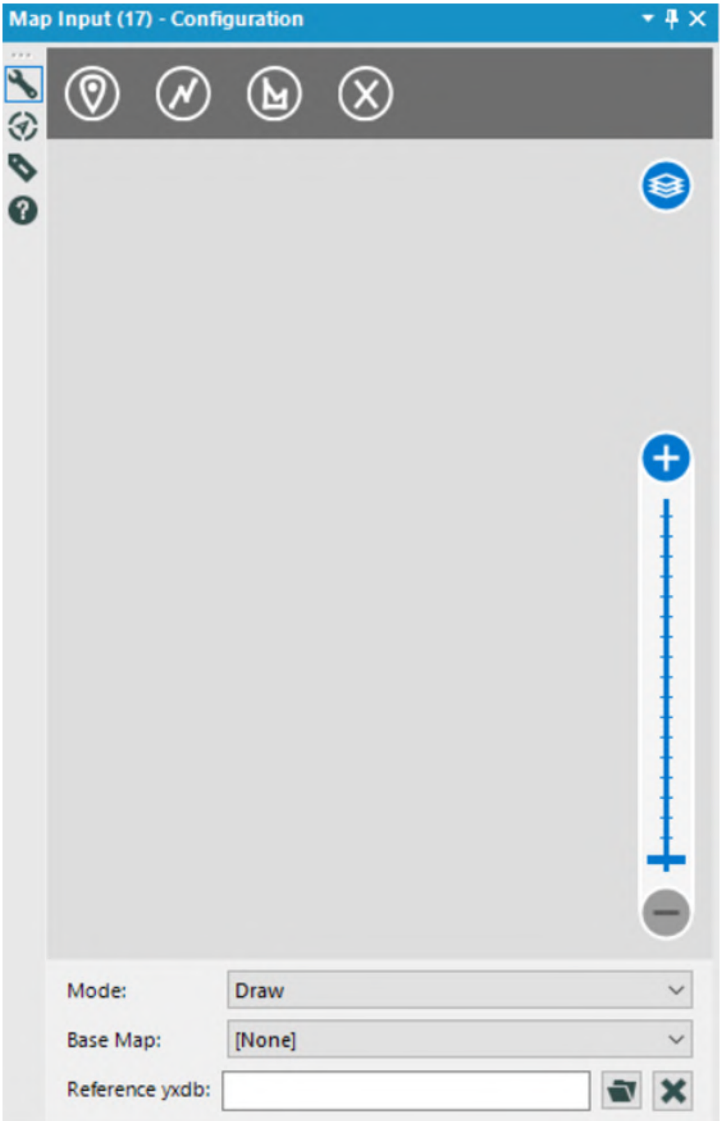


Figure-9-27-Map Tool Configuration

We can use the *Map* tool to create a map in the interface, which allows the end user to select a specific region.

If we connect the *Map* tool directly to the *Map Input* tool, we see the *Action* tool will auto-populate with the appropriate settings.

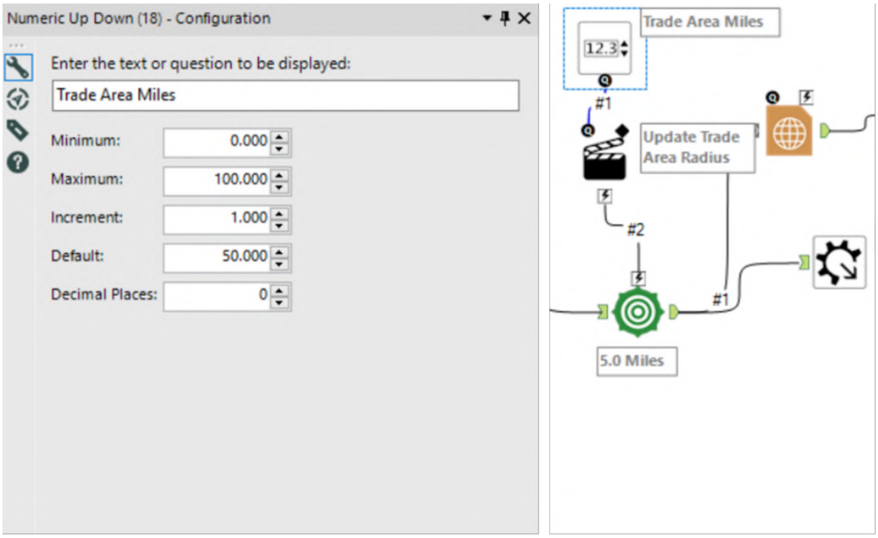


Figure-9-28-Mermaid coffee using numeric up down to set Number of miles a trade should cover

If we add a *Numeric Up Down* tool, we will have a numeric input so the user can set the number of miles a *Trade Area* should cover. If we connect this directly to the *Trade Area* tool, we can see the *Action* tool uses the special *Trade Area Radius* setting (so we don't need to configure the *Action* tool). We are giving a much wider set of selected radii here because we don't know how we may want this tool to be used in the future. Since we will be embedding this in another tool we can further limit the options for our current project.

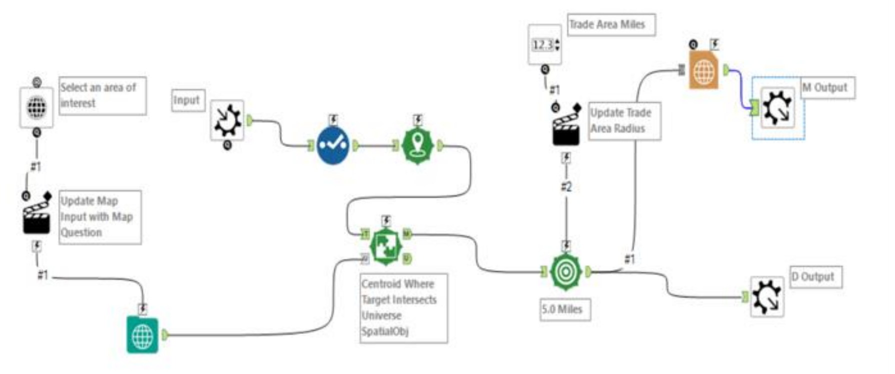


Figure-9-29-Mermaid Coffee Data Stream

Now that we have this macro, we can save it to use in other data streams. We are going to create a macro folder in *My Documents* called *Alteryx Macros* and create a sub-folder called *Training Macros*. Let us save this macro there, and we should modify the user settings so our macro is available in the *Tool Palette*.

Navigate to the Edit User Setting window in Tools > User Settings > Edit User Settings.

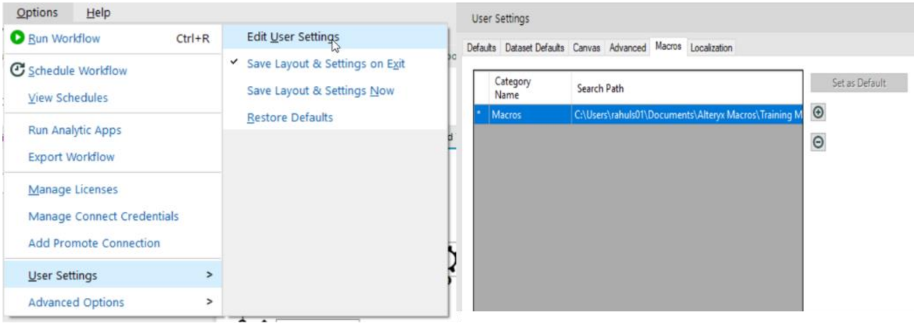


Figure-9-30-Editing User Settings of Macro

If we go to the Macros Tab, we see the categories we defined and the folder location related to those categories. We have a *General*

Macros category and a *Training Macros* category. If we click on the add symbol to the right of the window, it navigates to the *Training Macros*. Let's add it to this list with the title *Training Macros*

Figure-9-31-Tool Palette

If we take a look at the *Tool Palette,* we see at the far-right side a Category called *Training Macros*. The image shown here as well as description information can be edited if we modify the properties in the workflow *Configuration window* and *Interface designer.*

Figure-9-32-newly saved as macro

Now that we have created the macro, we can build the app specific to the Mermaid Coffee Company. Open a new workflow and connect to the Mermaid Coffee Company Locations file. We can see that all fields are V_String. But we need Latitude and

Longitude in a decimal format, which means we need to convert those fields to double so we will have no issue using our macro.

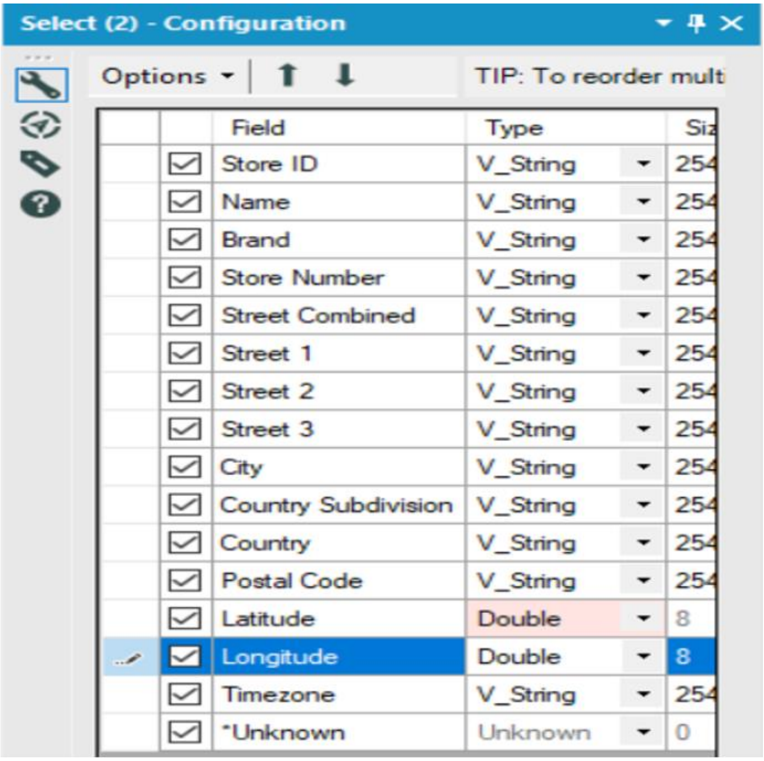


Figure-9-33-Mermaid Coffee Select

Alteryx Designer x64 - Engine_11744_41fe749d281b479c8739137468cd93fb_.yxdb

21,438 records displayed, 15 fields, , 2.4 MB

Table

15 of 15 Fields ▾ Cell Viewer ▾

Record #	Store ID	Name	Brand	Store Number	
1	1	Plaza Hollywood	Mermaid Coffee Company	34638-85784	Leve
2	2	Exchange Square	Mermaid Coffee Company	34601-20281	Sho
3	3	Telford Plaza I	Mermaid Coffee Company	34610-28207	Sho
4	4	Hong Kong Station	Mermaid Coffee Company	34622-64463	Con
5	5	Pacific Place, Central	Mermaid Coffee Company	34609-22927	Sho
6	6	Hung Hom KCRC	Mermaid Coffee Company	34633-55579	Sho
7	7	Citibank Plaza	Mermaid Coffee Company	34616-51681	Sho
8	8	MTR - Central Transfer Concourse	Mermaid Coffee Company	34630-69462	Kios
9	9	Wan Chai Tower	Mermaid Coffee Company	34615-55880	G/F,
10	10	Shun Tak Centre	Mermaid Coffee Company	34634-82422	Sho
11	11	Sun Hung Kai Centre - Lift Lobby	Mermaid Coffee Company	34605-28027	Sho
12	12	Lamcy Plaza	Mermaid Coffee Company	34247-62179	Oud
13	13	Wafi Mall	Mermaid Coffee Company	34249-62099	Oud
14	14	Al Gurrair Mall	Mermaid Coffee Company	34219-17920	Deir
15	15	Umm Sequeim	Mermaid Coffee Company	34291-80114	Jum
16	16	Jebel Ali	Mermaid Coffee Company	34292-76574	Jebe
17	17	Emirates Tower	Mermaid Coffee Company	34211-17922	Shei
18	18	Palm Strip	Mermaid Coffee Company	34210-16835	Jum

Figure-9-34- Mermaid Coffee Browse

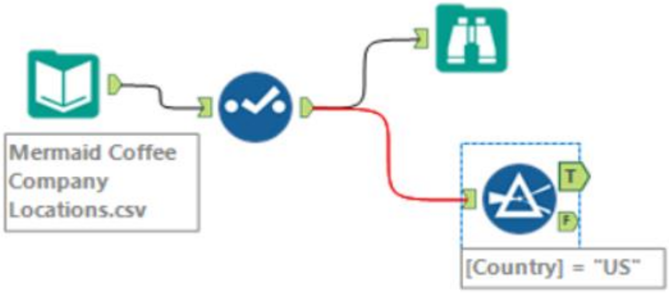


Figure-9-35- Mermaid Coffee US location

We can see from the email that they are only interested in US locations, so we can remove all other countries. This means the data is ready for use in our macro.

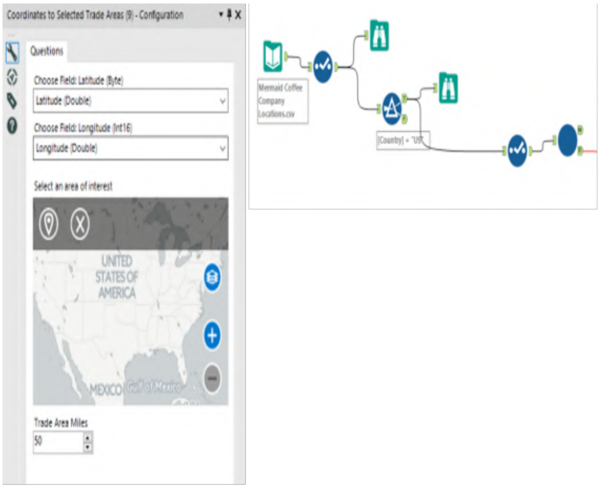


Figure-9-36-Mermaid Coffee trade Area Configuration to restrict to US

We must make sure the Latitude and Longitude fields match correctly but leave the other *Map* and *Trade Area Miles* as defaults for now. We will connect a *Map* tool and *Numeric Up Down* here to allow the end user to update these values.

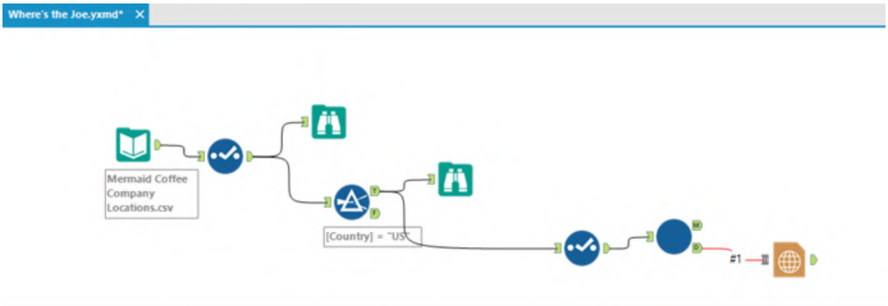


Figure-9-37-Mermaid Coffee US locations after setting trade area

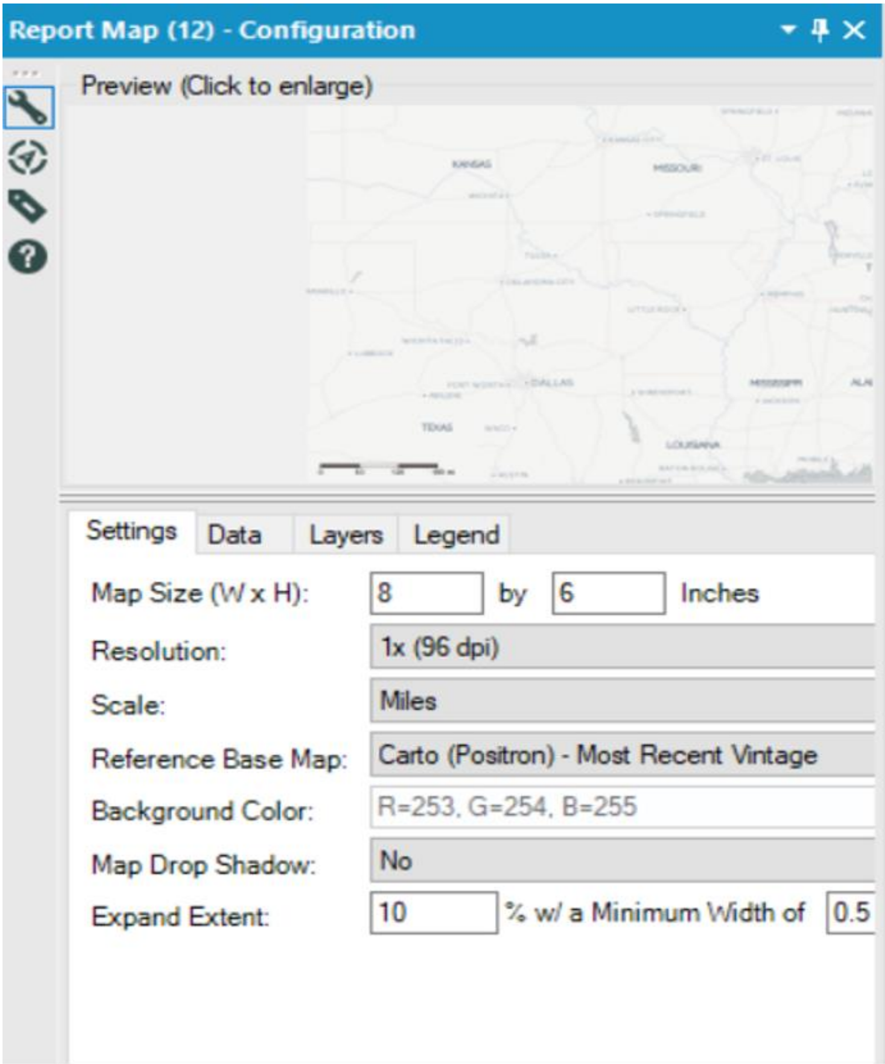


Figure-9-38-Mermaid Coffee Report Map Settings

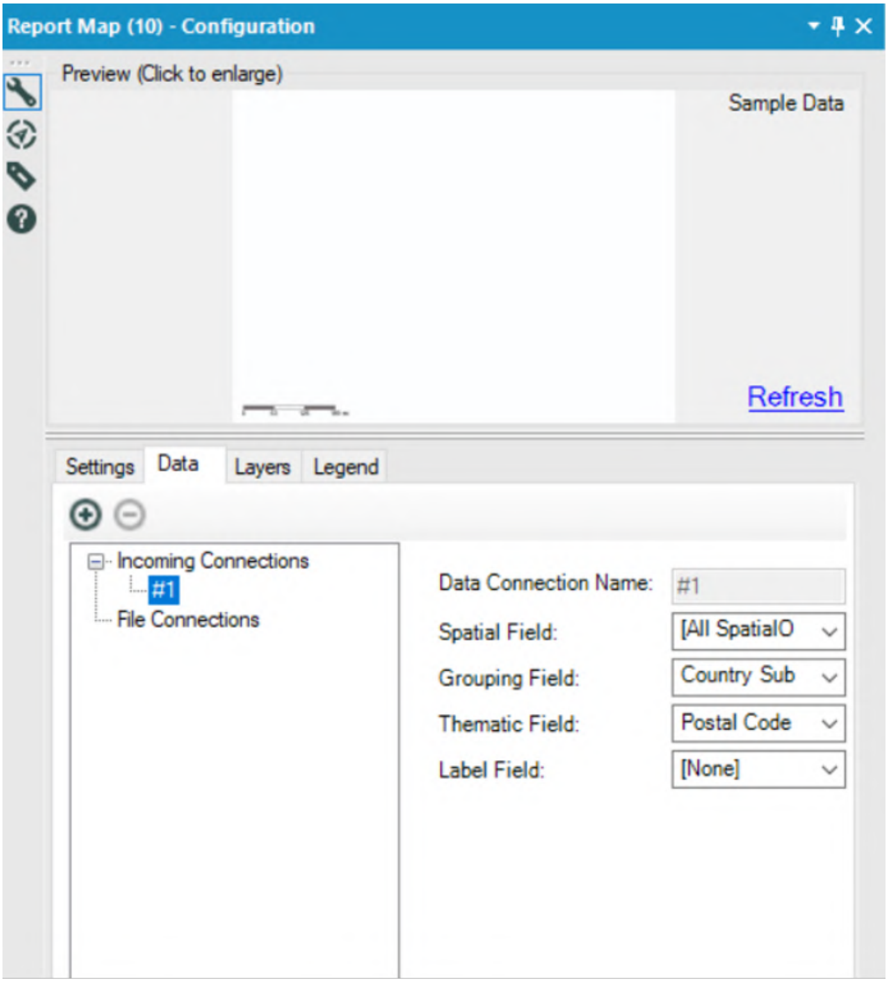


Figure-9-39-Mermaid Coffee Report Map Data

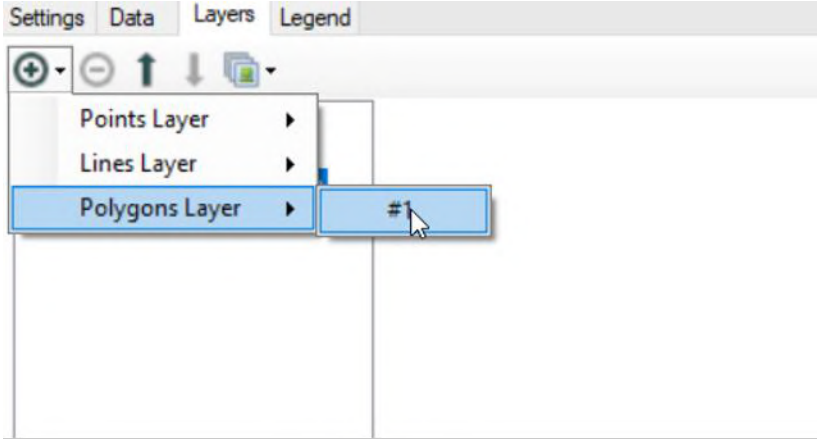


Figure-9-40-Mermaid Coffee Report Map Layers

We know our macro created a universal map, so we can use that directly. However, we also need individual maps for each state for which we use another *Report Map* tool with the previously detailed settings to create each of the individual state maps.

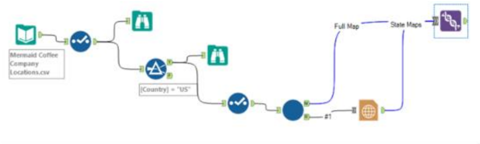


Figure-9-41-Mermaid Coffee Data Stream with Maps

Now that we have both sets of maps, we can combine them. If we want the overall map to precede the others, we need to adjust the order in the *Union* tool.

We have all of our maps in a single field. We can use the *Render* tool to create our report.

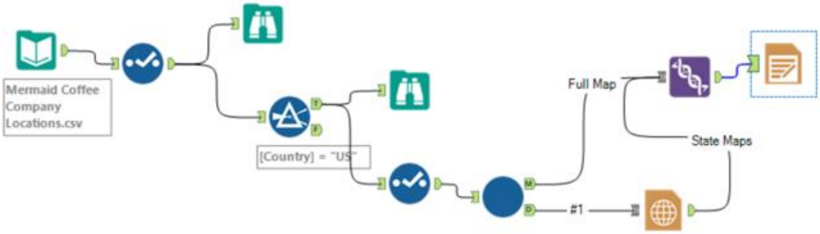


Figure-9-42-Mermaid Coffee using Union Tool to get overall map

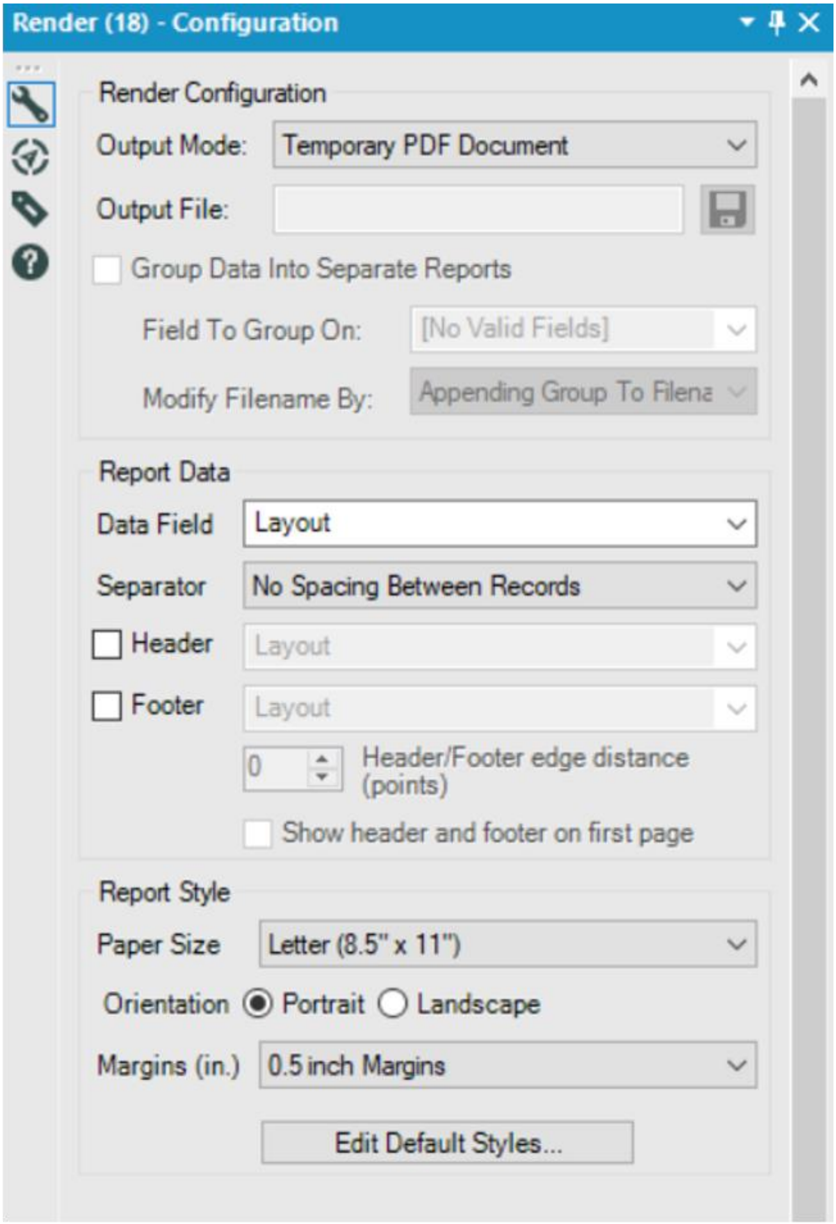


Figure-9-43-Mermaid Coffee creating Report from maps

Note that we are inserting section breaks between each record to make sure each map is on a page by itself.

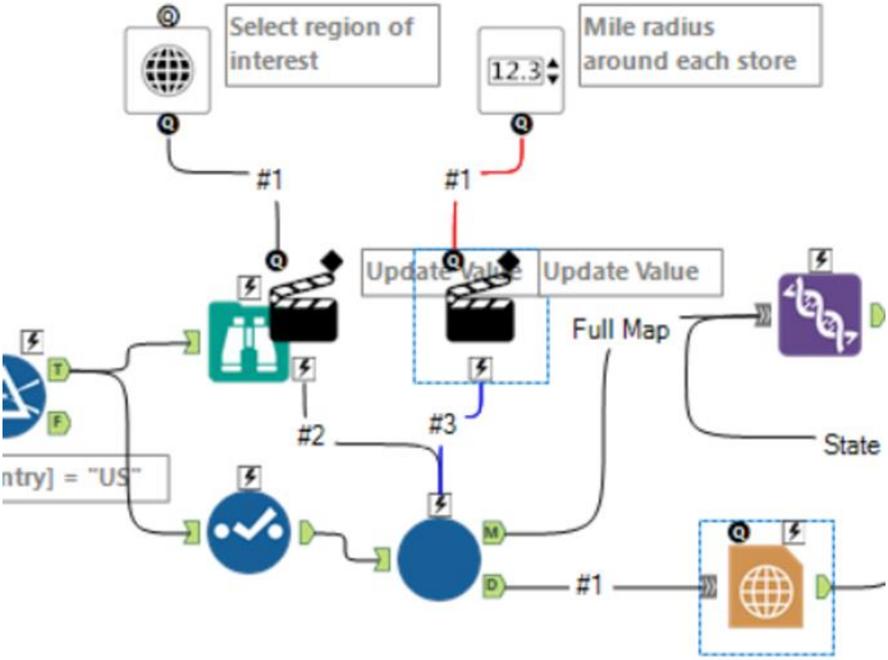


Figure-9-44-Mermaid Coffee selection range of interest

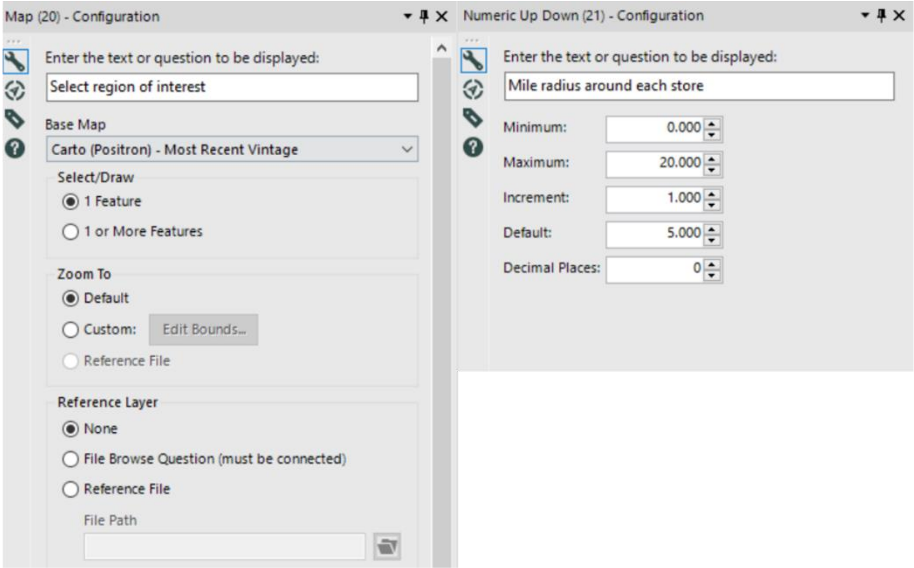


Figure-9-45-Mermaid Coffee map and Numeric Up down to get desired view

Now that we have the workflow developed, we can finish off this application by adding the two desired user inputs. We will define the *Map* input the same way we did in the macro. However, the *Numeric Up Down* will have a different range of possible values and default value so it reflects what the Mermaid Coffee Company would like to see.

The Where’s The Joe? data stream should look like this when it’s complete.

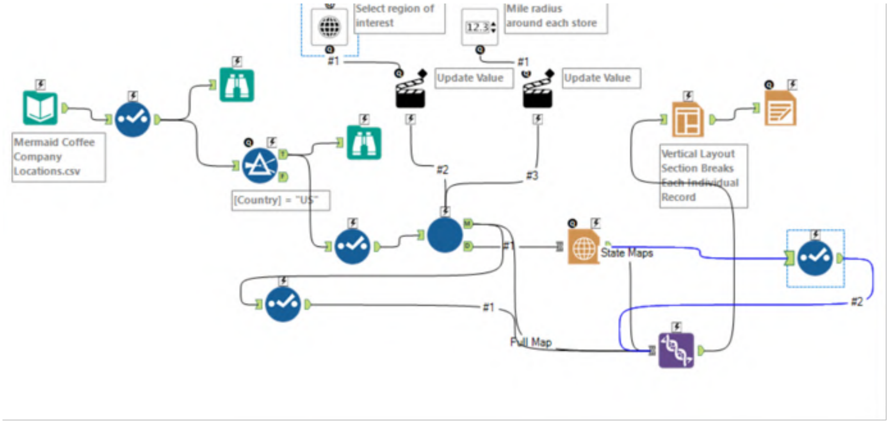
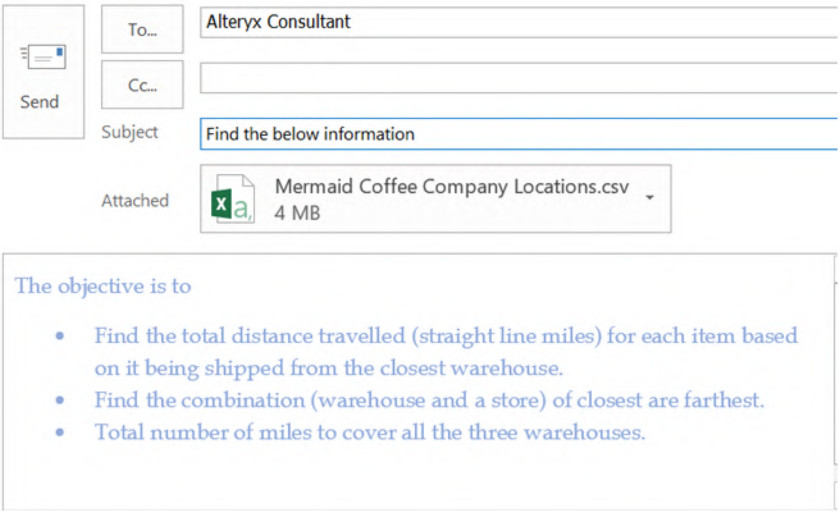


Figure-9-46-Mermaid Coffee Final Data Stream when complete

9.3 Find the locations



Here we are going t learn about few Spatial tools which we have not yet learnt. We will do that by solving given three problems above.

To find the total distance travelled (straight line miles) for each item based on it being shipped from the closest warehouse. We will use the datasets *Chapter 9 - Street Address Data.csv, Street Data.csv and Warehouse Data.csv.*

Let’s start with joining the datasets Street Address Data.csv, Street Data.csv. Join them using Join tool on Store. Then Once we have the results, we use the create points tool to generate the latitude and longitude.

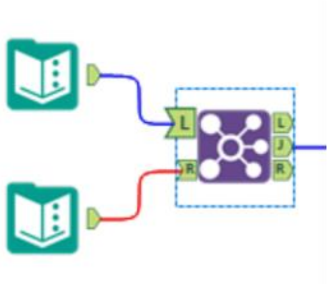


Figure-9-47-Find the Locations Data Stream

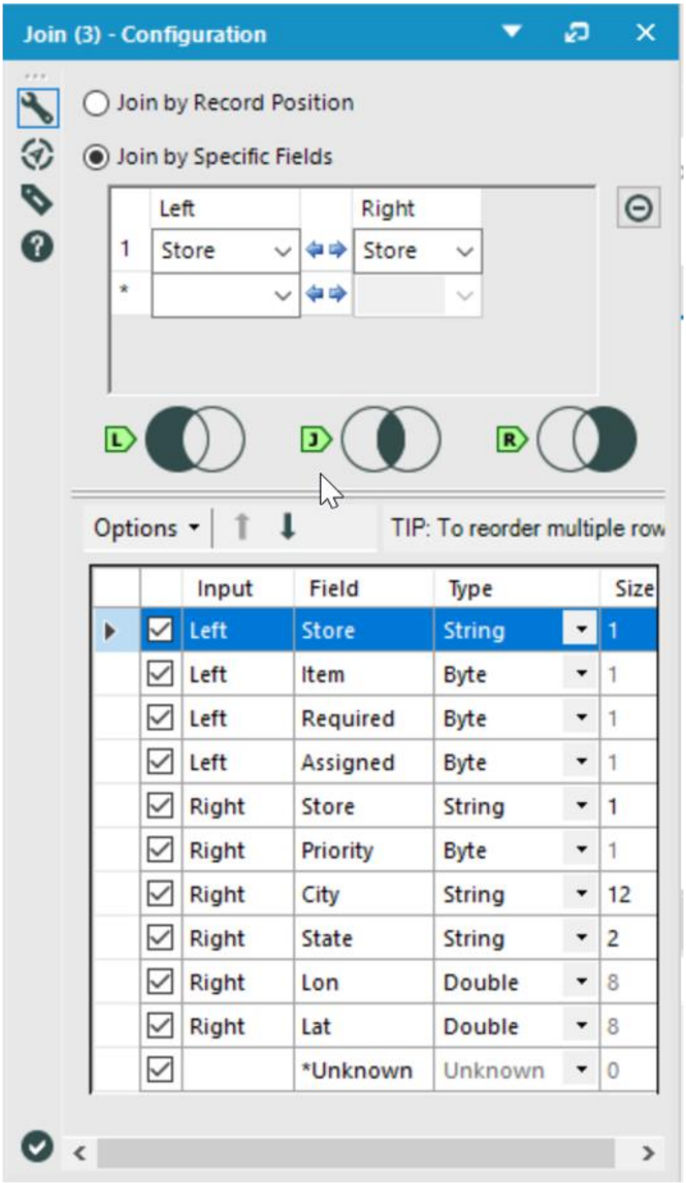


Figure-9-48-Join Configuration

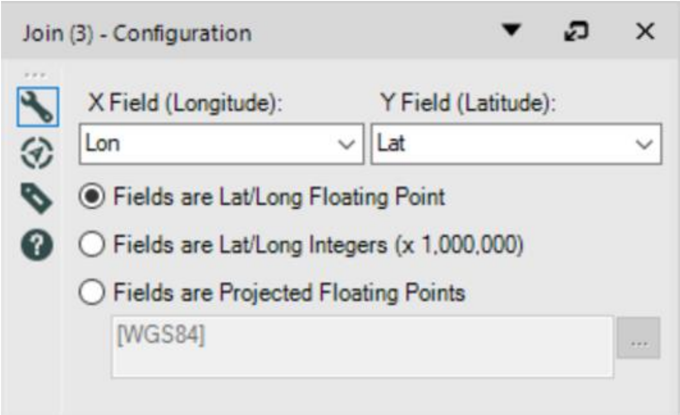
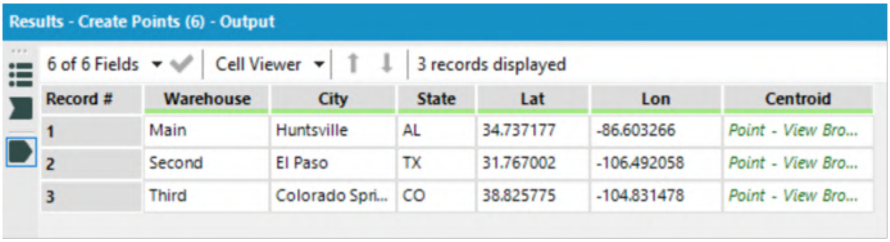


Figure-9-49-Create Points Configuration

Once we have generated the centroid, use the Find Nearest tool to combine results of both Create Points tool and generate the *distance Miles.*


Results - Create Points (6) - Output

6 of 6 Fields · Cell Viewer · 3 records displayed

Record #	Warehouse	City	State	Lat	Lon	Centroid
1	Main	Huntsville	AL	34.737177	-86.603266	Point - View Bro...
2	Second	El Paso	TX	31.767002	-106.492058	Point - View Bro...
3	Third	Colorado Spri...	CO	38.825775	-104.831478	Point - View Bro...

Figure-9-50-Create Points Result

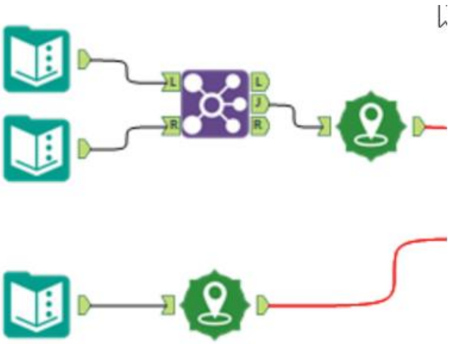


Figure-9-51-Find The Locations Stream

The Create Points Configuration has the below specifications.

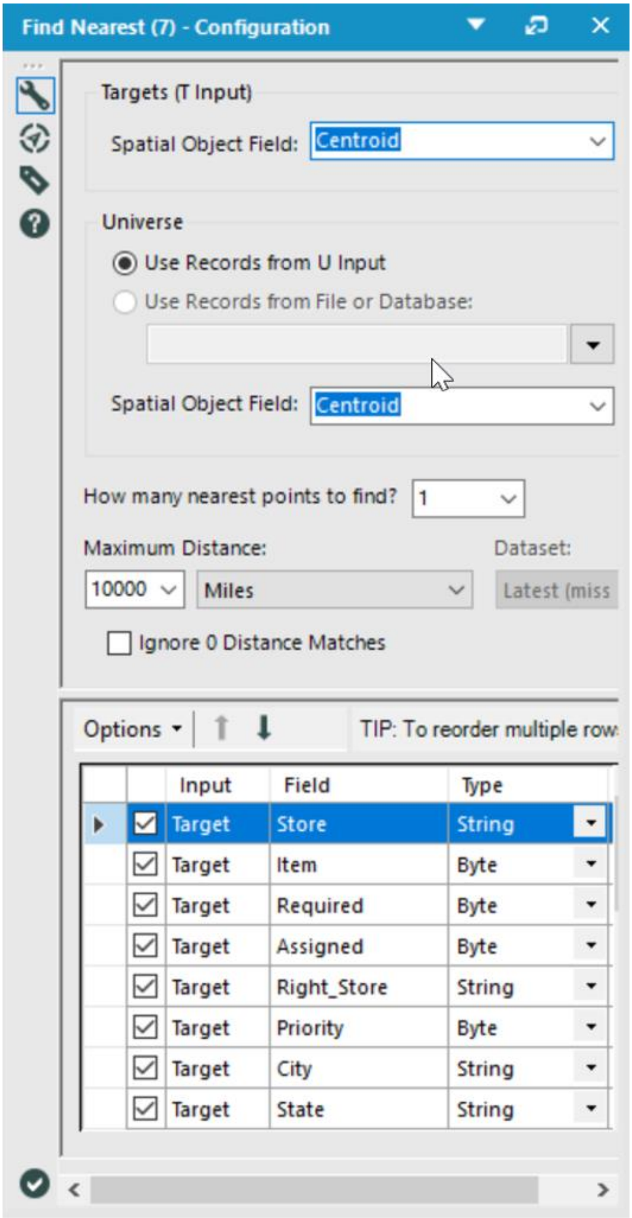


Figure-9-52-Find Nearest Configuration

Total Miles can be calculated using Formula Tool. The configuration for the Formula Tool is below.

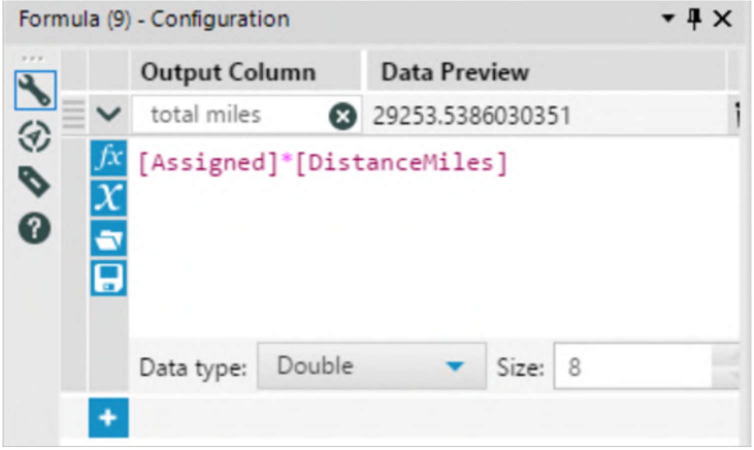


Figure-9-53-Formula Configuration

After getting the Total Miles value we use summarize tool to calculate the total miles distance per item.

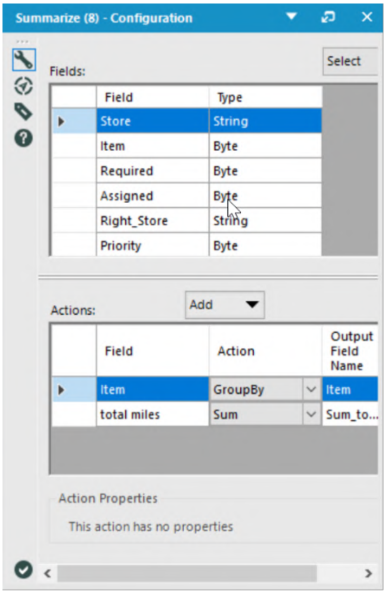
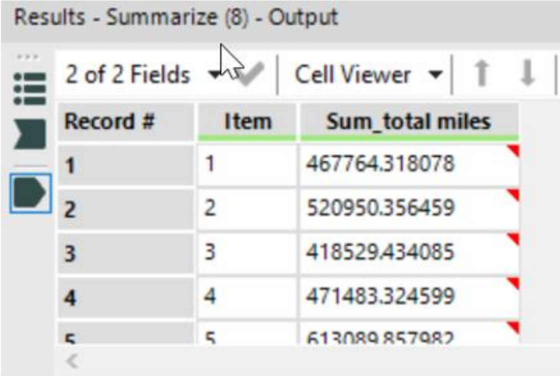


Figure-9-54-Summarize Configuration

The Output of summarize tool and the final workbook is provided below.


Results - Summarize (8) - Output

2 of 2 Fields | Cell Viewer

Record #	Item	Sum_total miles
1	1	467764.318078
2	2	520950.356459
3	3	418529.434085
4	4	471483.324599
5	5	613089.857982

Figure-9-55-Summarize Output

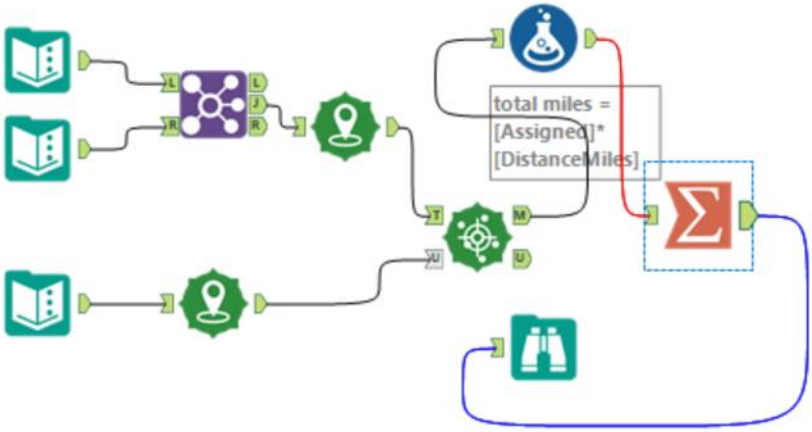


Figure-9-56-Find The Locations Data Stream When Complete part-1

To calculate and find the farthest and nearest warehouses on the map. We use both the datasets - *Street Address Data.csv, Warehouse Data.csv.*

We create centroids using Create Points using both the datasets. Use Append Fields tool to use combine both the datasets and utilize it.

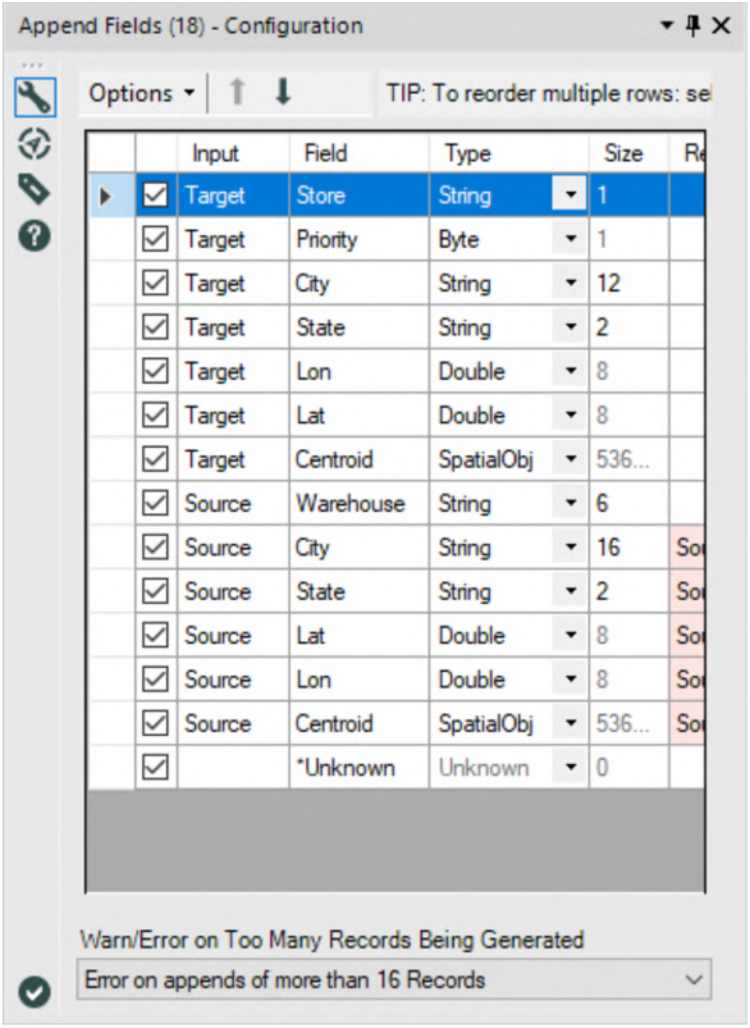


Figure-9-57-Append Filed Configuration

The workflow till now is below.

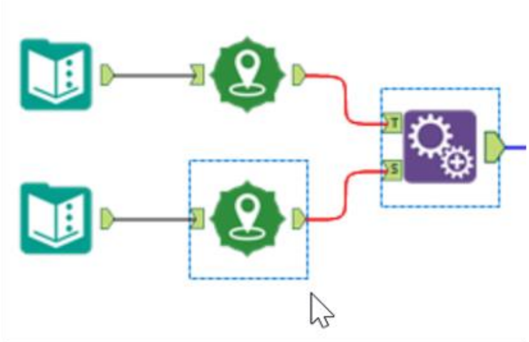


Figure-9-58-Find The Locations Data Stream part-2

We use a Distance Tool to compare and plot the distance on map for all the warehouses. Use the Sort tool to sort the DistanceMiles created.

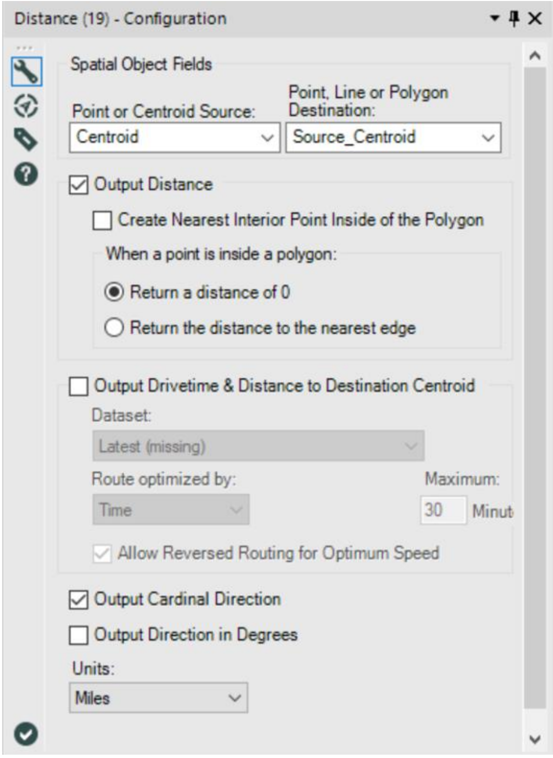


Figure-9-59-Distance Tool Configuration

The Output of this DataStream is below.

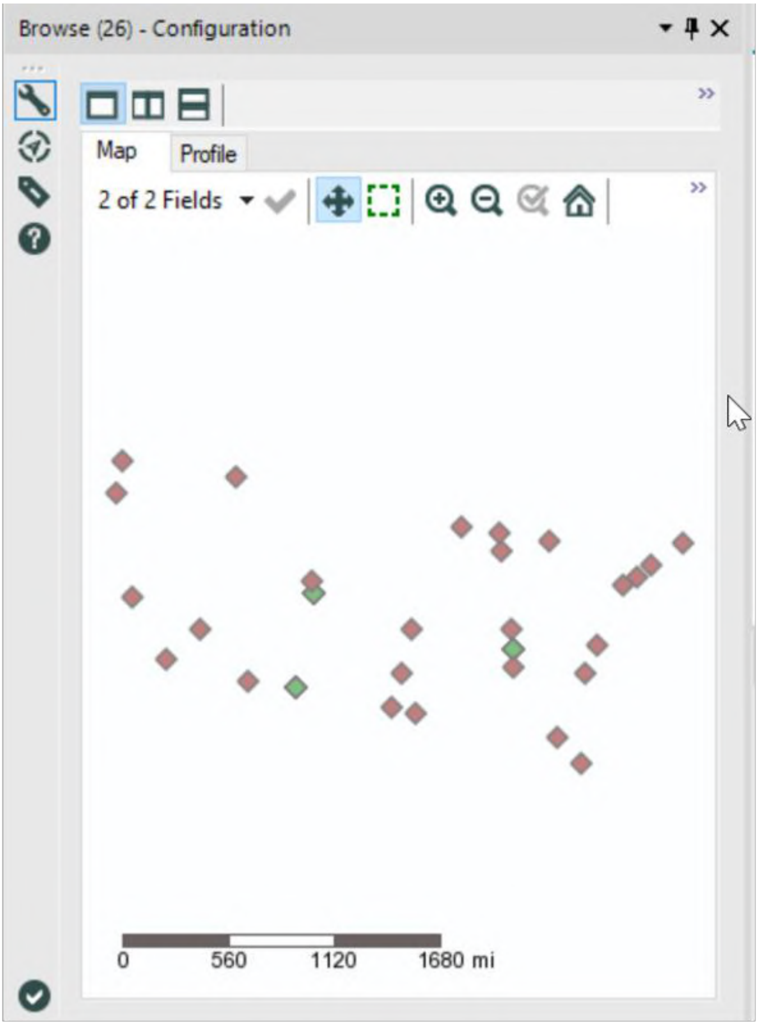


Figure-9-60-Find The Locations part-2 Output

The Find the Locations Data Stream part-2 when complete is below.

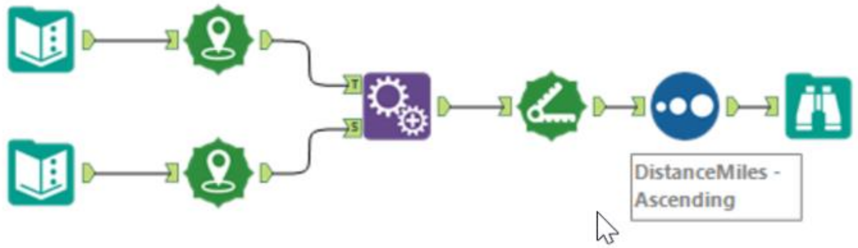


Figure-9-61-Find the Locations Data Stream part-2 When Complete

For the third part where we need to find out the total distance between the three warehouses, we use poly build tool to form a polygon. We will be using dataset *Warehouse Data.csv.*

Configuration of Polybuild Tool is below.

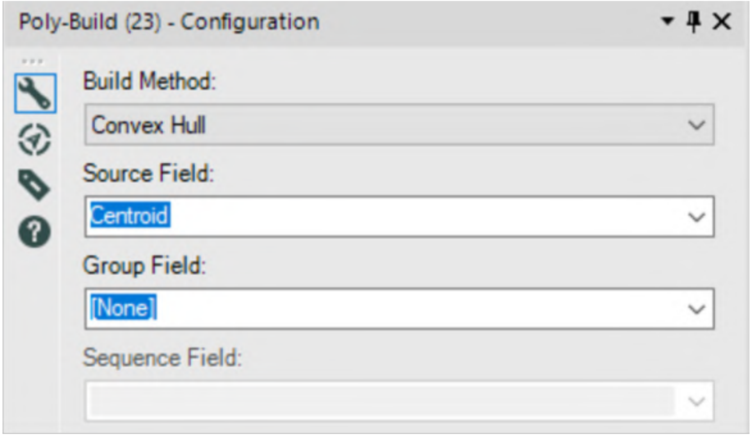


Figure-9-62-Poly-Build Configuration

We will use Spatial Info Tool to Find the length in miles the warehouses cover.

Spatial Info Configuration is below.

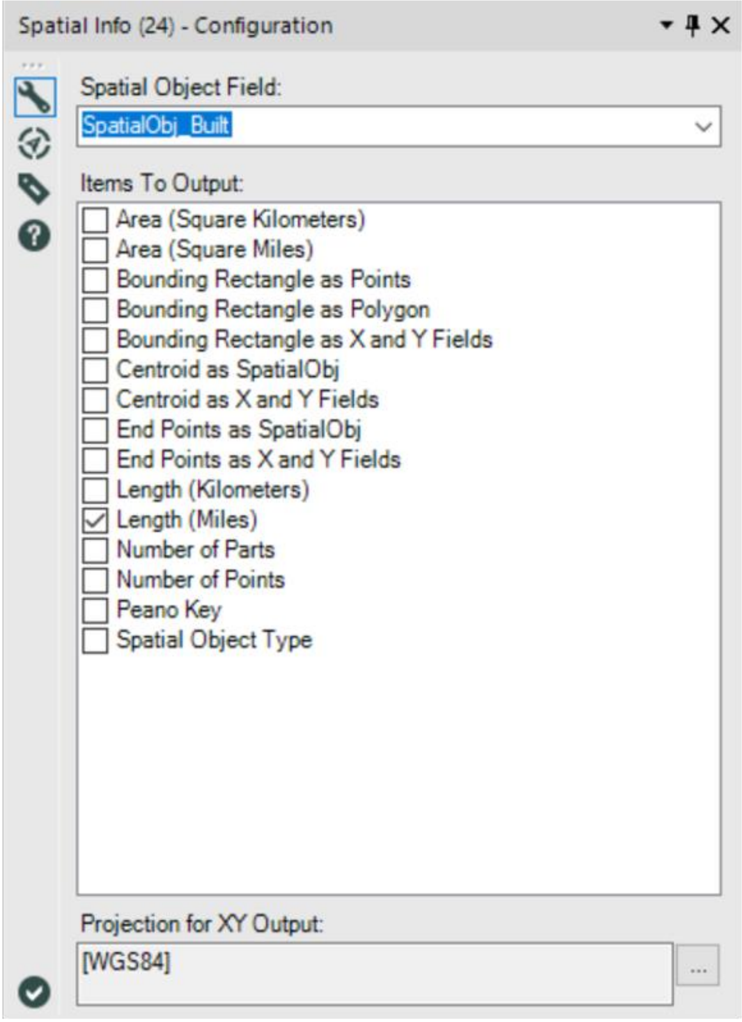


Figure-9-63-Spatial-Info Configuration

The final polygon area created after the workflow is below.

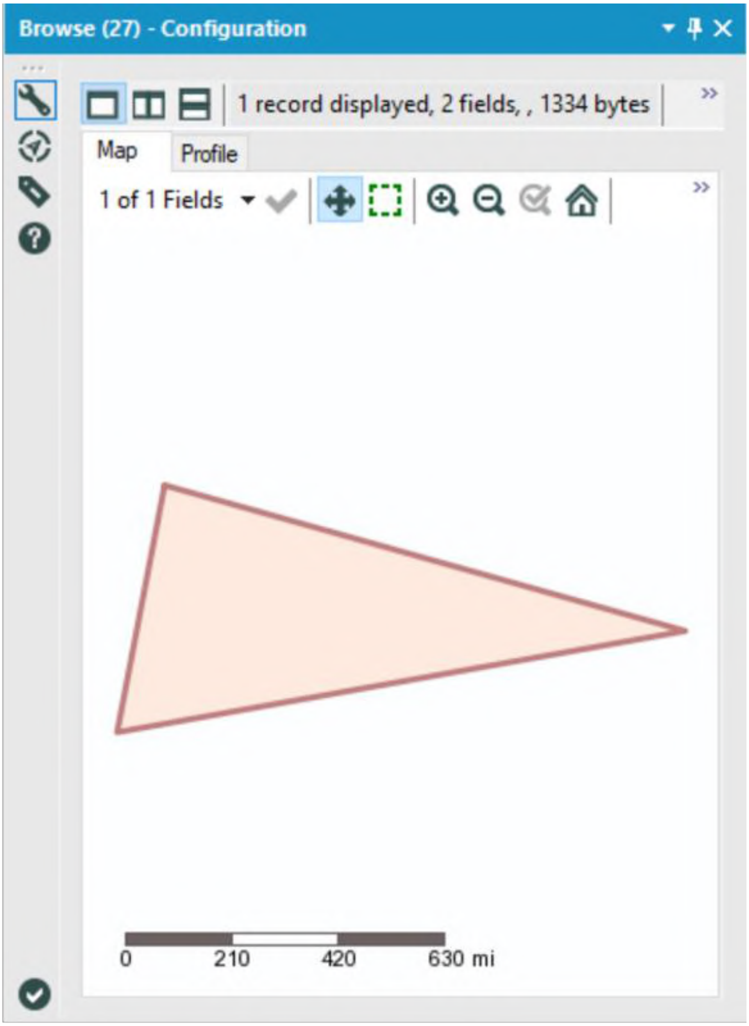


Figure-9-64-Find the Locations Data Stream Output

We can see that a polygon has been formed with the area covering all three warehouses.

Figure-9-65-Find the Location part-3 Data Stream When Complete

CHAPTER 10

Meta-Morphosis

To... Alteryx Consultants

Subject All The Tools Along the way

Attached JSON.txt

Hey,

I have been asked to make sure you know how to use everything in a list of tools from the higher-ups.

There are a few that you haven't gotten to know yet, so I have designed something for us to go over and then for you to work through.

Attached, you will find a file called JSON.txt. We will be using it to create an application such that when you check a field, the JSON data will be converted into a relational structure. If you don't check the field, it will output a file identical to the original JSON file.

Thanks

10.1 Tools & Concepts

10.1.1 Check Box
Concept - Alternative Data Prep

 **Figure 10-1 – Check Box**	**Tool Palette: Interface**
	This interface tool allows us to create a check box question. For more details use the link below. bit.ly/2qUHptH

10.1.2 Detour
Concept - JSON Files

 **Figure 10-2 – Detour**	**Tool Palette: Developer**
	This tool allows us to create alternative paths dependent on the user input. For more details use the link below. bit.ly/2qUfQk9

10.1.3 Detour End
Concept - Documentation

	Tool Palette: Developer

 **Figure 10-3 – Detour End**	This tool joins the two optional paths that were created using Detour. For more details use the link below. bit.ly/2qYau6m

10.1.4 Dynamic Rename
Concept - Data Cleaning

 **Figure 10-4 – Dynamic Rename**	**Tool Palette: Developer** This tool allows us to systematically rename fields. For more details use the link below. bit.ly/2KcIxAU

10.1.5 JSON Parse
Concept - JSON Files

 **Figure 10-5 – JSON Parse**	**Tool Palette: Developer** This tool allows us to read JSON formatted files into relational tables For more details use the link below. bit.ly/2Hr9IWT

10.1.6 Message

Concept - Information Display

 **Figure 10-6 – Message**	**Tool Palette: Developer**
	This tool allows us to create a message in the output window. For more details use the link below. bit.ly/2qYBF11

10.2 All the Tools Along the Way

To... Alteryx Consultants

Subject All The Tools Along the way

Attached JSON.txt

Hey,

I have been asked to make sure you know how to use everything in a list of tools from the higher-ups.

There are a few that you haven't gotten to know yet, so I have designed something for us to go over and then for you to work through.

Attached, you will find a file called JSON.txt. We will be using it to create an application such that when you check a field, the JSON data will be converted into a relational structure. If you don't check the field, it will output a file identical to the original JSON file.

Thanks,

The option for a tool to essentially just copy the file seems a little strange, but it will make a more sense when we start working on the follow-up assignment. We can write the file to a .csv instead of a .txt file at the end of the workflow. We will also be demonstrating the functionality using an Analytical App, which has been explained at the end of the chapter. Let us start by connecting to the data. The data is *Chapter 10- Mets-Morphosis-JSON.*

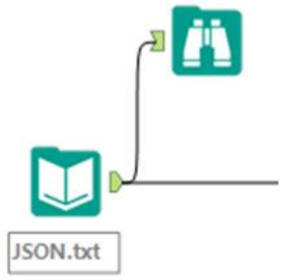


Figure 10-7 – Open JSON File

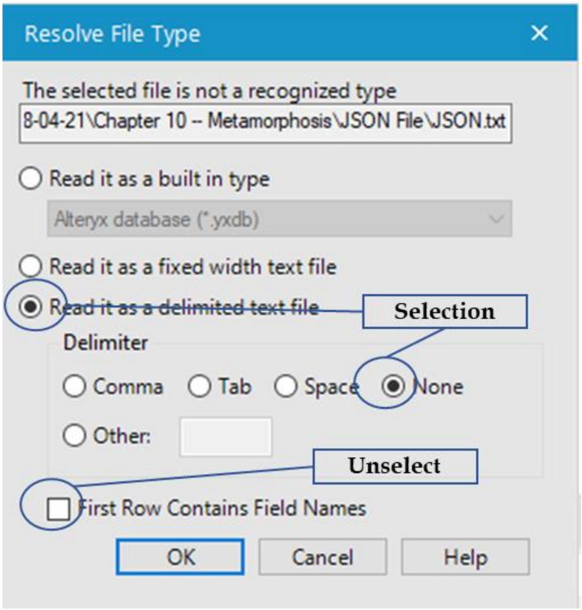


Figure 10-8 – Read JSON File

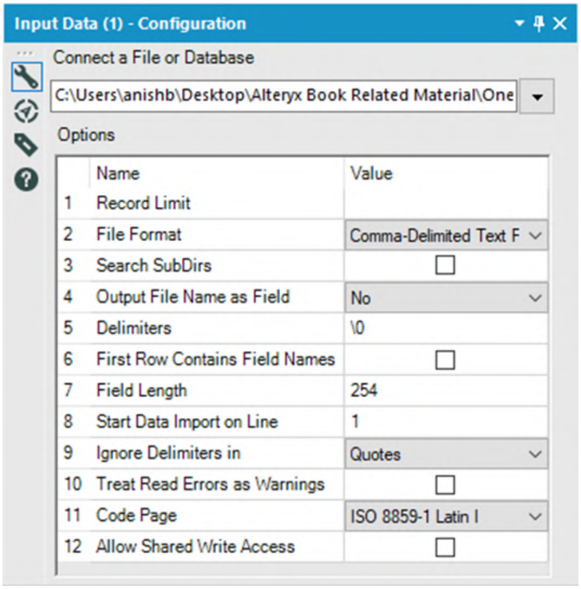


Figure 10-9 – Input tool Configuration

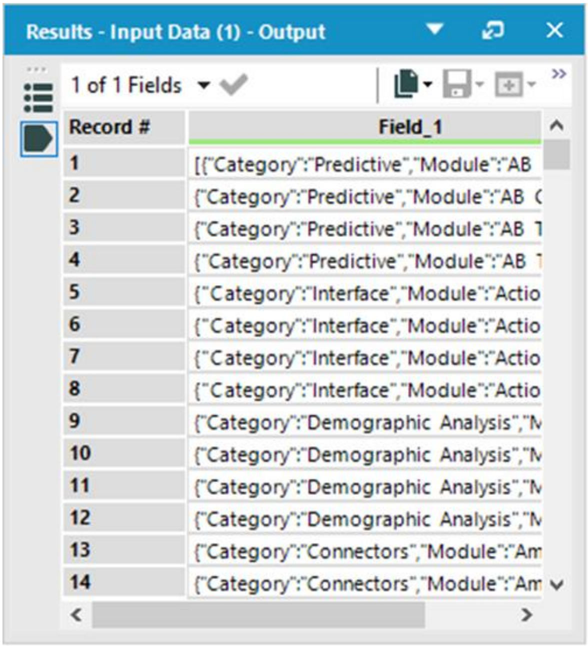


Figure 10-10 – Preview of JSON File

When we connect to this file, because it is *.txt*, Alteryx needs to know how it should read it. This is by design because file formats like JSON need to be read so that all data is in a single field and ideally one record per row. Because we want these settings, we can choose to have the file delimited with no delimiter so that nothing will break the field. If we take a look at the file, we can see there is no header row. This is because JSON carries the header names in every record and thus does not need a devoted row.

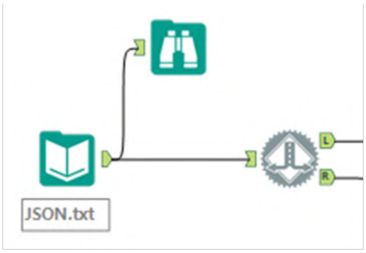


Figure 10-11 – JSON Detour

Figure 10-12 – JSON Detour Configuration

<u>Properties Window</u>:
The *Detour Configuration* window has only one setting.

- The default behavior for the detour tool is to always *divert to the left (L) data stream.*
- *Detour to the Right* allows us to reverse this default behavior.

Our next step will be to add a *Detour* tool. This tool allows us to use a question in order to decide which data stream to follow. The *Detour to the Right* option is so that while we are testing our workflow, we can properly check both sides. By design, the Detour tool detours to the left unless updated.

Figure 10-13 – JSON Detour to the left

Let’s start by designing the left (L) data stream. We are going to make this the side where we do not convert the data to a relational structure.

We are going to use *Message* tools to record what will happen to the data. The Message tool allows us to write something to the output window. In this case, we are going to write “*User Decided not to reformat the JSON data file*” so that it will be in the output log.

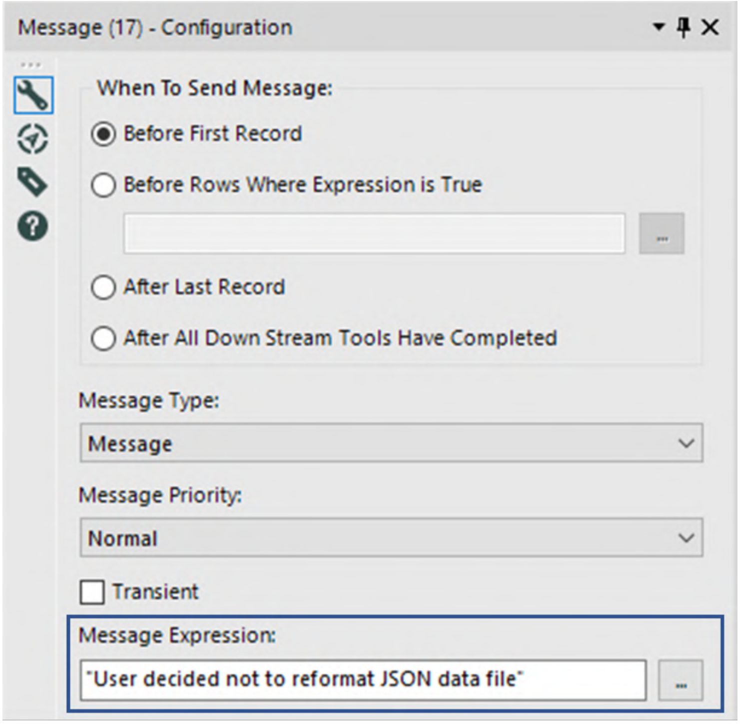


Figure 10-14 – JSON Message Configuration

<u>Properties Window</u>:
The *Message Configuration* window has five basic settings.

- *When to Send Message* defines what the trigger for writing out the message is. It can be just before the first record, before every record where a Boolean expression is true after the last record has passed through the tool, or when the data stream has terminated.
- *Message Type* defines what type of output should be written out and if it should stop the data flow (see Output Window in The Interface for more details).
- *Message Priority* determines what the message would do by default if it were in a macro. Normal will show the message if the tool is in the workflow. Medium will show the message if the tool is directly in a macro used in the workflow. High will show the message no matter how many macros deep the message occurred.

- *Transient,* when checked, means that if the Message tool writes more than one message, it will replace the single message instead of creating a new one every time.
- *Message Expression* is an expression field that will output the result to the output window.

Now let us add a Dynamic Rename tool. This is because once we bring the data back together to write, we will need to write the header. Since we don't want a header for this option on the data stream, we will modify it such that the header is actually the first record of data. This will complete all of the data prep we need before rejoining the data streams to be written out.

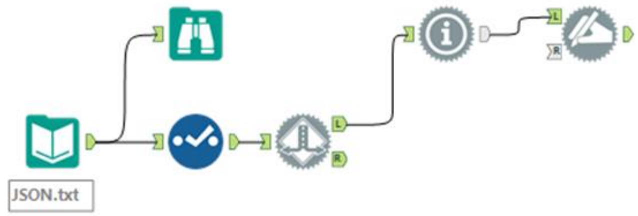


Figure 10-15 – JSON Left Data Stream

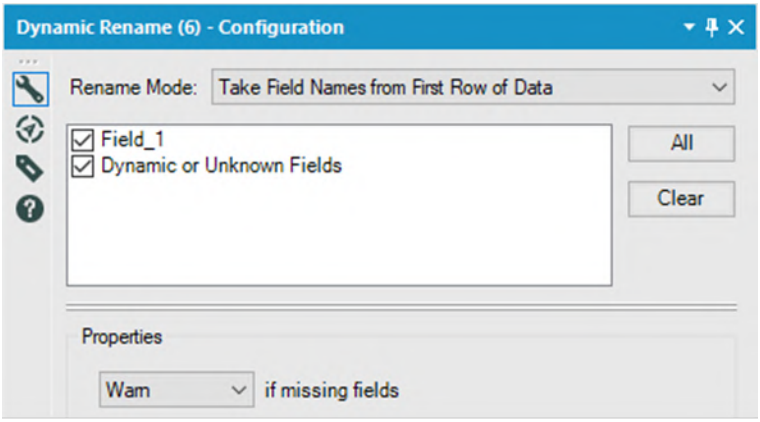


Figure 10-16 – JSON Dynamic Rename Tool Configuration

<u>Properties Window</u>:

The Dynamic Rename Configuration window has three core components and eight different modes.

- *Rename mode* allows us to select the method used to rename the fields in the left (L) input.

If we are using one of the single input rename modes, we will see a field selection window.

- The *Properties section* changes, depending on the Rename Mode.
- *Formula Rename Mode* allows us to define an expression that will name our columns.
- *Add Prefix/Suffix Rename Mode* allows us to define a string to add to the beginning or end of specific fields.
- *Remove Prefix/Suffix Mode* allows us to remove a common string from the beginning or end of specific fields.
- *Take Field Names from First Row of Data* allows us to replace the column headers with the first record of our data.
- *Take Field Descriptions from Right* Input Rows allows us to use a definition file to redefine the fields.
- *Take Field Names from Right Input* Rows allows us to use a definition file to rename the fields.
- *Take Field Names from Right Input Metadata* allows us to use a data stream that we know matches to rename the fields.

- *Take Field Descriptions from Right Input Metadata* allows us to use a data stream that we know matches to redefine the fields.

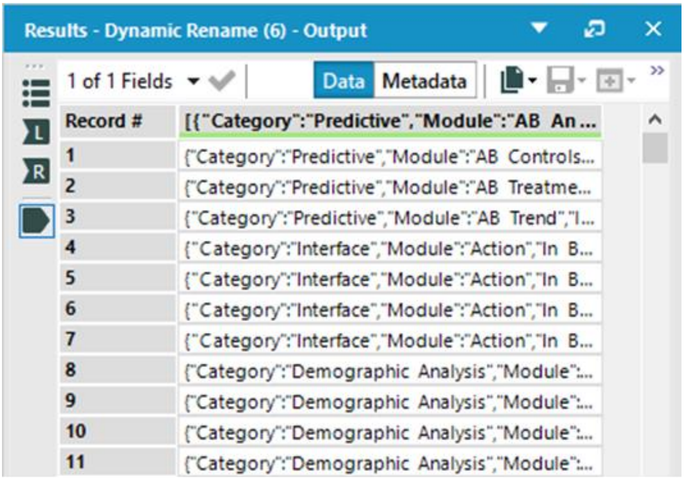


Figure 10-17 – Output After using Dynamic Rename

The left data stream has been completed and we will begin with setting up the other end of Detour that is the right Data Stream

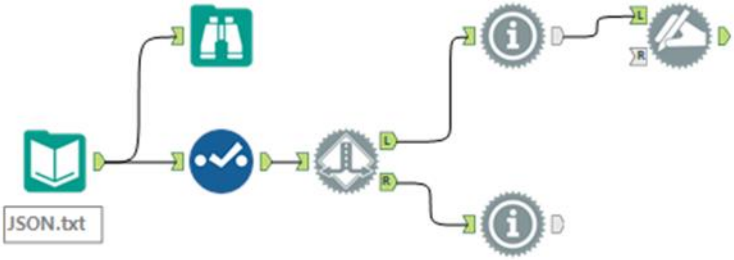


Figure 10-18 – JSON Right Data Stream

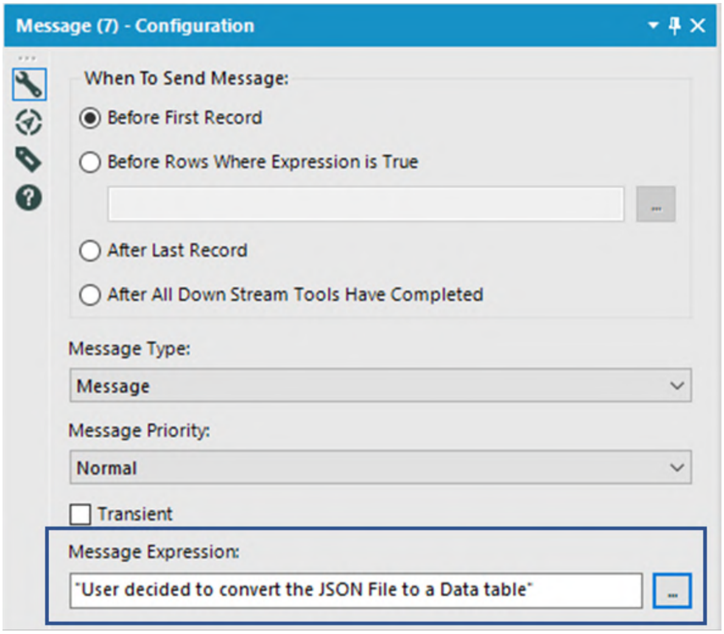


Figure 10-19 – JSON Message Configuration for Right Data Stream

We are going to start the right (R) side of the data stream by creating a parallel message to the left that reads *"User decided to convert the file from JSON to a data table"*.

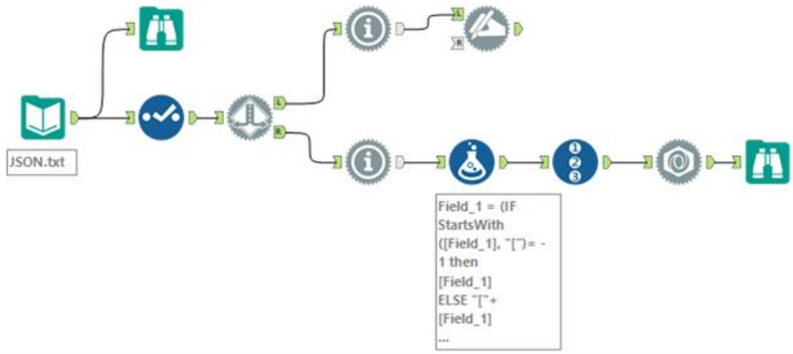


Figure 10-20 – JSON Right Data Stream

We will introduce a *Formula* tool now and convert each row of the incoming data into an array by introducing '[' and ']' symbols at the start and end of each row.

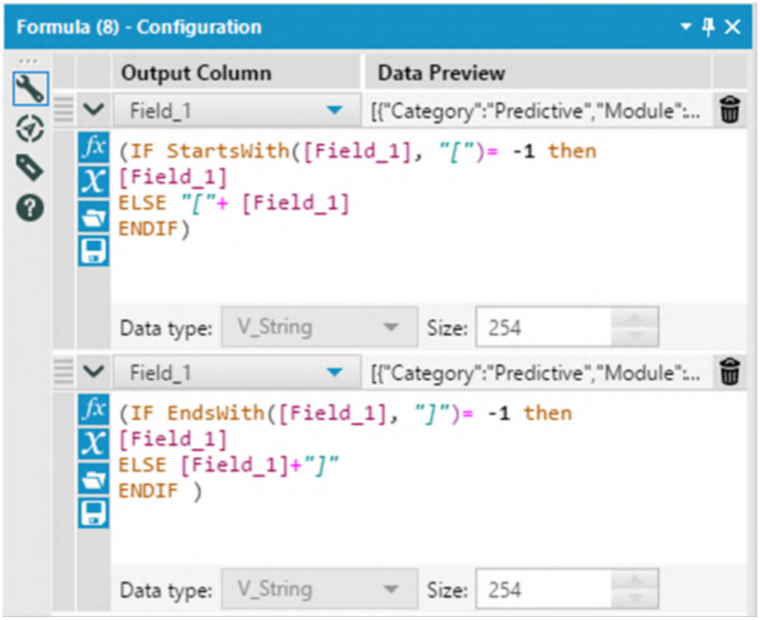


Figure 10-21 – Formula Field Calculation

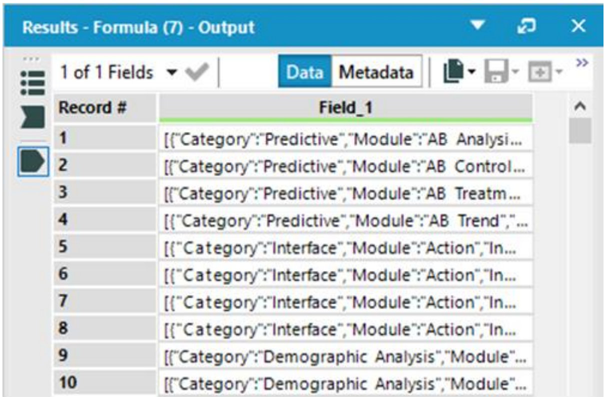


Figure 10-22 – Output after using Formula Tool

We then introduce Record ID tool to number each row which will eventually help us in grouping the data as we shall see. Now, we will bring the *JSON Parse* tool onto the canvas. This tool is designed to look at JSON data and convert it into relationally structured data. Unfortunately, if we just put the tool at the end of the data stream, it does not recognize any fields. This is because the *Detour* is diverting all information to the left, including the metadata. In order to configure the tools on this side of the *Detour,* we need to divert the tool to the right.

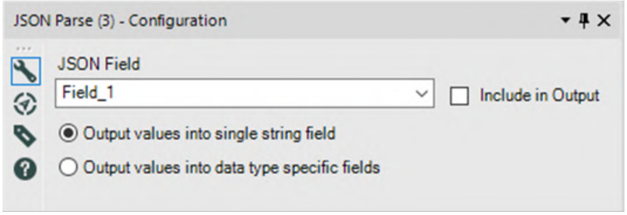
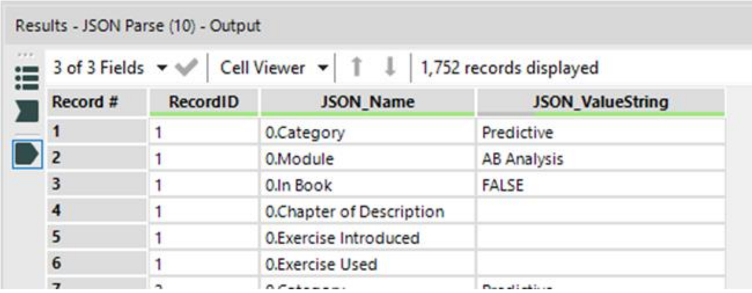


Figure 10-23 – JSON Parse Configuration


Results - JSON Parse (10) - Output

3 of 3 Fields | Cell Viewer | 1,752 records displayed

Record #	RecordID	JSON_Name	JSON_ValueString
1	1	0.Category	Predictive
2	1	0.Module	AB Analysis
3	1	0.In Book	FALSE
4	1	0.Chapter of Description	
5	1	0.Exercise Introduced	
6	1	0.Exercise Used	

Figure 10-24 – Output after using JSON Parse

Now that we have restructured the data, we should take a look at the result so we know what our next steps should be. We can see the JSON Parse tool has created two fields: *JSON_Name*, which has the record number and field name separated by a period, and *JSON_ValueString*, which has the values associated with each record and variable.

Note: If we wanted to convert data into JSON format using the *JSON Build* tool, this is the format it should be in.

Since we know we want a relational dataset that is easy to work with, we should make it tidy by creating a single column for each variable.

In order to do this, we first need to identify the unique column names by splitting them off of the record number. We can use the *Text To Columns* tool in order to create a separate column for both the record number and the column name so we can use the two of them to restructure the data.

Figure 10-25 – Addition of Text to columns in Right Data Stream

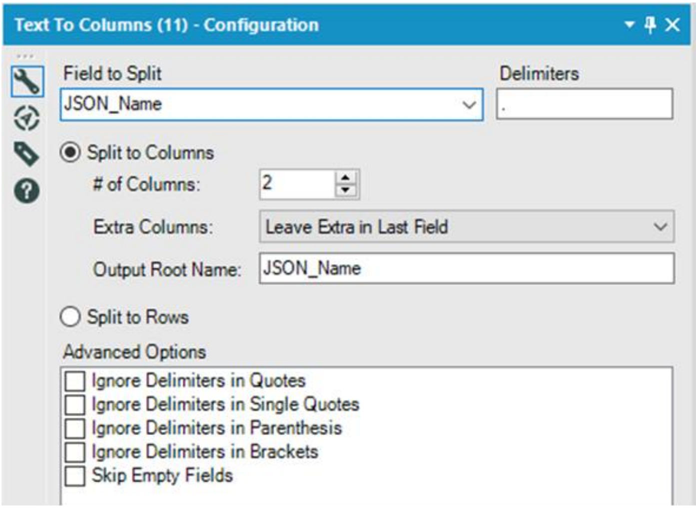


Figure 10-26 – Text to Columns Configuration

In order to make the data easier to understand we should rename our fields.

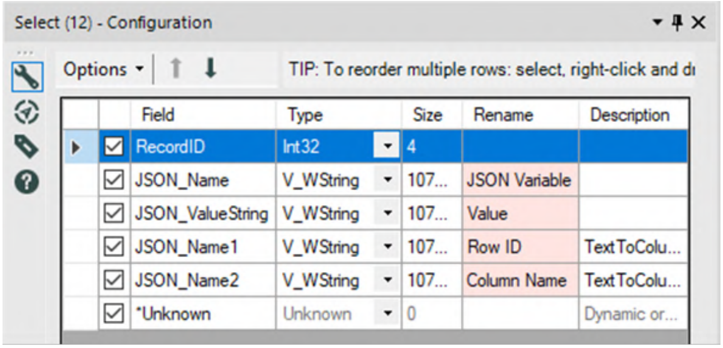


Figure 10-27 – Select tool Configuration for renaming of fields

Record #	RecordID	JSON Name	Value	Row ID	Column Name
1	1	0.Category	Predictive	0	Category
2	1	0.Module	AB Analysis	0	Module
3	1	0.In Book	FALSE	0	In Book
4	1	0.Chapter of Description		0	Chapter of Description
5	1	0.Exercise Introduced		0	Exercise Introduced
6	1	0.Exercise Used		0	Exercise Used

Figure 10-28 – Output After Configuring Select

Now that we have our fields well named, we can pivot the data into a more usable format. We are going to group the data by the *Record ID*, use the *Field Names* as our columns, and use the value field as the data in the intersection. We need to set a methodology. However, since we are using a unique combination of header and grouping fields, we do not need to worry about clashing data.

Figure 10-29 – Addition of Select and Crosstab for Relational Structure

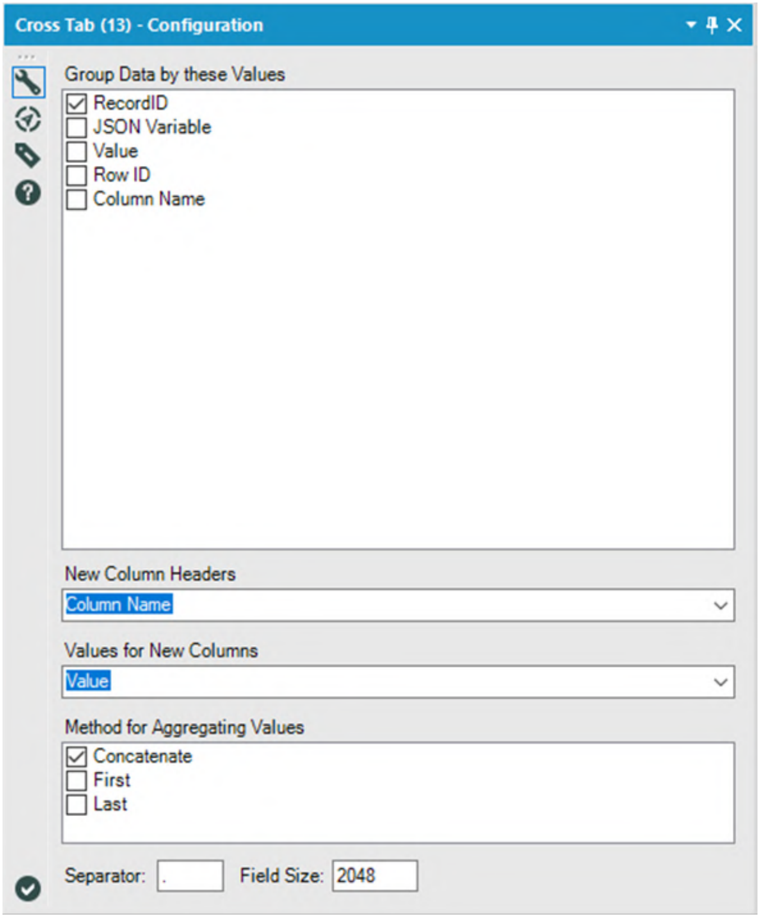


Figure 10-30 – Crosstab Configuration for JSON Data Grouping by Record ID

	Field	Type	Size	Rename	Description
☐	Row ID (Missing)	Int16	2		
☑	Module	String	2048		
☑	Category	V_WString	2048		
☑	Chapter_of_Description	V_WString	2048	Chapter of Description	
☑	Exercise_Introduced	V_WString	2048	Exercise Introduce	
☑	Exercise_Used	V_WString	2048	Exercise Used	
☑	In_Book	Bool	1	In Book	
☑	RecordID	Int32	4		
☑	*Unknown	Unknown	0		Dynamic or Unknown Fields

Figure 10-31 – Select Configuration

Results - Cross Tab (13) - Output

7 of 7 Fields · Cell Viewer · 292 records displayed · Data · Metadata

Record #	RecordID	Category	Chapter_of_Description	Exercise_Introduced	Exercise_Used
1	1	Predictive			
2	2	Predictive			
3	3	Predictive			
4	4	Predictive			
5	5	Interface	Traffic Tickets In Baltimore	Lets Build An App For That!	A Coffee Fix
6	6	Interface	Traffic Tickets In Baltimore	Lets Build An App For That!	A Coffee Fix -- Easy Mapping

Figure 10-32 – Output after including Crosstab

If we take a look at the data, we can see we have a tidy dataset, which means we can end this specialized portion of the workflow and bring the two data streams back together.

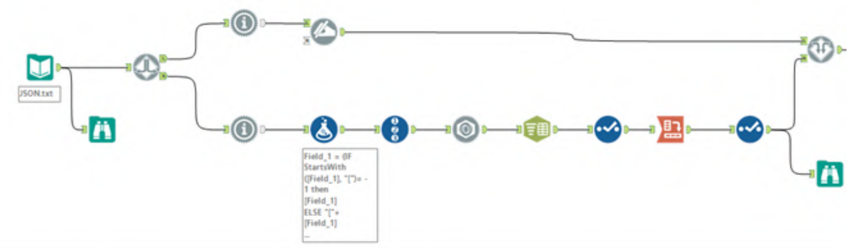


Figure 10-33 – Bringing both data stream together (using Detour End)

Figure 10-34 – Detour End Tool Configuration window

Now we can connect both sides of the data stream to a *Detour End* tool to bring them back together so we can use a single output.

Once we add the Output Data tool, we are almost done. As we can see, we have a fully operational data stream. The only problem we have is that we have no way to control if the file will be written as JSON or a relational table.

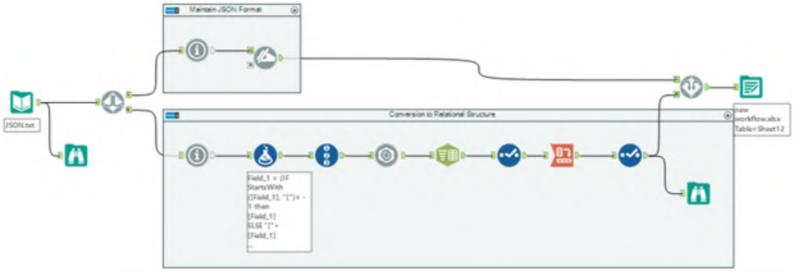


Figure 10-35 – Exporting the file using Output Tool after both Streams joined by Detour

10.3 The Analytical App

What we have done until can also be run in the form an app. To be more specific, an Analytical app. For this we need to add some interface tools, after which the workflow automatically can be used as an Analytical App. We are going to add a *Check Box* and *Action* tool to the workflow so we can give the option of selecting between converting the data or not. If we connect the question directly to the *Detour* tool, the action will be created exactly how we want it to work, except that we should take a look at the behavior of this action because it doesn’t just change the direction, like implied by the action type.

After adding both of the interface tools we will notice that a *Lightning Bolt Anchor* and an *Input Connector (Q in black circle)* will appear on the head of each tool present in the workflow. *Lightning Bolt Anchor* will connect to the Lightning Bolt input of a workflow tool. The connection between these anchors contains the user's value from the interface tool and the method of how it will update the connected workflow tool with that value and *Input Connector (Q in black circle)* accepts the output connections of Interface tools that use the same graphic. This Input connector accepts multiple inputs. It is the value from the incoming tool that will be applied to the Action and sent to update the downstream tool.

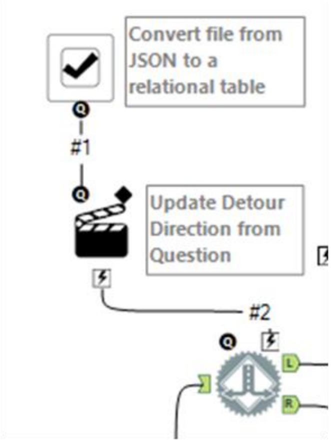


Figure 10-36 – Adding Check Box and Action Tool to Change Detour tool's Configuration

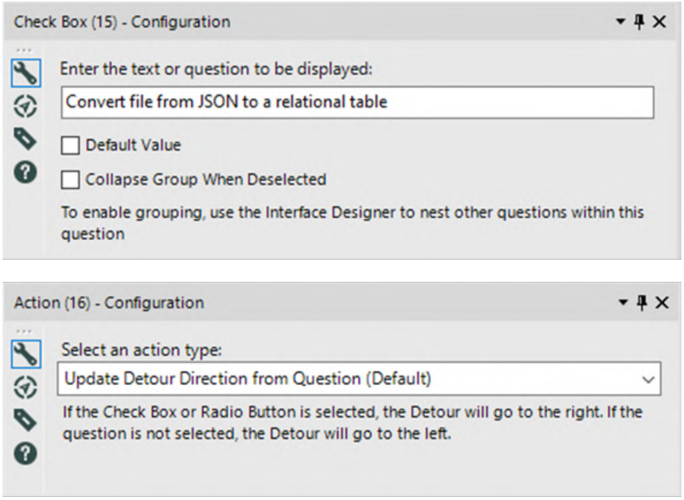


Figure 10-37 – Tool Configuration for Check Box and Action Tool

<u>Properties Window</u>:

The *Check Box Configuration* window has three Components.

- *Enter the text or question to be displayed* is a text box that will be the question prompt.
- *Default Value* can either be checked or unchecked to determine what the default result will be.

- *Collapse Group When Deselected* allows us to hide the relevant group in the interface designer.

We will see an analytical app option in the menu after adding the interface tools.

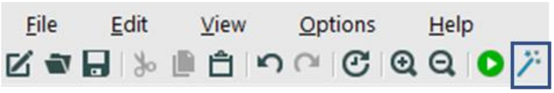


Figure 10-38 – Analytical App Option in the menu bar

Anytime we are running the Detour tool as an application, the default path is to go to the left. When we have an action to update this behavior, it will change the result to divert to the right. We should name the question to reflect this behavior.

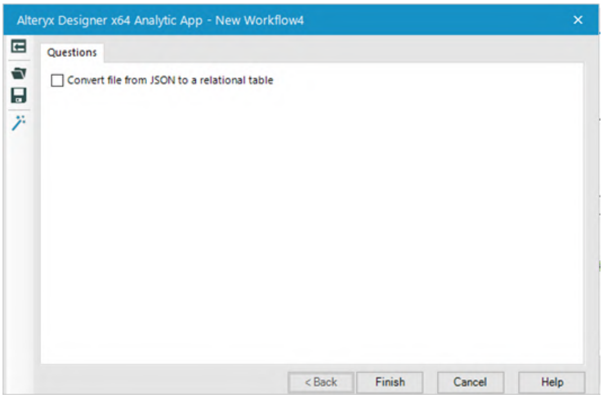
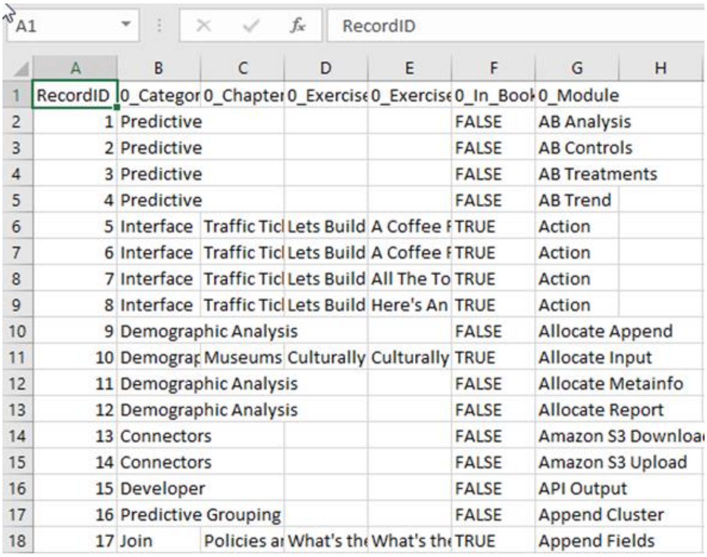


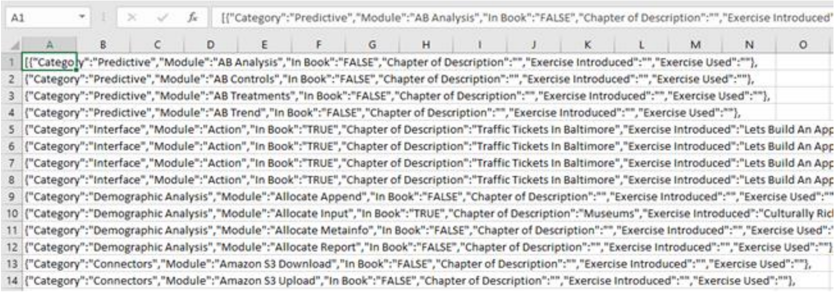
Figure 10-39 – App Dialog after clicking on the Analytical App Option in menu bar



A1 RecordID

	A	B	C	D	E	F	G	H
1	RecordID	0_Categor	0_Chapter	0_Exercise	0_Exercise	0_In_Book	0_Module	
2	1	Predictive				FALSE	AB Analysis	
3	2	Predictive				FALSE	AB Controls	
4	3	Predictive				FALSE	AB Treatments	
5	4	Predictive				FALSE	AB Trend	
6	5	Interface	Traffic Tic	Lets Build	A Coffee F	TRUE	Action	
7	6	Interface	Traffic Tic	Lets Build	A Coffee F	TRUE	Action	
8	7	Interface	Traffic Tic	Lets Build	All The To	TRUE	Action	
9	8	Interface	Traffic Tic	Lets Build	Here's An	TRUE	Action	
10	9	Demographic Analysis				FALSE	Allocate Append	
11	10	Demograp	Museums	Culturally	Culturally	TRUE	Allocate Input	
12	11	Demographic Analysis				FALSE	Allocate Metainfo	
13	12	Demographic Analysis				FALSE	Allocate Report	
14	13	Connectors				FALSE	Amazon S3 Downloa	
15	14	Connectors				FALSE	Amazon S3 Upload	
16	15	Developer				FALSE	API Output	
17	16	Predictive Grouping				FALSE	Append Cluster	
18	17	Join	Policies ar	What's the	What's the	TRUE	Append Fields	

Figure 10-40 – Final Output when Check box in App Dialog window is Checked (Relational Structure)



A1 [{"Category":"Predictive","Module":"AB Analysis","In Book":"FALSE","Chapter of Description":"","Exercise Introduced"

	A
1	[{"Category":"Predictive","Module":"AB Analysis","In Book":"FALSE","Chapter of Description":"","Exercise Introduced":"","Exercise Used":""},
2	{"Category":"Predictive","Module":"AB Controls","In Book":"FALSE","Chapter of Description":"","Exercise Introduced":"","Exercise Used":""},
3	{"Category":"Predictive","Module":"AB Treatments","In Book":"FALSE","Chapter of Description":"","Exercise Introduced":"","Exercise Used":""},
4	{"Category":"Predictive","Module":"AB Trend","In Book":"FALSE","Chapter of Description":"","Exercise Introduced":"","Exercise Used":""},
5	{"Category":"Interface","Module":"Action","In Book":"TRUE","Chapter of Description":"Traffic Tickets In Baltimore","Exercise Introduced":"Lets Build An App
6	{"Category":"Interface","Module":"Action","In Book":"TRUE","Chapter of Description":"Traffic Tickets In Baltimore","Exercise Introduced":"Lets Build An App
7	{"Category":"Interface","Module":"Action","In Book":"TRUE","Chapter of Description":"Traffic Tickets In Baltimore","Exercise Introduced":"Lets Build An App
8	{"Category":"Interface","Module":"Action","In Book":"TRUE","Chapter of Description":"Traffic Tickets In Baltimore","Exercise Introduced":"Lets Build An App
9	{"Category":"Demographic Analysis","Module":"Allocate Append","In Book":"FALSE","Chapter of Description":"","Exercise Introduced":"","Exercise Used":""
10	{"Category":"Demographic Analysis","Module":"Allocate Input","In Book":"TRUE","Chapter of Description":"Museums","Exercise Introduced":"Culturally Ric
11	{"Category":"Demographic Analysis","Module":"Allocate Metainfo","In Book":"FALSE","Chapter of Description":"","Exercise Introduced":"","Exercise Used":"
12	{"Category":"Demographic Analysis","Module":"Allocate Report","In Book":"FALSE","Chapter of Description":"","Exercise Introduced":"","Exercise Used":""}
13	{"Category":"Connectors","Module":"Amazon S3 Download","In Book":"FALSE","Chapter of Description":"","Exercise Introduced":"","Exercise Used":""},
14	{"Category":"Connectors","Module":"Amazon S3 Upload","In Book":"FALSE","Chapter of Description":"","Exercise Introduced":"","Exercise Used":""},

Figure 10-41 – Final Output when Check box in App Dialog window is not Checked (JSON Format Maintained)

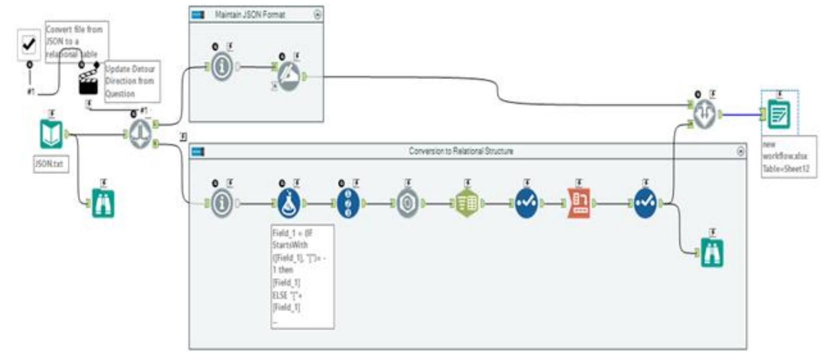


Figure 10-42 – JSON Data Stream on Completion

10.4 Narrowing Down the Search

To... Alteryx Consultants

Subject Narrowing Down The Search

Hey,

This has come along nicely, but here's where it gets complicated; I need you to add a question that will filter the data streams to the specific tool you type into it.

Thanks

CHAPTER 11

Let’s Do It (In-DB)

To... Alterxy Consultants

Subject What’s The Policy On That?

Hey,

As I’m sure you’re aware, we are working for a small company.

Frank has asked me to build a process that helps us stay on top of our new policies. I will be using a software program that, given a link, can automatically open the file or web address that it points to. But first, I need to prep the data.

I’m going to have you sit with me on this so you can see some more of Alteryx’s functionalities.

They only want policies that were published in the last 30 days to show up.

Thanks.

11.1 Tools & Concepts

11.1.1 Browse In-DB

Concept- Viewing data present in the database

 **Figure-11-1** **Browse In-DB**	**Tool Palette: In-Database**
	The Browse In-DB tool allows us to review the In-DB data stream at any point in the workflow. For more details use the link below. bit.ly/2JKjKD0

11.1.2 Connect In-DB

Concept- Connect to a database without bringing in the data (In-DB)

 **Figure-11-2** **Connect In-DB**	**Tool Palette: In-Database**
	The Connect In-DB tool allows us to connect to an In-DB data stream in the workflow. For more details use the link below. bit.ly/2HHd5wZ

11.1.3 Data Stream In

Concept- Bring in external data into the In-DB data stream

 **Figure-11-3** **Data Stream In**	**Tool Palette: In-Database**
	The Data Stream In tool loads a standard input data into In-DB stream by creating a temporary database table to load the data. For more details use the link below. bit.ly/2joL0vT

11.1.4 Data Stream Out

Concept- Bring the data from In-DB stream to external workflow

 **Figure-11-4** **Data Stream Out**	**Tool Palette: In-Database**
	The Data Stream Out tool provides the ability to move data from In-DB Stream to the workflow. For more details use the link below. bit.ly/2rgAyK7

11.1.5 Filter In-DB

Concept- Conditionally filter data present in the In-DB stream

 **Figure-11-5** **Filter In-DB**	**Tool Palette: In-Database**
	The Filter In-DB tool creates a criteria and outputs only those records which match the criteria For more details use the link below. bit.ly/2JMk4Bo

11.1.6 Formula In-DB

Concept- Transformation data in the In-DB stream

 **Figure-11-6** **Formula In-DB**	**Tool Palette: In-Database**
	The Formula In-DB tool allows the In-DB fields to be created/updated using SQL expressions based on the native database language. For more details use the link below. bit.ly/2HI6C4P

11.1.7 Join In-DB

Concept- Join multiple tables within the In-DB stream

 **Figure-11-7** **Join In-DB**	**Tool Palette: In-Database**
	The Join In-DB tool allows joining two database tables from In-DB stream. For more details use the link below. bit.ly/2jlogN5

11.1.8 Macro Input In-DB

Concept- Entry point for macro instructions for In-DB Stream

 Figure-11-8 Macro Input In-DB	**Tool Palette: In-Database** The Macro Input In-DB tool is used to display In-DB input anchors on a macro tool for use in In-DB workflows. For more details use the link below. bit.ly/2IaXODZ

11.1.9 Macro Output In-DB

Concept- Exit point for macro instructions for In-DB Stream

 Figure-11-9 Macro Output In-DB	**Tool Palette: In-Database** The Macro Output In-DB tool is used to display In-DB output anchors on a macro tool for use in In-DB workflows. For more details use the link below. bit.ly/2HIGWFk

11.1.10 Select In-DB

Concept- Select and rename fields present in the In-DB Stream

<table>
<tr><td rowspan="2">
Figure-11-10
Select In-DB</td><td>**Tool Palette: In-Database**</td></tr>
<tr><td>The Select In-DB tool allows manipulation of fields in an In-DB stream by selecting, deselecting, renaming and reordering the fields.
For more details use the link below.
bit.ly/2jmhdnm</td></tr>
</table>

11.1.11 Summarize In-DB

Concept- Aggregate/Roll-up data present in the In-DB Stream

<table>
<tr><td rowspan="2">
Figure-11-11
Summarize In-DB</td><td>**Tool Palette: In-Database**</td></tr>
<tr><td>The Summarize In-DB tool summarize the input data using different aggregate functions.
For more details use the link below.
bit.ly/2rfuvql</td></tr>
</table>

11.1.12 Union In-DB

Concept- Vertically stack datasets present in the In-DB Stream with same table structure

 Figure-11-12 Union In-DB	**Tool Palette: In-Database**
	The Union In-DB tool combines two or more In-DB streams of data which have a similar structure. For more details use the link below. bit.ly/2jmOVcz

11.1.13 Write Data In-DB

Concept- Write to a database from the In-DB Stream

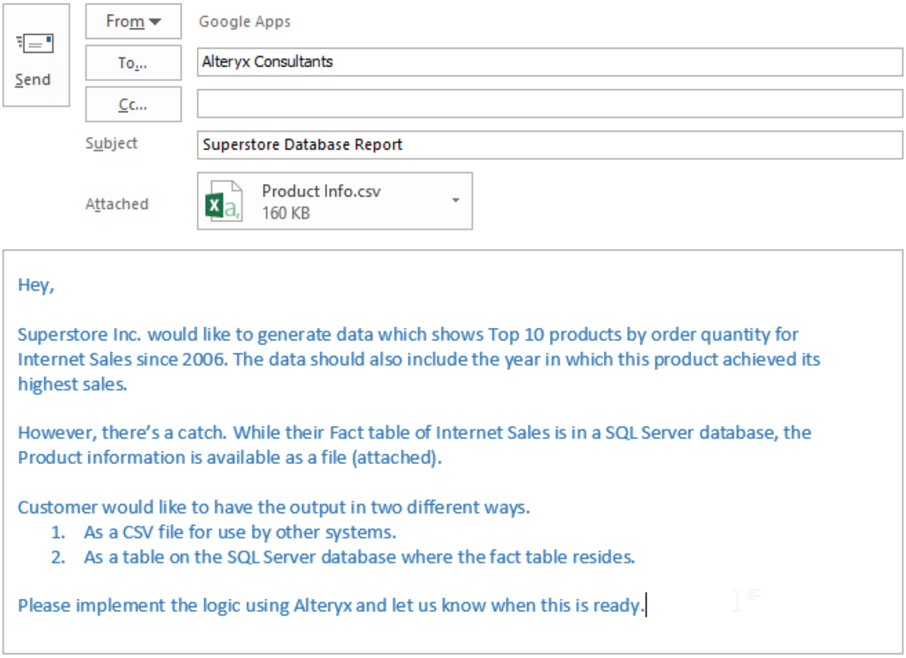
 Figure-11-13 Write Data In-DB	**Tool Palette: In-Database**
	Creates a data stream that has the contents of a Directory or Folder. For more details use the link below. bit.ly/2HLiPSf

11.1.14 Sample In-DB

Concept- Provides selection of data subset

<table>
<tr><td rowspan="2">
Figure-11-14
Sample In-DB</td><td>**Tool Palette: In-Database**</td></tr>
<tr><td>The Sample In-DB tool allows sampling of the In-DB stream based on numbers or percentage of a total number of records.
For more details use the link below.
bit.ly/2rhU0Gm</td></tr>
</table>

11.2 Superstore Database Report


Send

From Google Apps

To... Alteryx Consultants

Cc...

Subject Superstore Database Report

Attached Product Info.csv 160 KB

Hey,

Superstore Inc. would like to generate data which shows Top 10 products by order quantity for Internet Sales since 2006. The data should also include the year in which this product achieved its highest sales.

However, there’s a catch. While their Fact table of Internet Sales is in a SQL Server database, the Product information is available as a file (attached).

Customer would like to have the output in two different ways.

1. As a CSV file for use by other systems.
2. As a table on the SQL Server database where the fact table resides.

Please implement the logic using Alteryx and let us know when this is ready.

Alteryx allows us to use data from different databases using In-database tools the database being SQL Server in this case. Further, the functionality can be extended by importing data from a file input and using it alongside data from the database to produce desired results.

It is important to understand, while using In-DB tools, the data is not imported into the local environment, rather it continues to be processed using the DB’s resources.

We are going to approach our solution by using *Connect In-DB* tool to connect to the Fact table from SQL Server and *Input Data* tool to connect to the flat file which contains Product information. The dataset is available in *Chapter 11- Product Info.csv.*

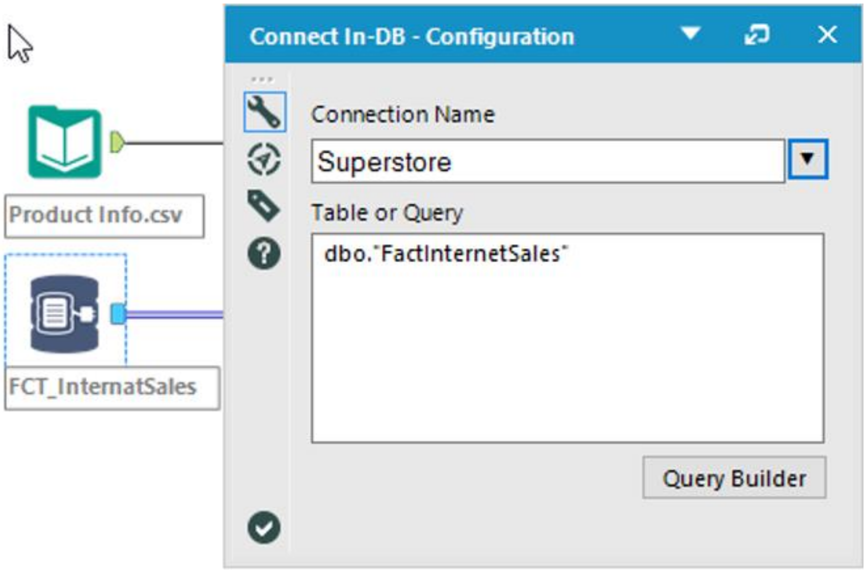


Figure-11-15 Superstore Data, Connect In-DB

We know from the email, the data from the database needs to be considered for 2006 onwards, hence we need to make use of *Filter In-DB* tool to restrict the records accordingly. Further, we need to use a *Data Stream In* tool to convert the file input stream to an In-DB stream. For the purpose of this exercise, we are going to set the *Creation Mode* in the *Data Stream In* tool to *Create Temporary Table*.

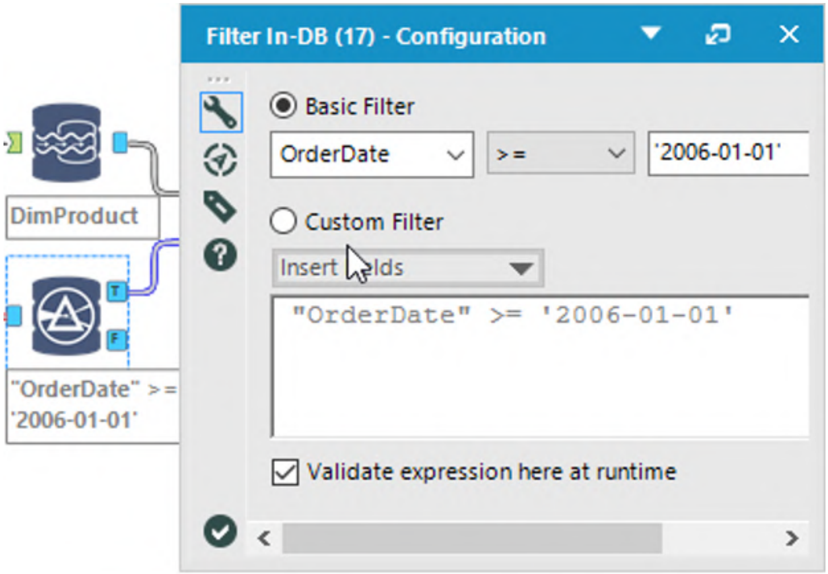


Figure-11-16 Superstore Data, Filter In-DB

Before proceeding further, we need to understand how to bring *Product Name* into the data stream considering this column is not part of the Fact table. The *Product Info.csv* file has a *Product Name* column, which we require, and also the *ProductKey* column which is also present in the database table. This field serves as a common column between the database table and the file. We will join the two data streams by using a *Join In-DB* tool and defining *ProductKey* as the join condition.

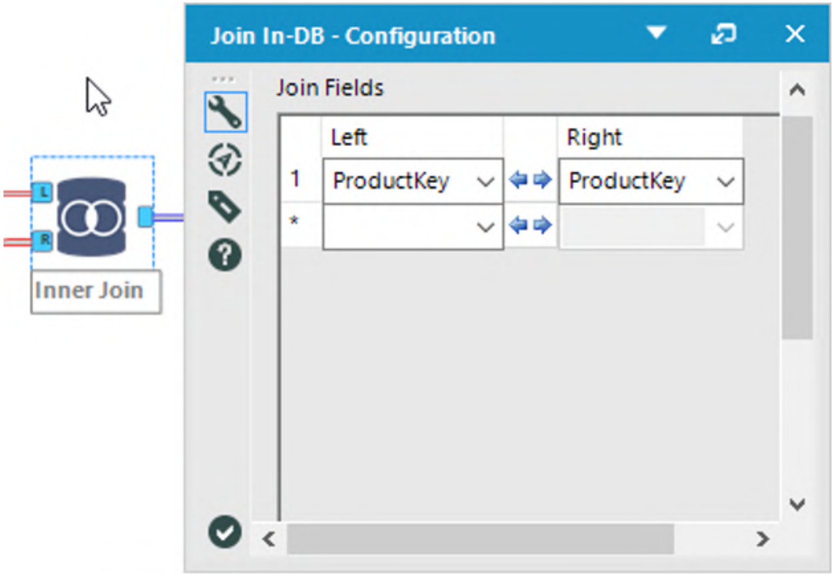


Figure-11-17 Superstore Data, Join In-DB

The output of the *Join In-DB* tool contains many fields, but moving forward we require only a select few of these fields. To achieve this, we will use *Select In-DB* tool to select only relevant columns i.e. *EnglishProductName* renamed as *ProductName*, *OrderQuantity*, and *OrderDate*. Once selected, we will use the *Formula In-DB* tool to calculate the *year of OrderDate*. For the purpose of calculation, we’ll use *YEAR("OrderDate")* as the SQL expression in *Formula In-DB* tool. We’ll rename the new field as *OrderYear* and keep the *Type* as *Double*.

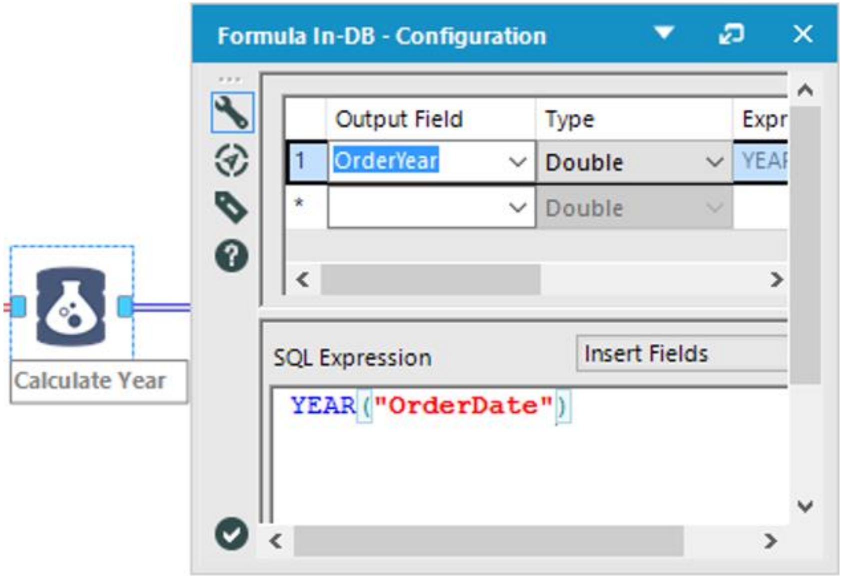


Figure-11-18 Superstore Data, Formula In-DB

After *OrderYear* has been calculated, we are now ready to aggregate *OrderQuantity* to obtain quantity values for each year. To achieve this we will use the *Summarize In-DB* tool. Add *GroupBy* action for *ProductName* and *OrderYear* while adding *Sum* action from the *Numeric* section for *OrderQuantity* field. For *Output* Field Name, keep the auto generated field names intact.

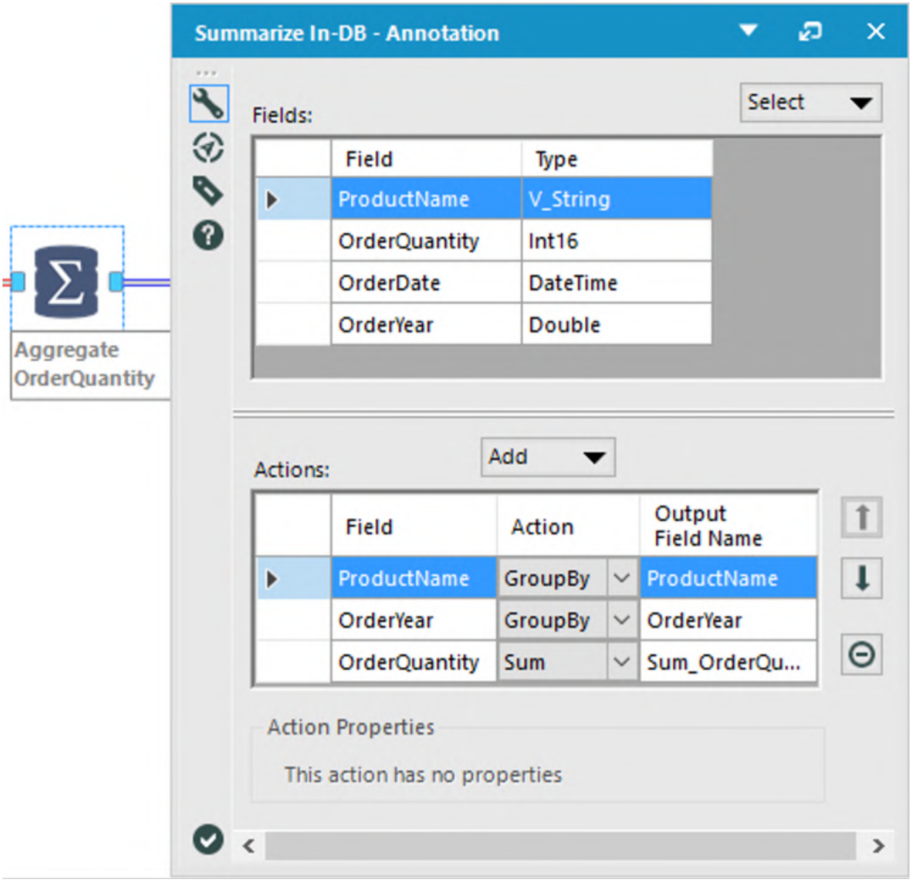
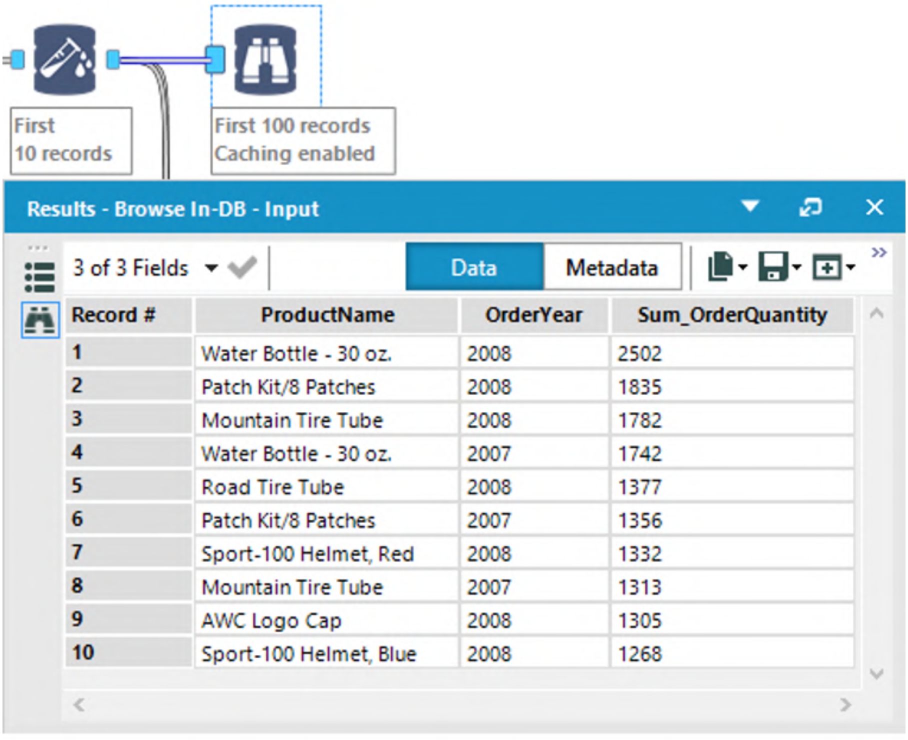


Figure-11-19 Superstore Data, Summarize In-DB

We need to now select only the records with Top 10 order quantities. In order to achieve this, we first need to sort the records in descending order of the sum of order quantity and then select top 10 records from this list. Make use of *Sample In-DB* tool for this purpose by specifying *Number of records to sample* as 10. Also, we need to select the *Sample records based on order* checkbox. Select *Sum_OrderQuantity* under fields section and choose *Descending* as its order.

At this stage, the data should be ready and as per the requirements. However, to confirm this, we have to add *Browse In-DB* tool and then run the workflow to see the output in the *Results* window.



Record #	ProductName	OrderYear	Sum_OrderQuantity
1	Water Bottle - 30 oz.	2008	2502
2	Patch Kit/8 Patches	2008	1835
3	Mountain Tire Tube	2008	1782
4	Water Bottle - 30 oz.	2007	1742
5	Road Tire Tube	2008	1377
6	Patch Kit/8 Patches	2007	1356
7	Sport-100 Helmet, Red	2008	1332
8	Mountain Tire Tube	2007	1313
9	AWC Logo Cap	2008	1305
10	Sport-100 Helmet, Blue	2008	1268

Figure-11-20 Superstore Data, Sample I-DB, Browse In-DB

Now that the data in the *Results* window appears as per the expected requirements, it is now ready for the output. We need to output the results in two formats. First, a *CSV* file to be saved locally and secondly, as a database table. In order to do this, we are going to add *Data Stream Out* and *Write Data In-DB* tool next to *Sample In-DB* tool. While *Data Stream Out* does not need any configuration changes, for *Write Data In-DB*, select *Overwrite Table (Drop)* in *Creation Mode* and mention the table name as *ProductSummary*. Add *Output Data* tool next to *Data Stream Out* tool to write output data to a *CSV* file. Name the file as *ProductSummary.csv*

The completed workflow would look something like the below image.

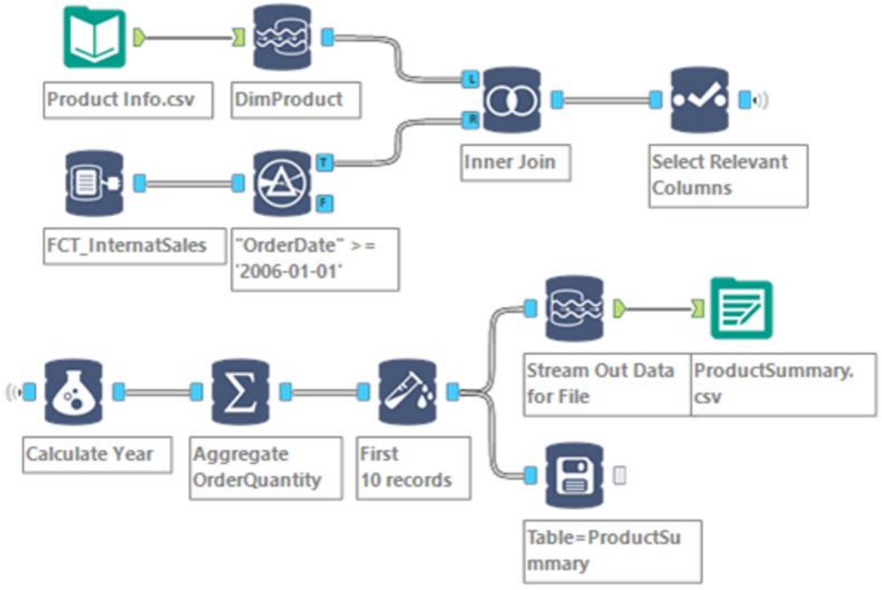
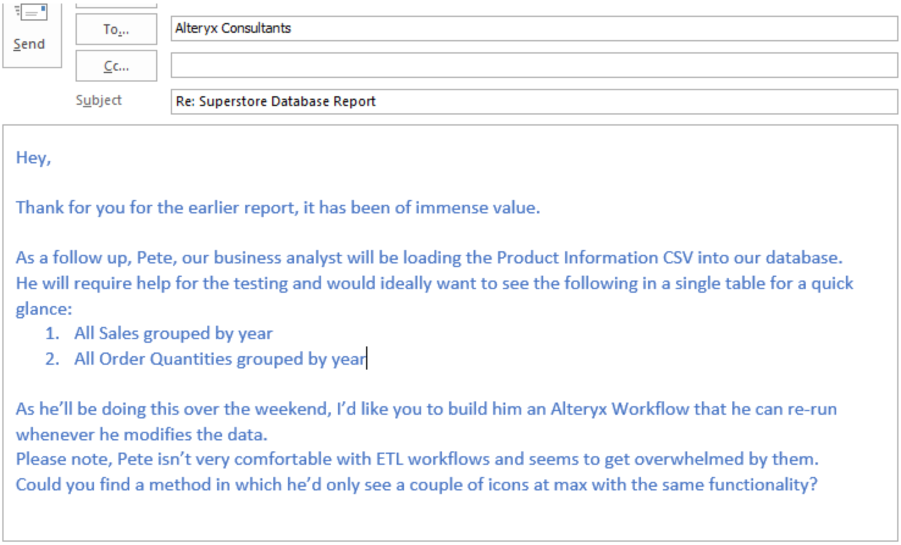


Figure-11-21 Superstore Data, Complete Workflow

ProductName	OrderYear	Sum_OrderQuantity
Water Bottle - 30 oz.	2008	2502
Patch Kit/8 Patches	2008	1835
Mountain Tire Tube	2008	1782
Water Bottle - 30 oz.	2007	1742
Road Tire Tube	2008	1377
Patch Kit/8 Patches	2007	1356
Sport-100 Helmet, Red	2008	1332
Mountain Tire Tube	2007	1313
AWC Logo Cap	2008	1305
Sport-100 Helmet, Blue	2008	1268

Figure-11-22-Superstore Database Report Output

11.3 Superstore Database Report with Macro



Send

To...

Alteryx Consultants

Cc...

Subject

Re: Superstore Database Report

Hey,

Thank for you for the earlier report, it has been of immense value.

As a follow up, Pete, our business analyst will be loading the Product Information CSV into our database.
He will require help for the testing and would ideally want to see the following in a single table for a quick glance:

1. All Sales grouped by year
2. All Order Quantities grouped by year

As he'll be doing this over the weekend, I'd like you to build him an Alteryx Workflow that he can re-run whenever he modifies the data.
Please note, Pete isn't very comfortable with ETL workflows and seems to get overwhelmed by them.
Could you find a method in which he'd only see a couple of icons at max with the same functionality?

The task at hand is very similar to the previous request with the difference being that we'll also be analyzing sales amount by year. We'll also have to condense workflow into a simpler workflow for Pete.

To begin with, let's consider how we'll go about condensing the workflow. Since we want to retain the power of the workflow and yet show only the minimum required workflow icons, we can consider using the concept of *Macros* – and in this case: *Macro In-DB.*

As a starting point, use the *Macro Input In-DB* tool and point it to one of the database tables (in our case we'll point it to *FactInternetSales*). Since the macro will accept one input, we can set the *DimProduct* table (database table corresponding to Product Info.csv) as a source using *Connect In-DB* tool.

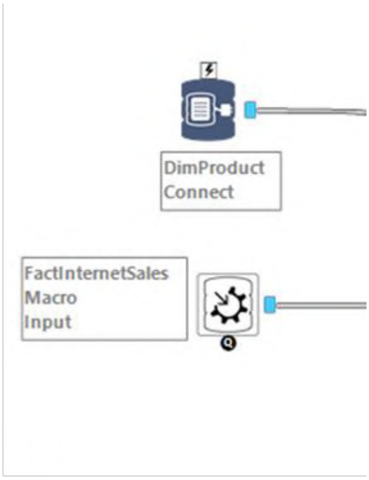


Figure-11-23-Superstore Database Report with Macro Data Stream

Connect the two data streams by joining them on *ProductKey* using the *Join In-DB* tool similar to the previous task.

Figure-11-24-Join In DB

We now need to derive the year from *OrderDate* as well as fix the issue with *SalesAmount* not being a numeric value. For this we use the *Formula In-DB* tool and use the expression *YEAR("OrderDate")* to extract the *OrderYear* and use *SalesAmount* as is but change the *Type* to *Float* to derive the field *SalesAmountValue*.

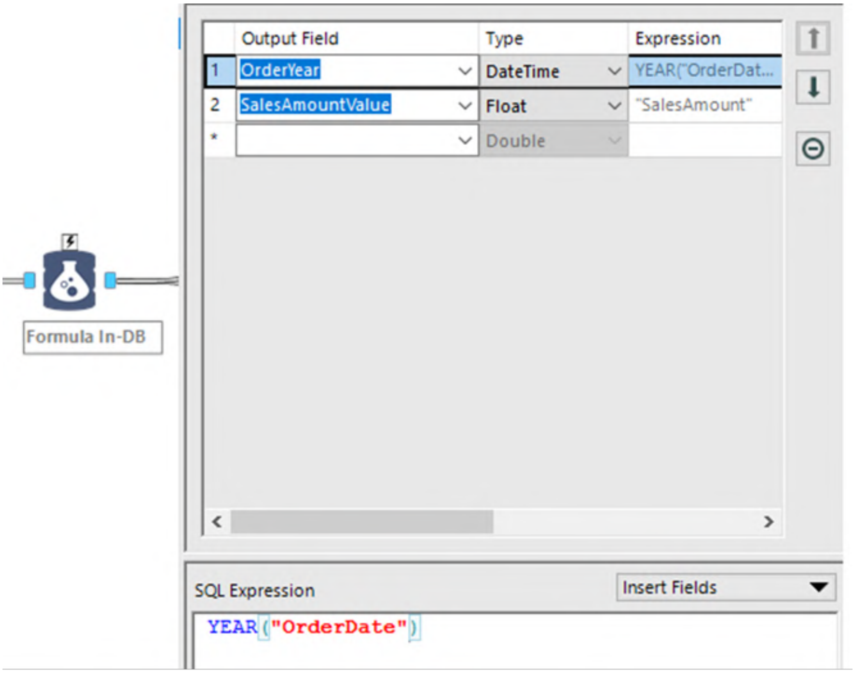


Figure-11-25-Formula In DB Configuration

As per requirement, we'll have to aggregate the *Order Quantity* separately and *SalesAmount* separately by year, and later combine the data into one ouput. For now, let's branch the analysis into two streams: one for quantity and one for sales amount.

The two streams will be similar, both with select *EnglishProductName* and *OrderYear* using the *Select In-DB* tool. The only difference is that the quantity flow will also select *OrderQuantity* field and the Sales Amount flow will also select *SalesAmountValue*. We will be renaming these two numeric fields as *MetricValue* in the *Select In-DB* tool to enable us to combine the data in the future.

Now that we have chosen the select few field we need, we will aggregate the *MetricValue* in their respective workflow. Similar to previous task, add *EnglishProductName* and *OrderYear* to *GroupBy*

Action and add *MetricValue* to *Sum* Action under the *Numeric* option. This will roll up your *MetricValues* to total values per year per product.

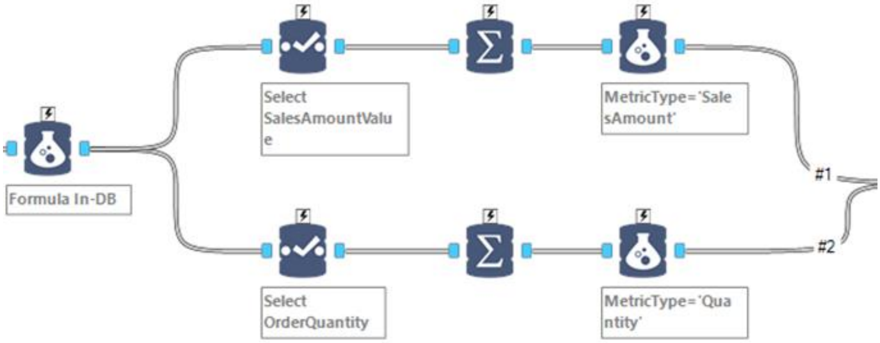


Figure-11-26-Superstore Database Report with Macro Data Stream

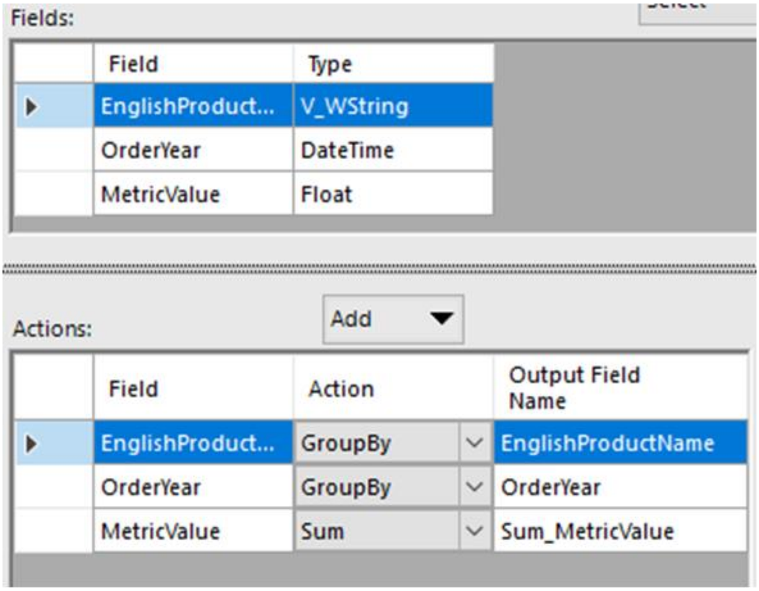


Figure-11-27-Summarize In DB Configuration

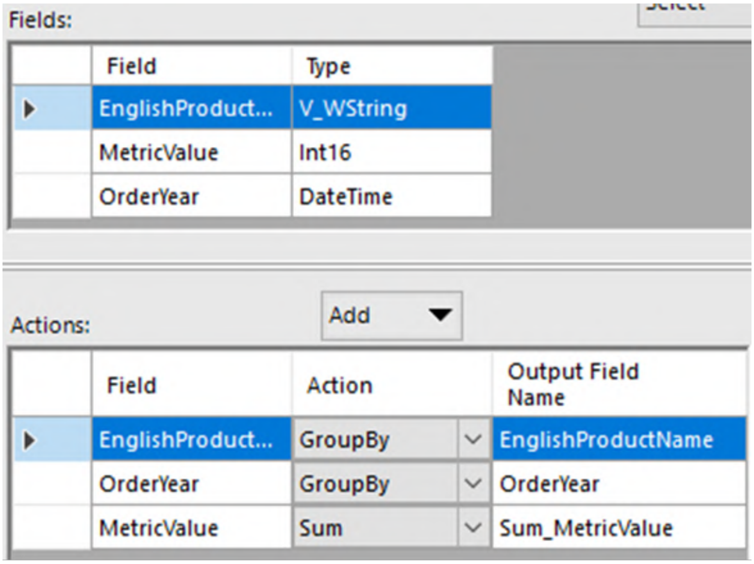


Figure-11-28-Summarize In DB Configuration

Before we go on to combining the two aggregated streams, we need to be able to identify what sort of metric exists at each row. To enable this, we will use a *Formula In-DB* tool to add a fixed string value to each of the streams. The name the field will be *MetricType* and will be fixed to the value *SalesAmount* for the *SalesAmountValue* stream and *Quantity* for the *OrderQuantity* stream.

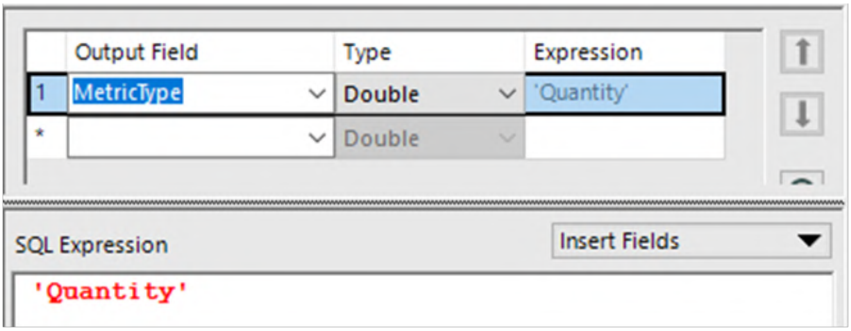


Figure-11-29-Formula In DB Configuration

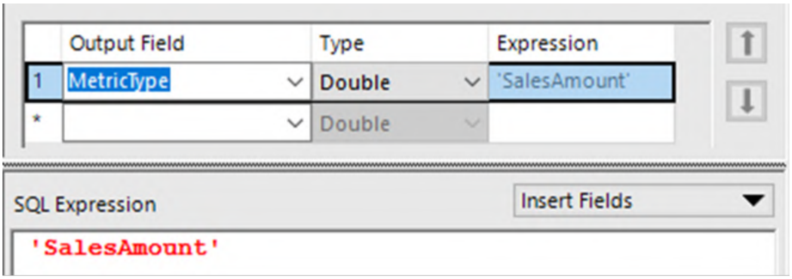


Figure-11-30 Formula IN DB Configuration

You will notice how the two streams now contain four columns with the same names: *EnglishProductName*, *OrderYear*, *MetricValue* and *MetricType*. Having the very same columns lets us easily combine the two datasets using the *Union In-DB* tool. This tool reads columns from both the datasets and stacks the data from similar column names on top of each other. While using *Union In-DB* tool for this workflow, you can let the default tool configuration remain.

Finally, as we have out final output, we can exit the macro using the *Macro Output In-DB*. Proceed by saving the file with a *.yxmc* extension which identifies it as a macro file.

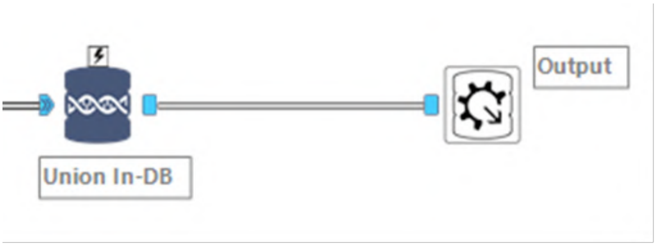


Figure-11-31-Union IN DB and Macro Output Data Stream

The final macro workflow should look like the workflow below (some connections have been made wireless for clarity).

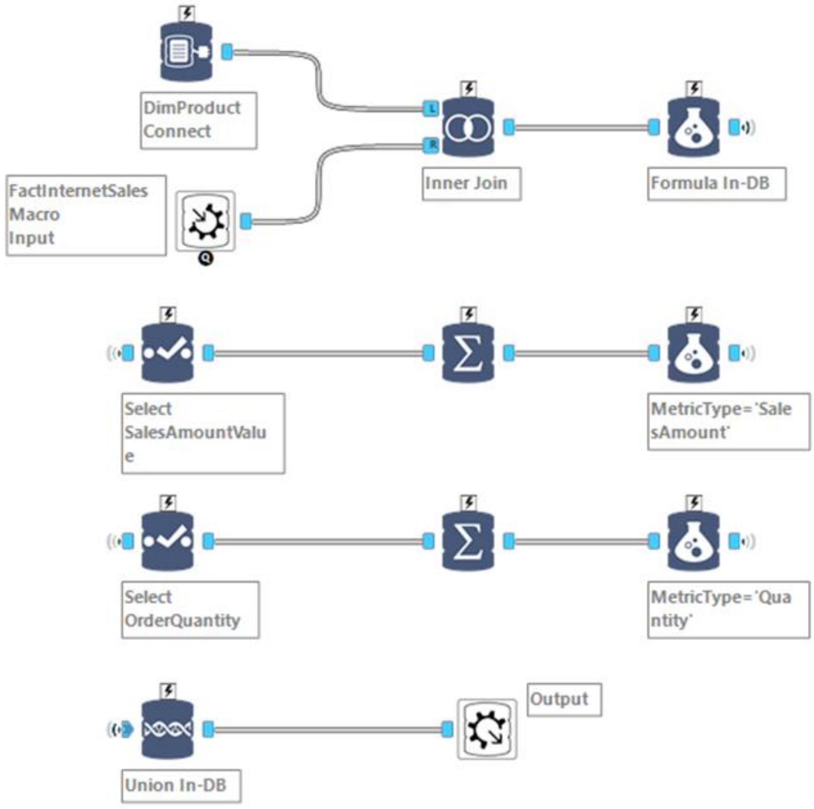


Figure-11-32-Superstore Database Report with Macro Data Stream when Complete

Start a new workflow file of the extension type. yxmd. This will be the main workflow Pete will interact with. Add the *Connect In-DB* tool to the workflow and point it to *FactInternetSales*. This will mirror the fields that the *Macro Input In-DB* tool had earlier been configured to read.

Right click anywhere on the workflow screen and select *Insert*>*Macro*. You should be able to see the name of the macro created by you here. Select the macro name and it should add a nondescript circular icon to your workflow. This encapsulates the entire logic written earlier in the macro file. Finally, add a *Browse In-DB* to the workflow to enable Pete to see the results on refresh.

This simple workflow should look like the workflow below.

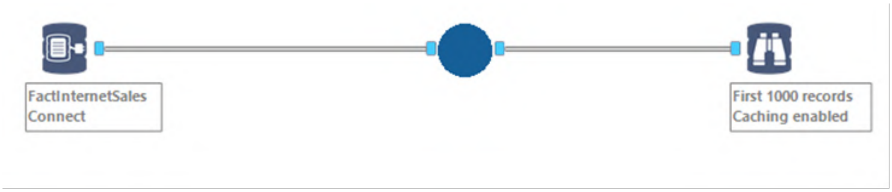


Figure-11-33-Superstore Database Report with Macro Data Stream with Macro UI when Complete

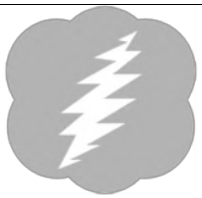
CHAPTER 12

GREEN ON THE GO

12.1 Tools & Concepts

12.1.1 Download

Concept- Retrieve data from internet/intranet environment

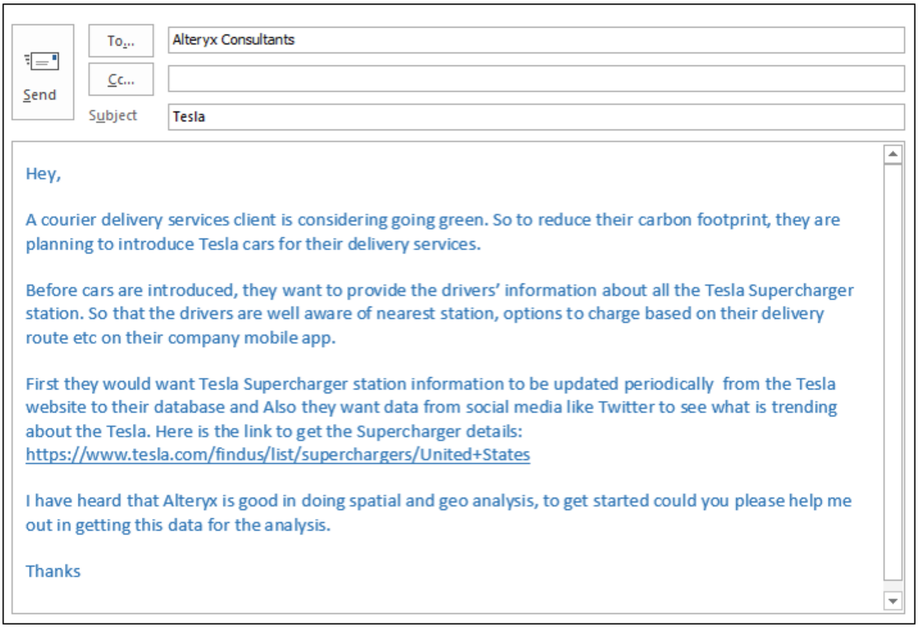
 Figure-12-1-Download	**Tool Palette: Connectors**
	It will retrieve data from a specified URL to be used in downstream processing or to be saved to a file. For more details use the link below. bit.ly/2K9tKqq

12.1.2 Twitter Search

Concept- Search Tweets

 Figure-12-2-Twitter Search	**Tool Palette: Connectors**
	It allows you to search tweets by given search terms, with the location as an optional property. The search will only retrieve tweets from the previous seven days. For more details use the link below. bit.ly/2HKVkw2

12.2 Green on the go



Send | To... Alteryx Consultants | Cc... | Subject Tesla

Hey,

A courier delivery services client is considering going green. So to reduce their carbon footprint, they are planning to introduce Tesla cars for their delivery services.

Before cars are introduced, they want to provide the drivers' information about all the Tesla Supercharger station. So that the drivers are well aware of nearest station, options to charge based on their delivery route etc on their company mobile app.

First they would want Tesla Supercharger station information to be updated periodically from the Tesla website to their database and Also they want data from social media like Twitter to see what is trending about the Tesla. Here is the link to get the Supercharger details:
https://www.tesla.com/findus/list/superchargers/United+States

I have heard that Alteryx is good in doing spatial and geo analysis, to get started could you please help me out in getting this data for the analysis.

Thanks

Based on the request to create the supercharger station data for the spatial and geo analysis, we will start looking at how to pull the information which is available on the Tesla website. Read the instructions file before using these tools. The file is at *Chapter 12-Instructions*. First, let us list down what we should be doing to extract the supercharger station data for the website in a specific format.

1. Get the URL of Tesla supercharger station information web page.
2. Look at the webpage, see what data you would extract for analysis. Like,

   Supercharger Station Name, Street Address, State,
   Zip and Roadside assistance

3. Download the page on Alteryx.
4. Parse the data from the above data points
5. Prepare and filter the data

We already know the URL where we can find the Tesla supercharger station information

URL:https://www.tesla.com/findus/list/superchargers/United+States

Let us use the URL in Text Input and pass the URL to the Download tool.

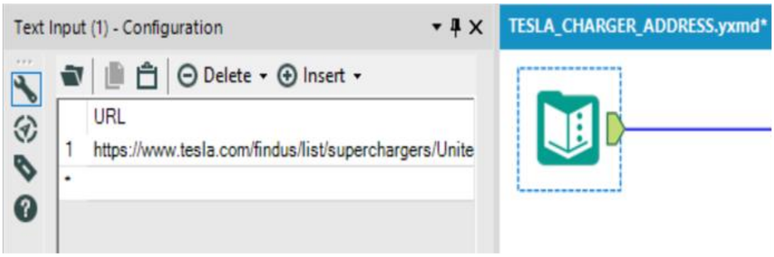


Figure-12-3-Green On the Go-Text Input

Download tool takes the URL as input and retrieves data from a specified URL. We will configure the download tool to output data as a string. This option returns the data as a new wide string type field. A wide string supports Unicode characters.

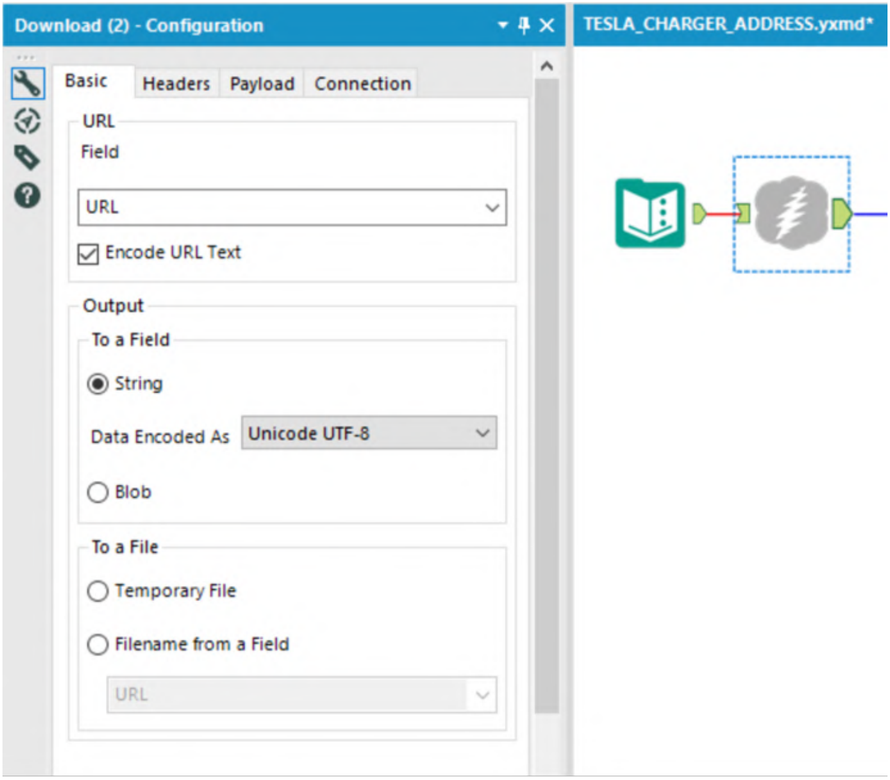


Figure-12-4-Green On the Go-Download Configuration

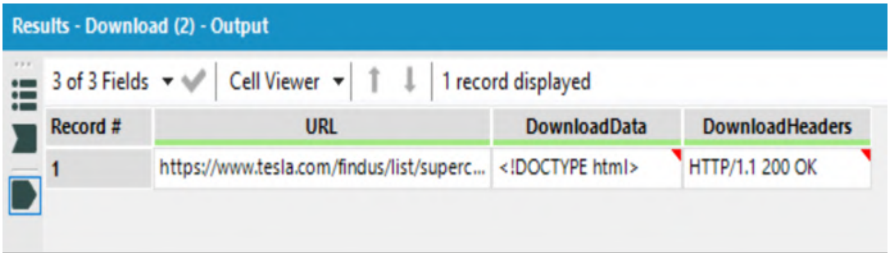


Figure-12-5-Green On the Go-Download Output

So now we have the result in a string or the text. Next step is to convert the string or the text to rows. We use the Text to column or row tool for this task.

At this stage running process shows us the following,

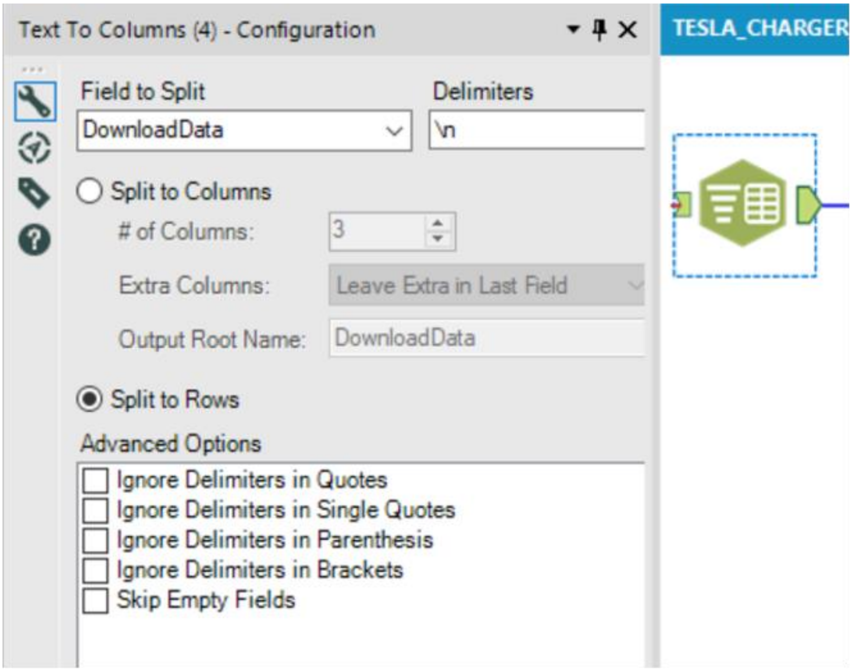


Figure-12-6-Green On the Go-Text to Column Configuration

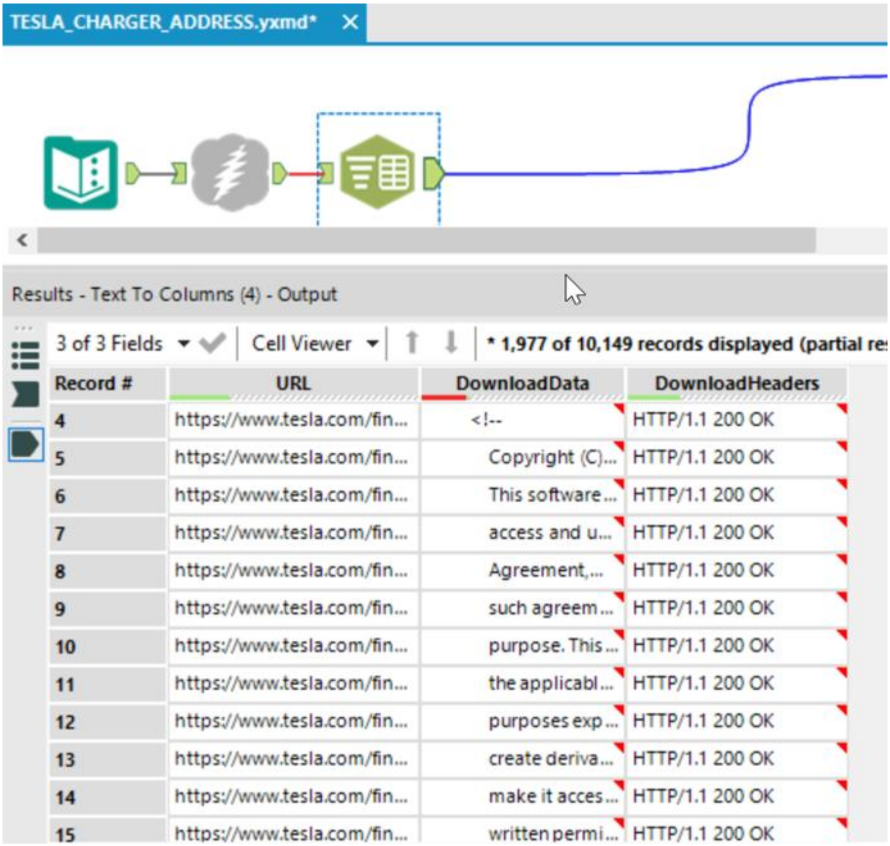


Figure-12-7-Green On the Go-Text to Column Output

After converting the data to rows, the challenge is to get rid of the unwanted rows which were part of the string/text converted from the URL.

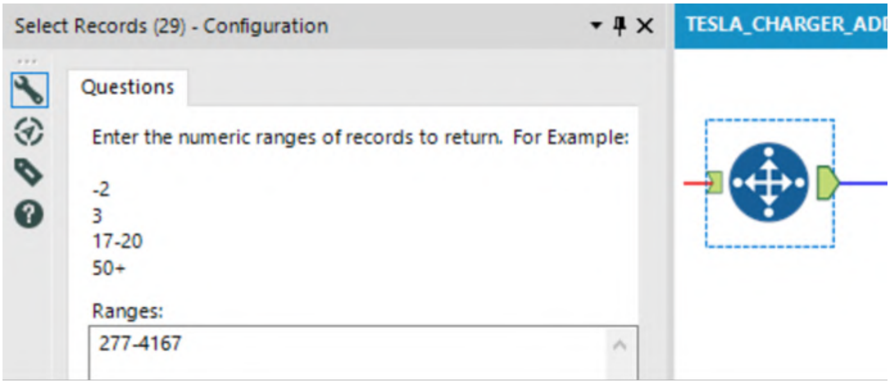


Figure-12-8-Green On the Go-Select Records Configuration

Using the select record, we will select the records from the row number 277-4167. Looking at the previous output we found the first 276 rows does not contain the information that we want. So, we are filtering them out for the data stream.

Now is time to clean the data further. Using the data cleansing tool, let us do few cleansing operations.

Like,

- Replacing NULL with Blank for the string data
- Replacing NULL with 0 for the numeric data
- Removing unwanted characters like leading and trailing whitespaces, Tabs, Line Breaks etc.

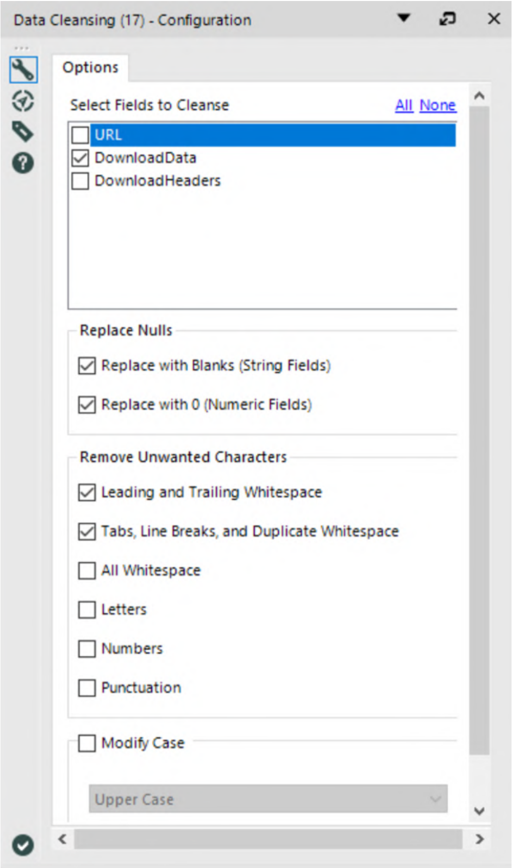


Figure-12-9-Green On the Go-Data Cleansing Configuration

Now we got a lot cleaner data.

Next step would be to create a mapping Create a mapping table using text input. Mapping table will have the data nodes that we would like to extract.

Roadside Assistance: <span class="value">

The above data nodes are mapped to the data points,

Supercharger Station Name

Street Address

State & Zip

Roadside assistance

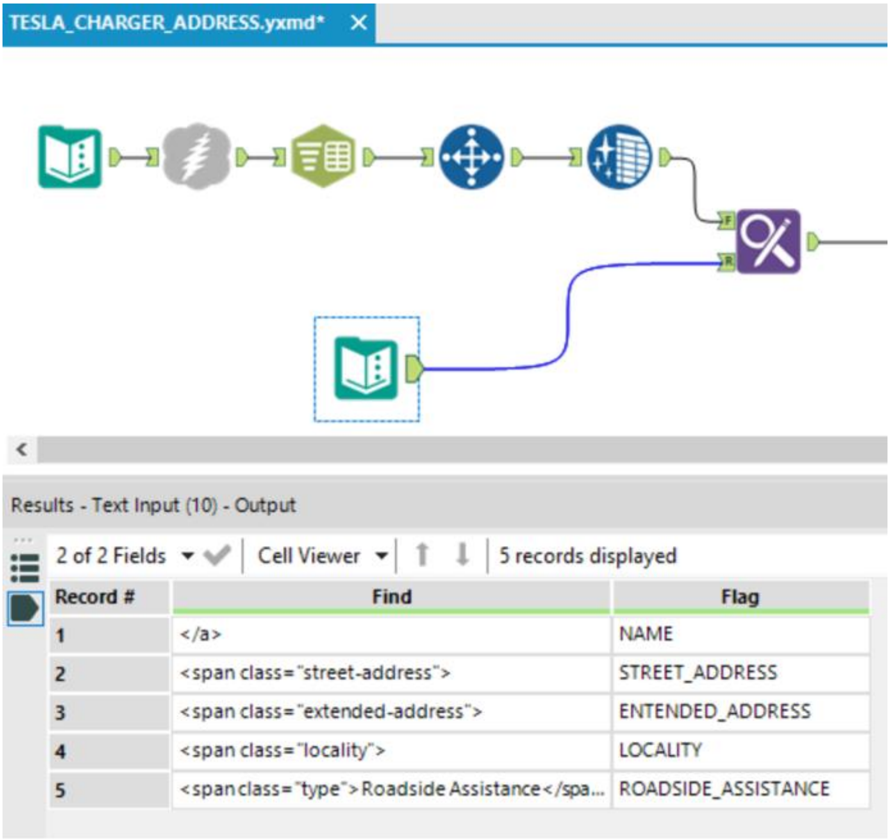


Figure-12-10-Green On the Go-Text Input

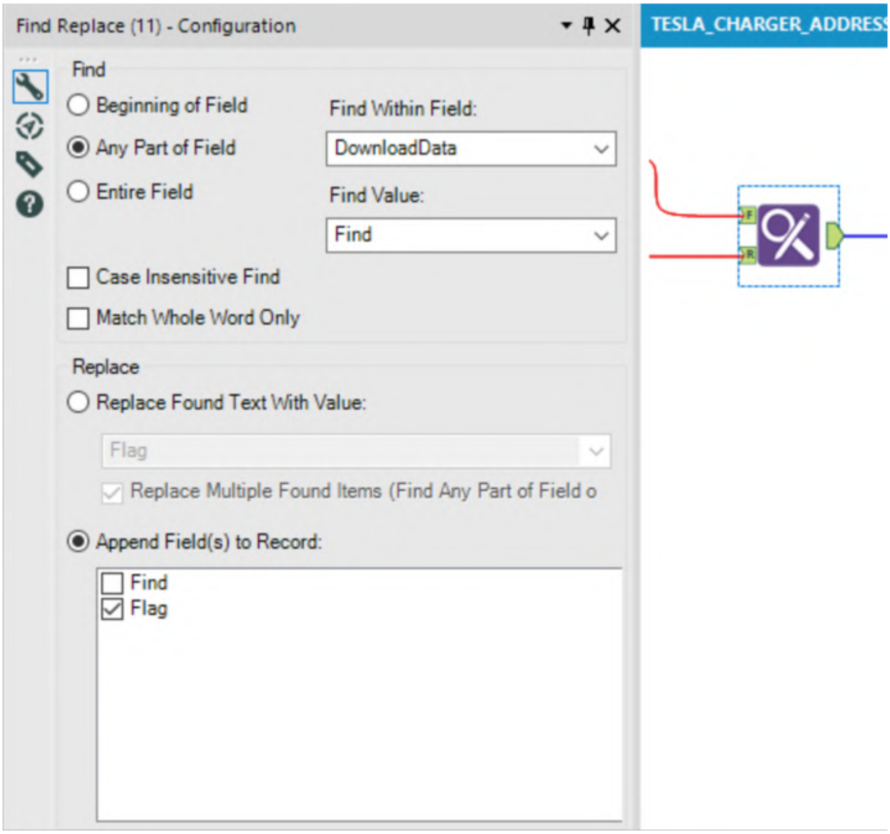


Figure-12-11-Green On the Go-Find Replace Configuration

Filter the rows containing the data for the data point mentioned above.

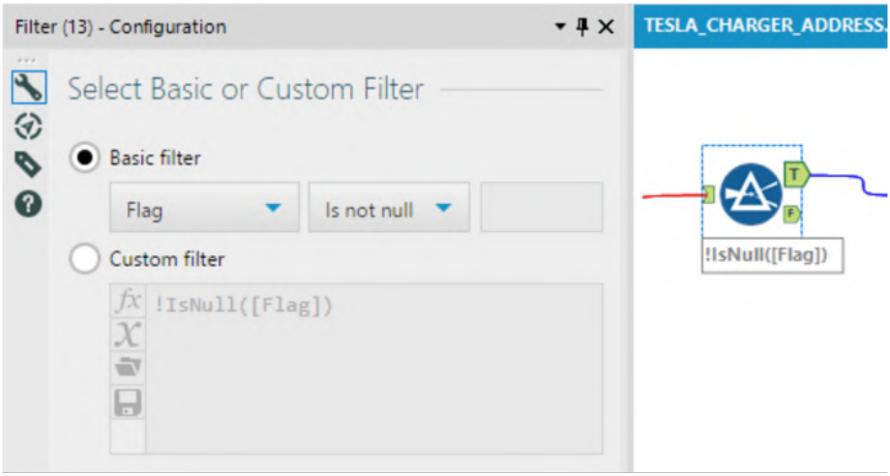


Figure-12-12-Green On the Go-Filter Configuration

We have the data for Supercharger Station Name, Street Address, State & Zip and Roadside assistance

Since we have all the information which was requested, now time to clean up the unwanted HTML tags in the row and assign the unique ID to each Supercharger Station records.

Next 2 tools Multi-Row Formula and Formula tools will help us in assigning a unique number with a new column ID and removing unwanted HTML tags.

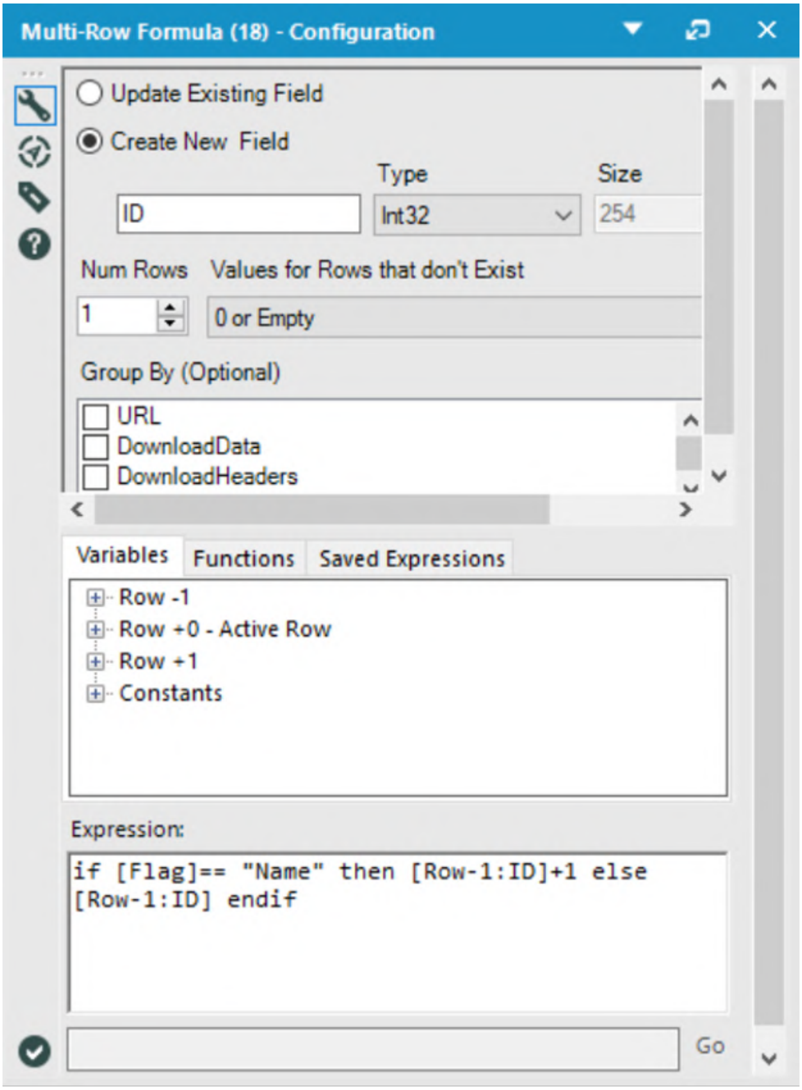


Figure-12-13-Multi Row Formula Configuration

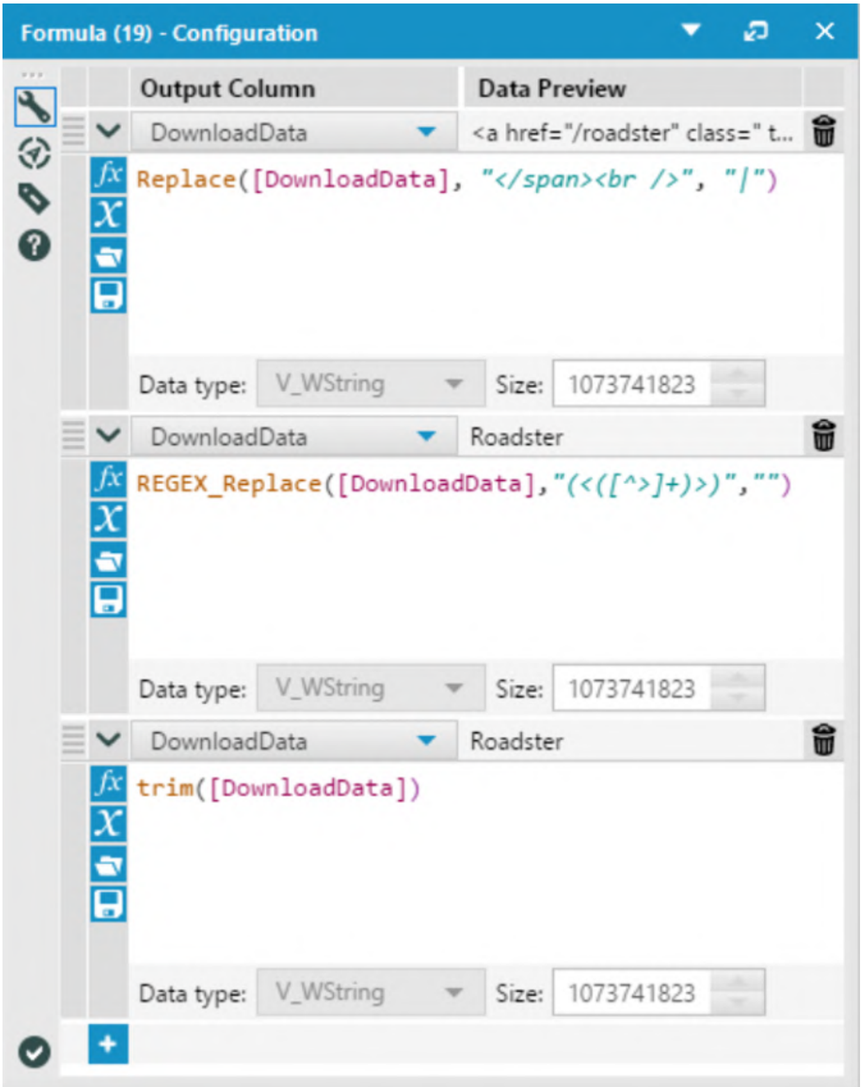


Figure-12-14-Formula Configuration

Till now we were massaging Supercharger Station data which were in rows. So now we would like to convert the relevant information as a column.

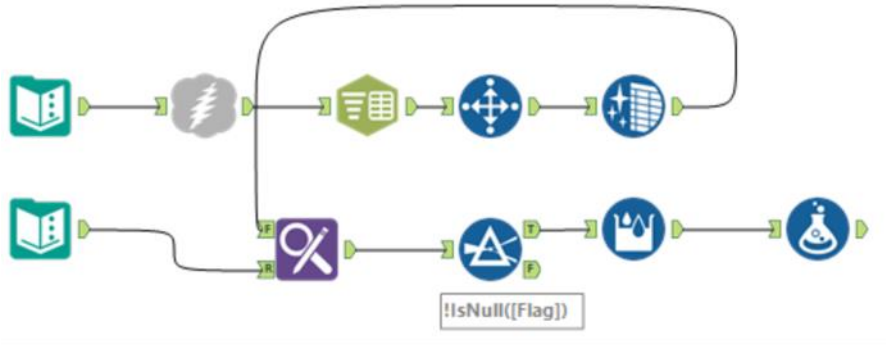


Figure-12-15-Green On the Go Data Stream

ID column which we just generated will help us to do a group for Supercharger Station and convert other information like the address, state, and zip into respective columns. We will use Cross Tab tool to reshape the data stream.

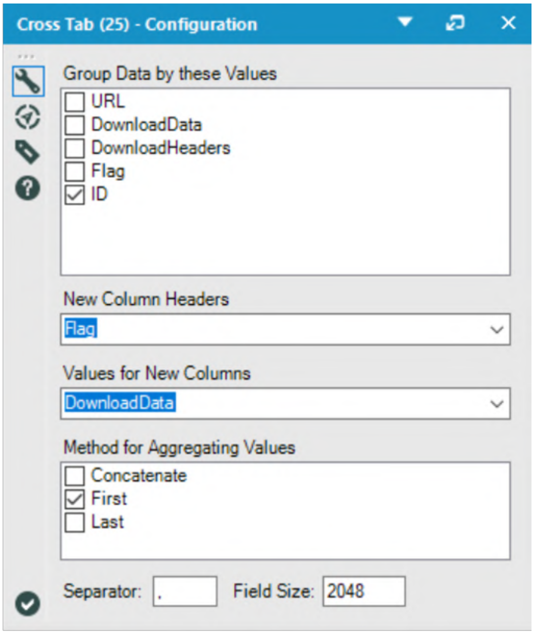


Figure-12-16-Cross Tab Configuration

The last step is to get rid of the unnecessary column and renaming the output column as required.

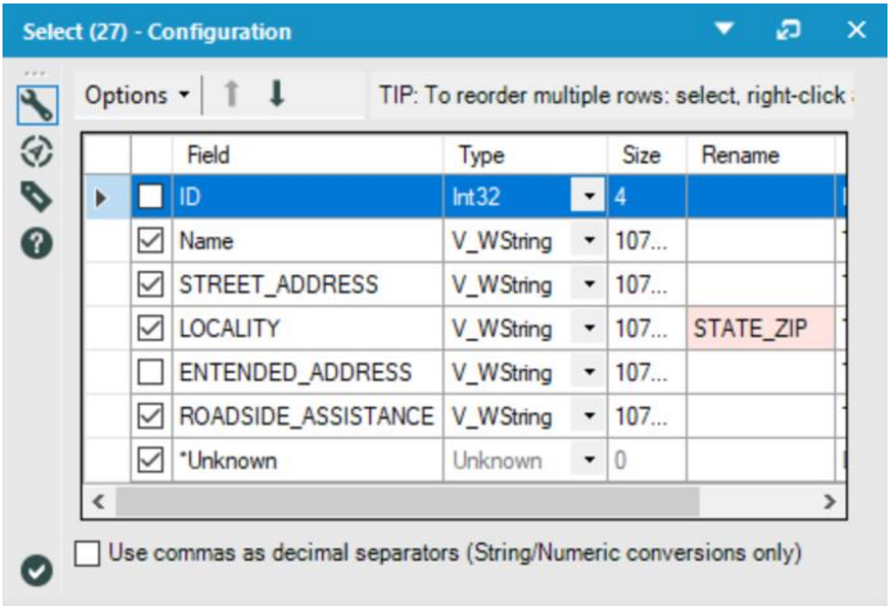
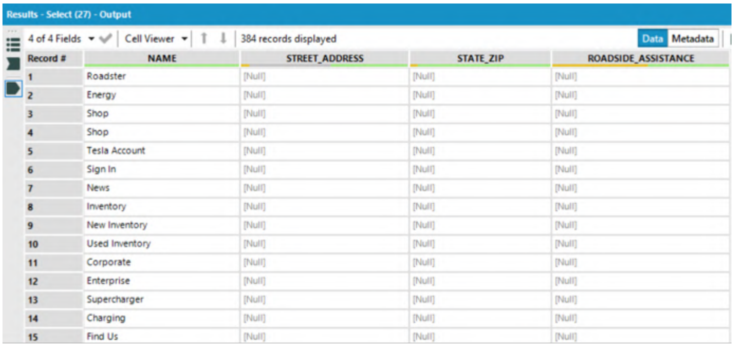


Figure-12-17-Select Configuration

Select Tool Output shows 23 rows with null data. So, we use Sample Tool to delete the null rows.



Record #	NAME	STREET_ADDRESS	STATE_ZIP	ROADSIDE_ASSISTANCE
1	Roadster	[Null]	[Null]	[Null]
2	Energy	[Null]	[Null]	[Null]
3	Shop	[Null]	[Null]	[Null]
4	Shop	[Null]	[Null]	[Null]
5	Tesla Account	[Null]	[Null]	[Null]
6	Sign In	[Null]	[Null]	[Null]
7	News	[Null]	[Null]	[Null]
8	Inventory	[Null]	[Null]	[Null]
9	New Inventory	[Null]	[Null]	[Null]
10	Used Inventory	[Null]	[Null]	[Null]
11	Corporate	[Null]	[Null]	[Null]
12	Enterprise	[Null]	[Null]	[Null]
13	Supercharger	[Null]	[Null]	[Null]
14	Charging	[Null]	[Null]	[Null]
15	Find Us	[Null]	[Null]	[Null]

Figure-12-18-Select Tab Output

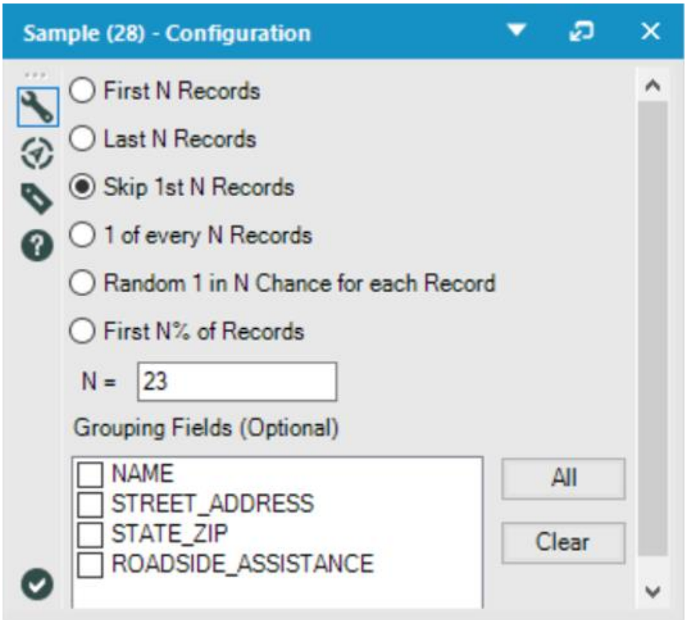
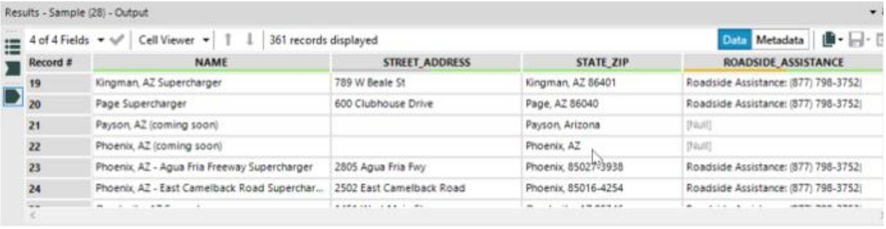


Figure-12-19-Sample Configuration

One last time, make sure that there are no NULL rows in the data stream. We find many rows with null value. So, we use data cleansing tool to clean the data.



Record #	NAME	STREET_ADDRESS	STATE_ZIP	ROADSIDE_ASSISTANCE
19	Kingman, AZ Supercharger	789 W Beale St	Kingman, AZ 86401	Roadside Assistance: (877) 798-3752|
20	Page Supercharger	600 Clubhouse Drive	Page, AZ 86040	Roadside Assistance: (877) 798-3752|
21	Payson, AZ (coming soon)		Payson, Arizona	[Null]
22	Phoenix, AZ (coming soon)		Phoenix, AZ	[Null]
23	Phoenix, AZ - Agua Fria Freeway Supercharger	2805 Agua Fria Fwy	Phoenix, 85027-3938	Roadside Assistance: (877) 798-3752|
24	Phoenix, AZ - East Camelback Road Superchar...	2502 East Camelback Road	Phoenix, 85016-4254	Roadside Assistance: (877) 798-3752|

Figure-12-20-Sample Output

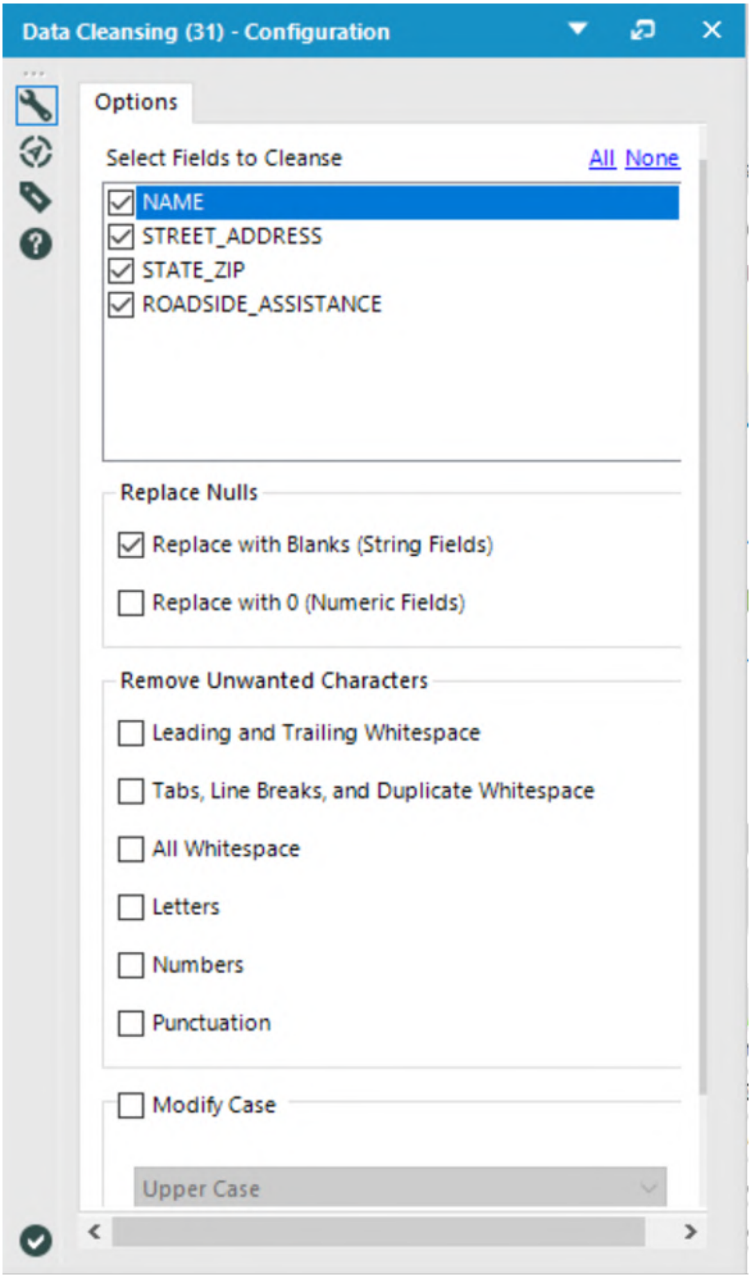


Figure-12-21-Data Cleansing Configuration

Here, we are replacing all the nulls with blanks. The data cleansing tool has other options as well, such as replacing with 0. Removing unwanted characters etc.

That's it. We have the data in a required format for the spatial and geo analysis.

Results - Data Cleansing (31) - Output26

4 of 4 Fields | Cell Viewer | 361 records displayed

Record #	NAME	STREET_ADDRESS	STATE_ZIP
1	Athens, AL Supercharger	21282 Athens-Limestone Blvd	Athens, AL
2	Auburn Alabama Supercharger	1627 Opelika Road	Auburn, AL 36830
3	Birmingham, AL Supercharger	2221 Richard Arrington Junior Blvd	Birmingham, AL 35203-1103
4	Dothan, AL (coming soon)		Dothan, Alabama
5	Greenville Supercharger	219 Interstate Drive	Greenville, AL 36037
6	Mobile Supercharger	3201 Airport Blvd	Mobile, AL 36606
7	Montgomery, AL (coming soon)		Montgomery, AL
8	Oxford, AL Supercharger	1105 Oxford Exchange Blvd	Oxford, AL 36203-0000
9	Steele Supercharger	905 Steele Station Road	Steele, AL 35987
10	Tuscaloosa, AL (coming soon)		Tuscaloosa, Alabama
11	Buckeye, AZ Supercharger	416 S Watson Rd	Buckeye, AZ 85326

Figure-12-22-Data Cleansing Output

Below is the full view of the workflow that we just created.

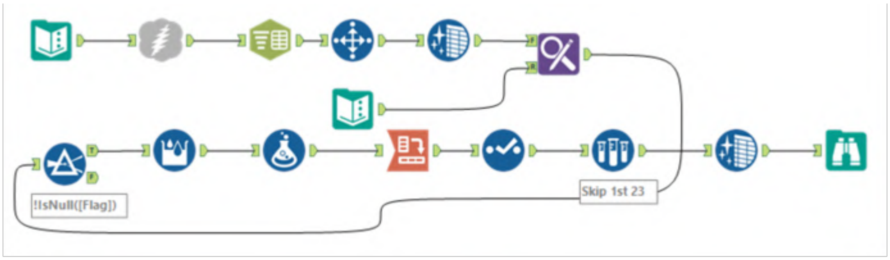
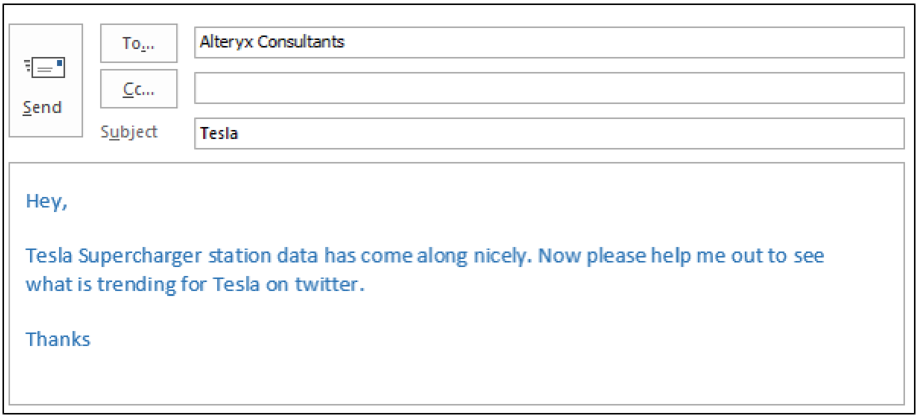


Figure-12-23-Green On the Go-Data Stream on Completion

12.3 What is trending for Tesla?


Send
To...
Alterxy Consultants
Cc...
Subject
Tesla

Hey,

Tesla Supercharger station data has come along nicely. Now please help me out to see what is trending for Tesla on twitter.

Thanks

We would like to search for the hashtag Tesla in Twitter to see what is trending for Tesla. It could be about their car models, new release, supercharger stations, car recalls etc.

Steps below will help us accomplish the goals above,

The Twitter Search tool has to be downloaded from the Alteryx gallery.

To configure Twitter Search, first, we need a Twitter account and have to generate token.

Log in to your Twitter account at https://apps.twitter.com, click "Create a new application", and complete the form (a placeholder website may be used and there is no need for a Callback URL). Once you have submitted the application form, you will be provided with a Consumer Key and Consumer Secret that you can use to configure the tool.

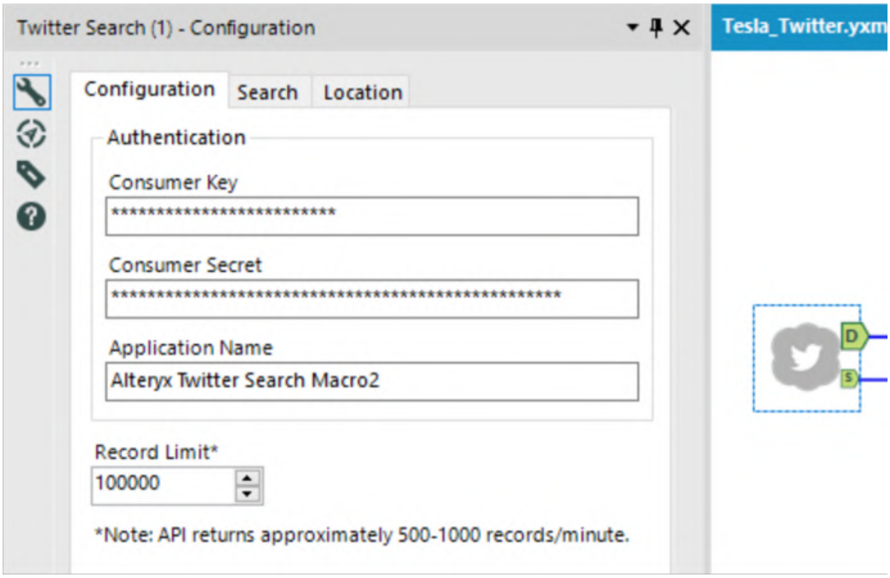


Figure-12-24-Twitter Search Configuration

Enter the hashtag to look for on Twitter. For us, it is "Tesla"

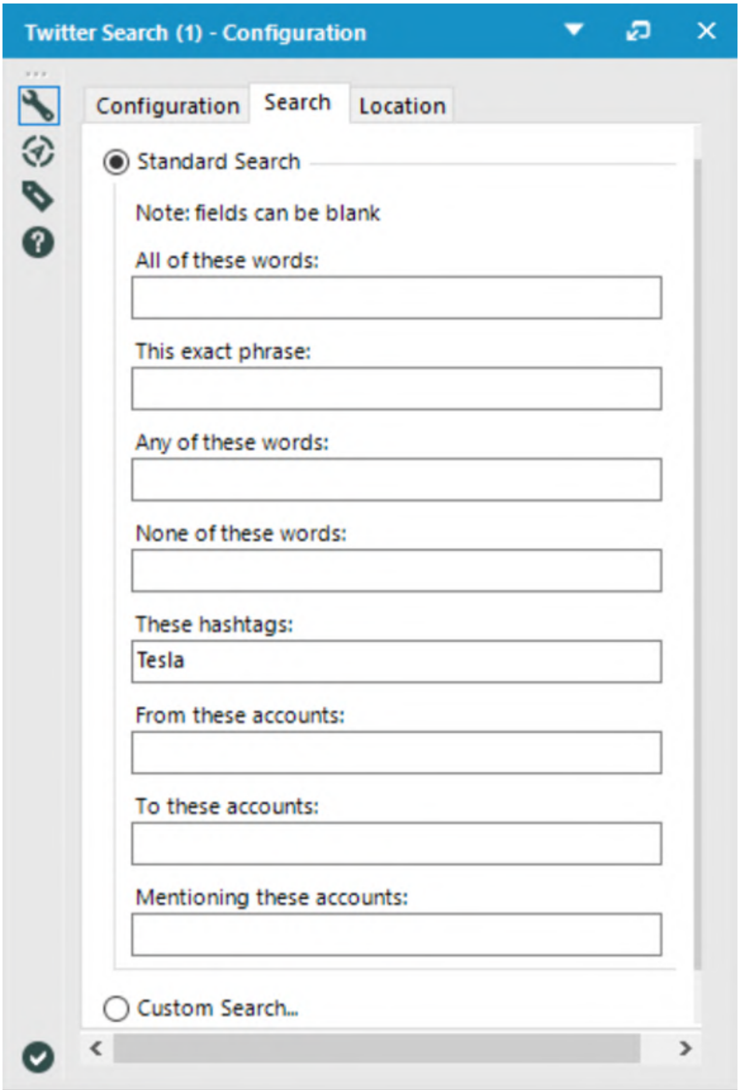


Figure-12-25-Twitter Search Configuration

We get a lot of information from the twitter. To reduce the size of data, we will select only specific fields that we need for analysis. For that we would use Select Tool.

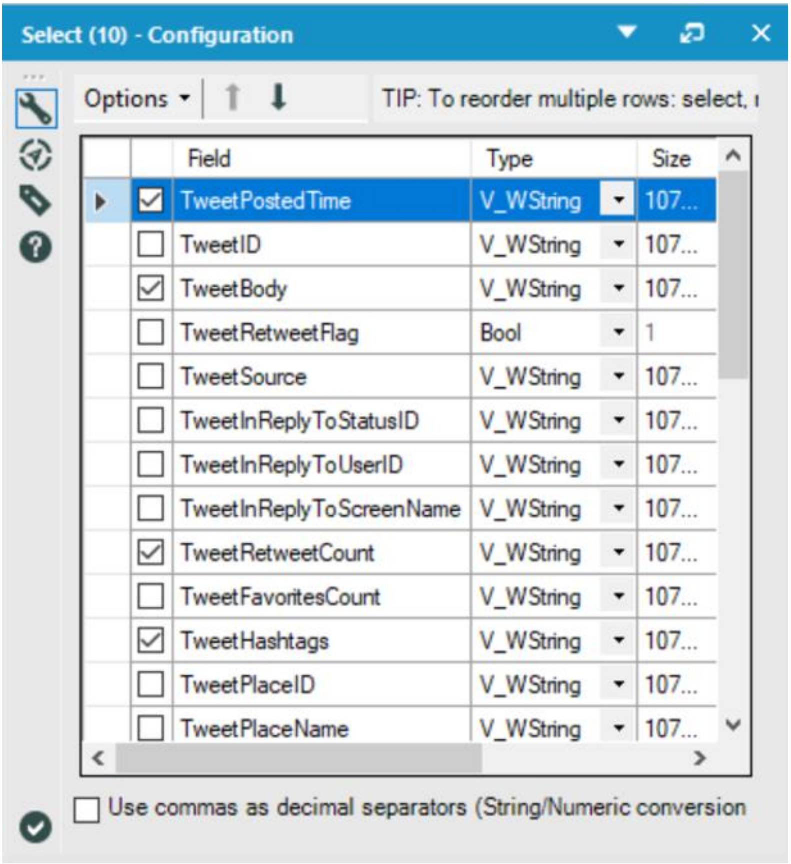


Figure-12-26-Select Configuration

Let us select below data point from the Twitter search output,

TweetPostedTime
TweetBody
TweetRetweetCount
TweetHashtags
Dynamic or Unknown Fields

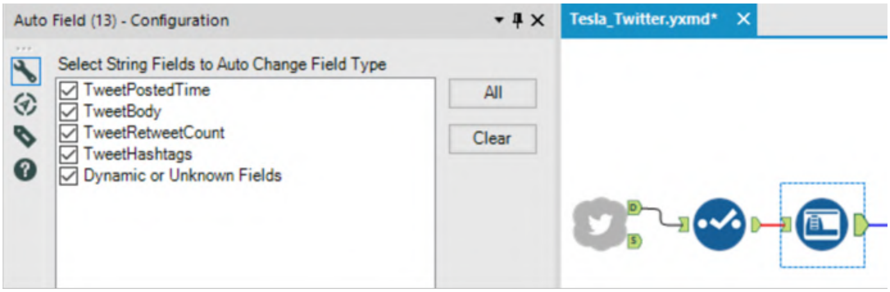


Figure-12-27-Auto Field Configuration

All tweet fields are a string. we use Auto field tool which reads through all the records of an input and sets the field type to the smallest possible size relative to the data contained within the column.

Now we have to find the patterns for these tweets. You can use multiple tools to find the patterns. Like formula tool and Regex tool. In our approach, we will use the Formula tool.

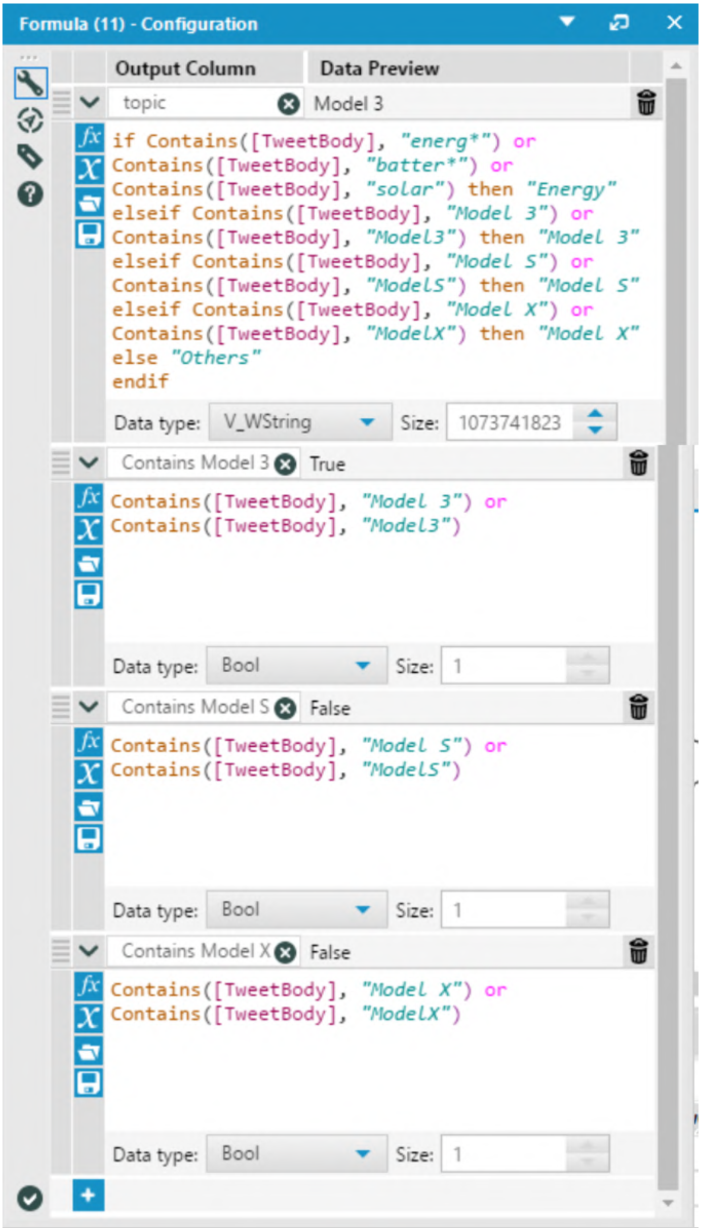


Figure-12-28-Formula Configuration

Now we have a lot of rows with the same patterns and we need to aggregate them using the Summarize tool.

We use the Summarize tool twice.

First, to group by topic and get the number of tweets and retweets.

Second, to calculate a total number of tweets. This number or measure will help us in calculating % of a particular topic in all tweets.

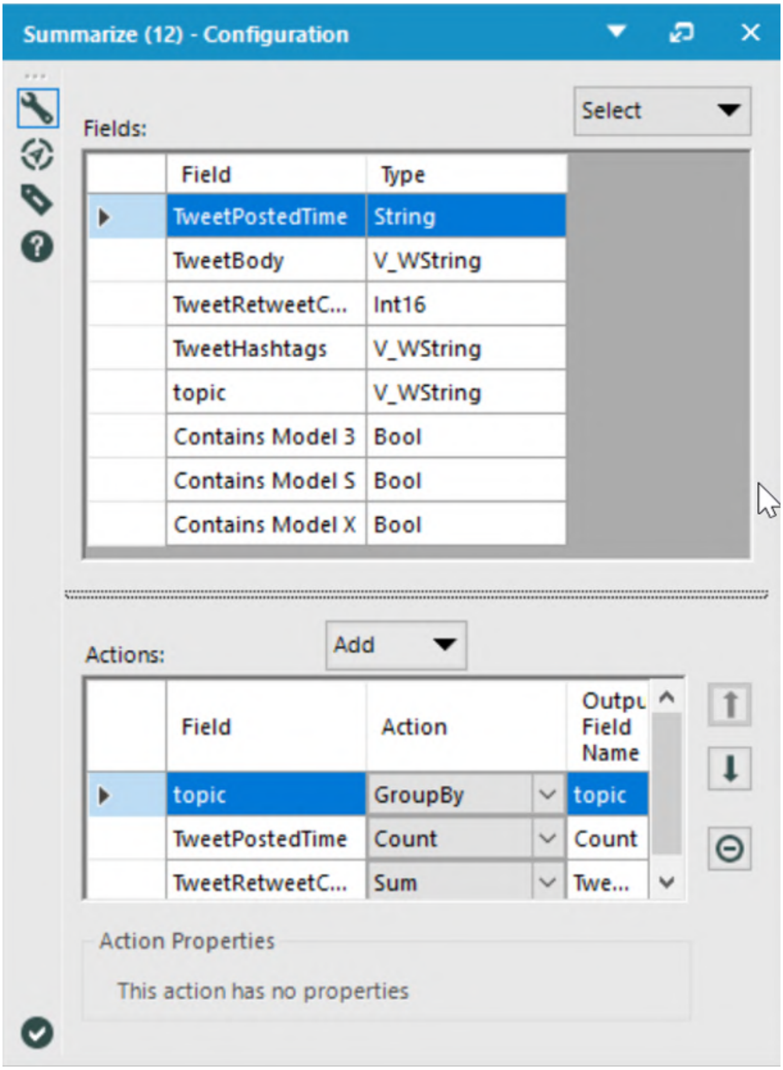


Figure-12-29-Summarize configuration

We will use the append fields tools to append a total number of tweets to each topic group.

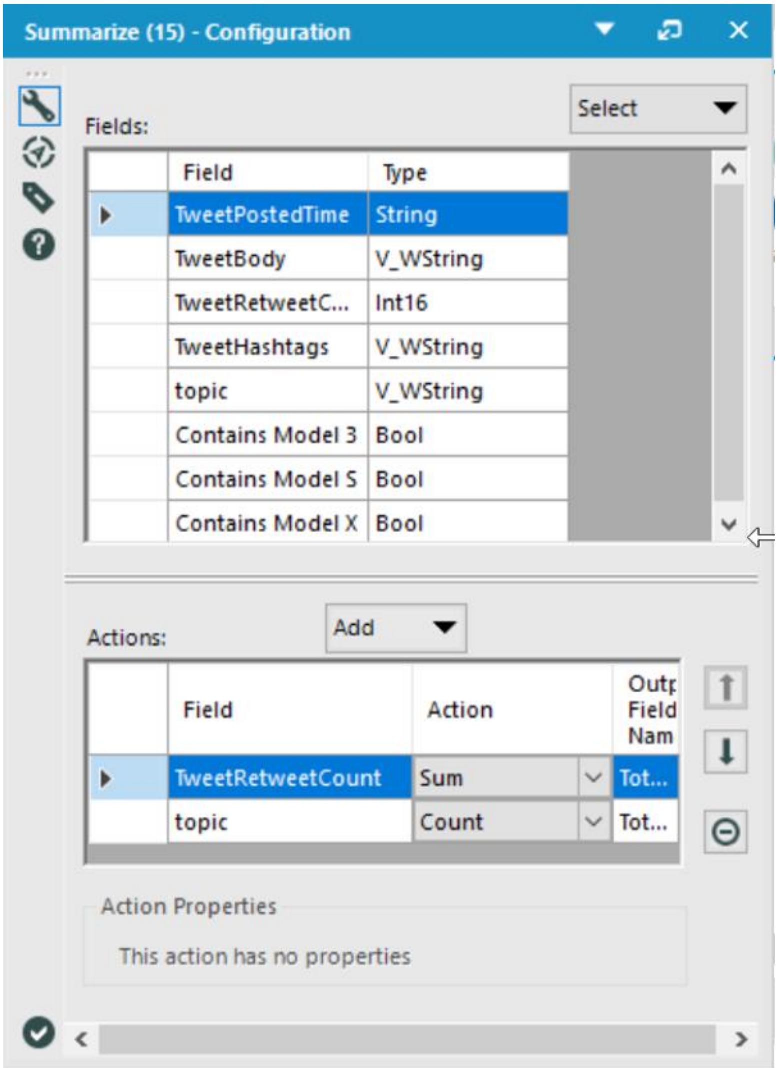
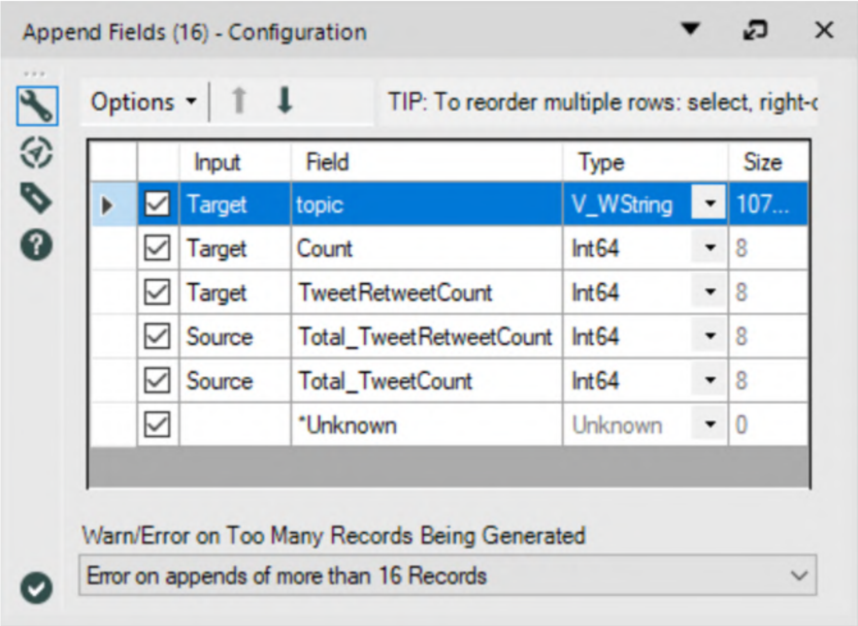


Figure-12-30-Summarize configuration

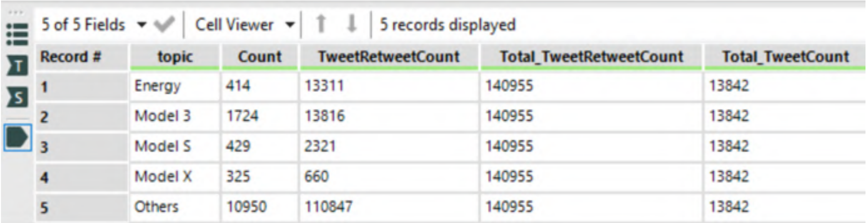
Now we use Append Tool to join and append data from both summarize tools into one.



	Input	Field	Type	Size
☑	Target	topic	V_WString	107...
☑	Target	Count	Int64	8
☑	Target	TweetRetweetCount	Int64	8
☑	Source	Total_TweetRetweetCount	Int64	8
☑	Source	Total_TweetCount	Int64	8
☑		*Unknown	Unknown	0

Figure-12-31-Append Tool Configuration

Now Let's check the result. See if we have all the numbers and fields that we need for analysis.



Record #	topic	Count	TweetRetweetCount	Total_TweetRetweetCount	Total_TweetCount
1	Energy	414	13311	140955	13842
2	Model 3	1724	13816	140955	13842
3	Model S	429	2321	140955	13842
4	Model X	325	660	140955	13842
5	Others	10950	110847	140955	13842

Figure-12-32-Append Tool Result

Let's calculate % of tweets and retweets for each group.

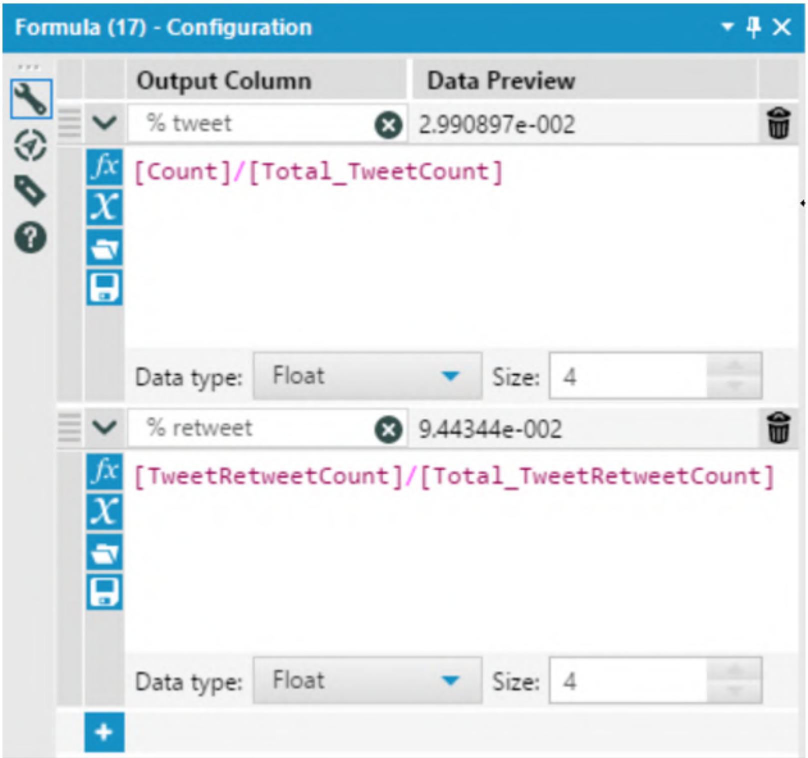
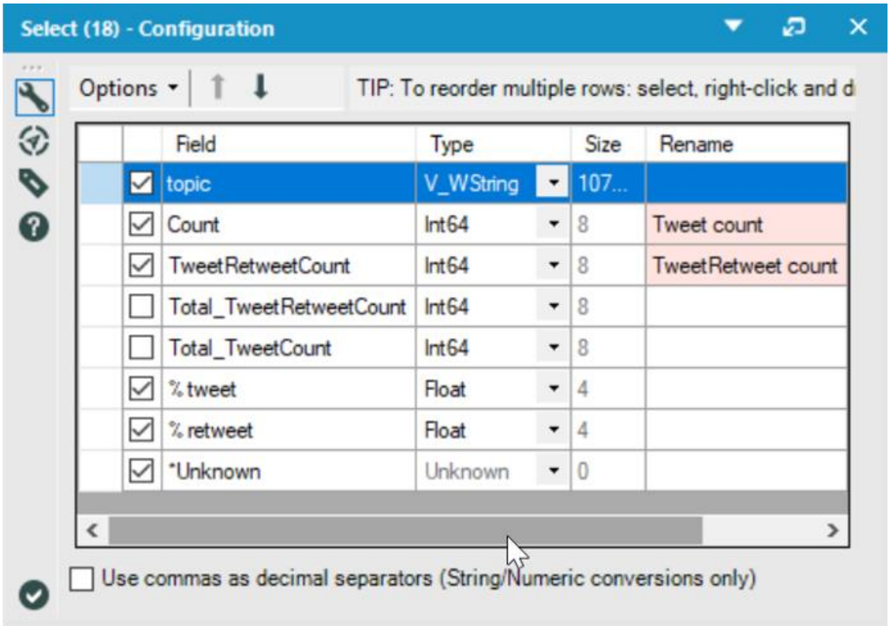


Figure-12-33-Formula Configuration

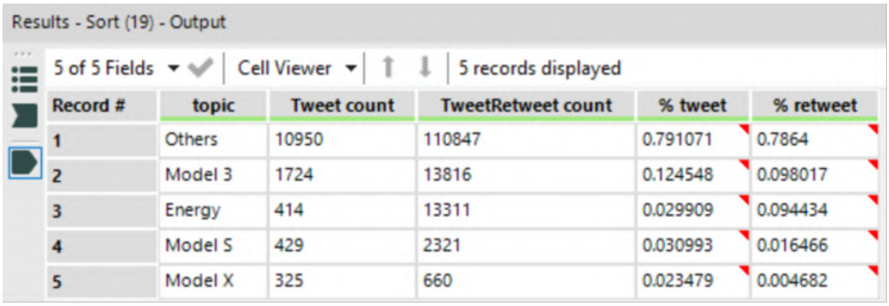
We are almost there. Now let's select all necessary fields and give them appropriate names.



	Field	Type	Size	Rename
☑	topic	V_WString	107...	
☑	Count	Int64	8	Tweet count
☑	TweetRetweetCount	Int64	8	TweetRetweet count
☐	Total_TweetRetweetCount	Int64	8	
☐	Total_TweetCount	Int64	8	
☑	% tweet	Float	4	
☑	% retweet	Float	4	
☑	*Unknown	Unknown	0	

Figure-12-34-Select Configuration

Here is our final output. We have sorted the output based on the % of tweets in descending using Sort tool.



Record #	topic	Tweet count	TweetRetweet count	% tweet	% retweet
1	Others	10950	110847	0.791071	0.7864
2	Model 3	1724	13816	0.124548	0.098017
3	Energy	414	13311	0.029909	0.094434
4	Model S	429	2321	0.030993	0.016466
5	Model X	325	660	0.023479	0.004682

Figure-12-35-Sort Tool Result

The whole workflow is here.

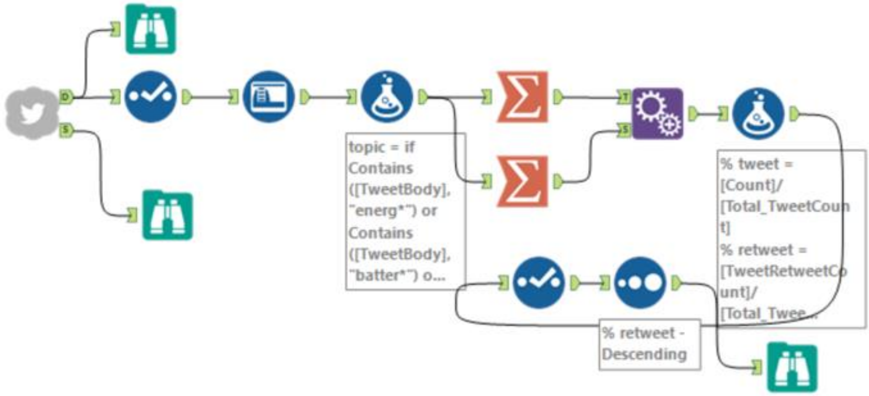


Figure-12-36-What is Trending for Tesla-Data Stream On Completion

CHAPTER 13

Meet the programmer in You

Send

To... Alteryx Consultants

Cc...

Subject Real-Estate Speculators

Hi there!

As an company focused on collating, organizing and presentation of data, we were approached by a large speculator of real-estate supply and demand. This client's super smart business analysts are considering the impact of local temperature on prices.

They are looking for help in gathering and reading temperature data from internet sources. Unfortunately, the data is available in bits and pieces and as html pages or through web API. They realize that the process involves connecting, parsing, sorting and other steps that programmers regularly use.

Can you please help us with this weather information gathering? You can use http://www.wunderground.com

Regards

13.1 Tools & Concepts

13.1.1 Block Until Done

Concept- Sequencing tasks/ Pausing

<table>
<tr><td rowspan="2">
Figure-13-1
Block Until Done</td><td>**Tool Palette: Developer**</td></tr>
<tr><td>Stops data sets from moving downstream until the last record in the set has been processed by all previous tools.
For more details use the link below.
bit.ly/2HslDYf</td></tr>
</table>

13.1.2 Dynamic Input

Concept- SQL and substitution

<table>
<tr><td rowspan="2">
Figure-13-2
Dynamic Input</td><td>**Tool Palette: Developer**</td></tr>
<tr><td>Dynamically chooses what records are read in. Filter criteria, prefix, suffix, table and path can be changed by incoming data.
For more details use the link below.
bit.ly/2JqDtaG</td></tr>
</table>

13.1.3 Dynamic Replace

Concept- Current Date/Time

 **Figure-13-3** **Dynamic Replace**	**Tool Palette: Developer** Quickly replace data values on a series of fields, based on a condition. It is a powerful and visual alternative to REPLACE function available in formula tool. For more details use the link below. bit.ly/2HQi2mb

13.1.4 Dynamic Select

Concept- Field Types and Transpose Tool

 **Figure-13-4** **Dynamic Select**	**Tool Palette: Developer** Like the Select Tool but configurable to pick fields to be selected (or de-selected) either by field type or via a formula. For more details use the link below. bit.ly/2HQS8Pm

13.1.5 Field Info

Concept-Data types and metadata programming

<table>
<tr><td rowspan="2">
Figure-13-5
Field Info</td><td>**Tool Palette: Developer**</td></tr>
<tr><td>The Field Info tool outputs the description of the data. This tool can be used for validation of schema before scheduling processing.
For more details use the link below.
bit.ly/2JslDnO</td></tr>
</table>

13.1.6 Run Command

Concept- Windows batch files

<table>
<tr><td rowspan="2">

Figure-13-6
Run Command</td><td>**Tool Palette: Developer**</td></tr>
<tr><td>Run command tool is used to run external programs as part of an Alteryx process.
For more details use the link below.
bit.ly/2qYLH2X</td></tr>
</table>

13.1.7 Test

Concept- Regression Testing

 **Figure-13-7 Test**	**Tool Palette: Developer**
	The Test tool is useful for testing assumptions about the data. It is a good practice to verify data before an update is committed to the database. For more details use the link below. bit.ly/2FidmQL

13.1.8 Throttle

Concept- Cost of using 3rd party paid webservices

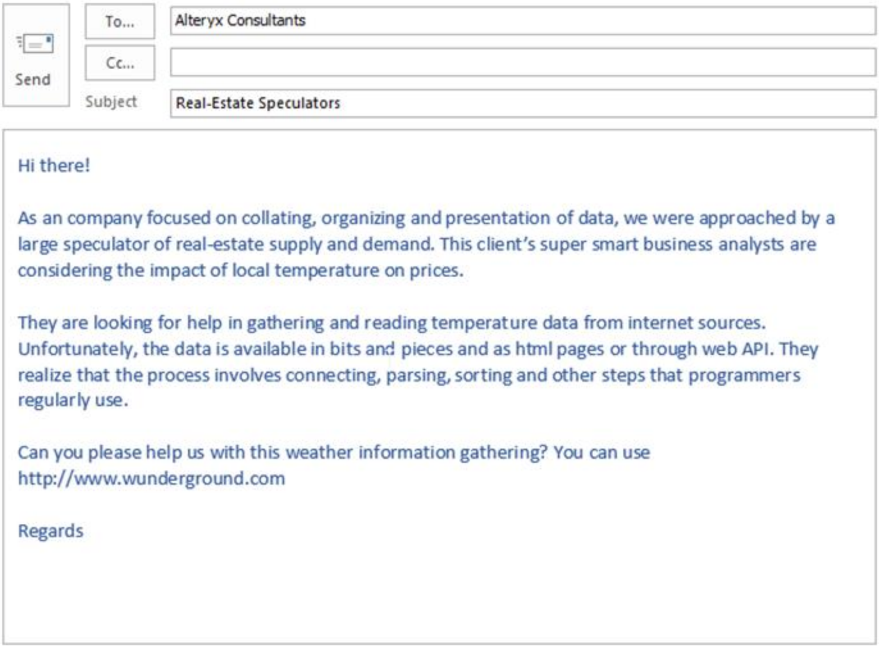
 **Figure-13-8 Throttle**	**Tool Palette: Developer**
	Slows down the speed of downstream processing by limiting the number of records that are passed through it. For more details use the link below. bit.ly/2r0hMXC

13.1.9 JSON Build

Concept- Objects and key-value pairs

<table>
<tr><td rowspan="2">
Figure-13-9
JSON Build</td><td>**Tool Palette: Developer**</td></tr>
<tr><td>The JSON Build tool allows us to read table schema and build it into Java Script Object Notation.
For more details use the link below.
bit.ly/2FiPSef</td></tr>
</table>

13.2 Jeeves! What was the weather like at Chicago last year?


Send
To... Alteryx Consultants
Cc...
Subject Real-Estate Speculators

Hi there!

As an company focused on collating, organizing and presentation of data, we were approached by a large speculator of real-estate supply and demand. This client's super smart business analysts are considering the impact of local temperature on prices.

They are looking for help in gathering and reading temperature data from internet sources. Unfortunately, the data is available in bits and pieces and as html pages or through web API. They realize that the process involves connecting, parsing, sorting and other steps that programmers regularly use.

Can you please help us with this weather information gathering? You can use http://www.wunderground.com

Regards

Here we are putting on our Developer hats and proceeding but don't let that moniker scare you. These steps are simple logical extensions to some of the sophisticated tools you saw in the past. You will see as we start building a workflow (*Weather Reader)* that will help fulfil the client's request. You may have to sign up for an API key from your web data provider. We just obtained our free key from https://www.wunderground.com/weather/api/d/pricing.html.

On this page, click on Purchase Key and then Sign up for free. They give you an encryption API key which is displayed as Key ID of the Key Settings page to make sure your workflows are authorized. The API Table of Contents has link of Data Features and we choose history. This page instructs us to setup the URL as http://api.wunderground.com/api/API_Key/history_YYYYMMDD /q/CA/San_Francisco.json. Increasingly, sources set this up to ensure that they provide an appropriate Quality of Service. Paid subscribers get more attention if the service has a disruption.

Figure-13-10-URL Input to get weather

We are going to start with a blank canvas and save it as *Weather Reader*. Next, bring in the *Text Input* tool; and setup the URL to pull data. We see that the URL takes in location, year, month, date and API key hence set those up as fields in the text input tool too. In the Text Input tool, as you Insert a new column a fieldname is generated such as Field1, Field2, etc. Rename these fieldnames and setup as shown in Figure-13-10. The values can change as per your convenience. You can even enter in multiple rows of data here, if you wish that the flow executes for multiple dates.

Alteryx will try to predetermine the type and set fields like year to number type but we want all these fields to be string as we will send the whole set to our website as a single string. Use the *Select* tool, and make sure the datatypes are set to string.

Our next step is make sure that Alteryx can replace the date based on values we provided to the text input tool. Hence run a replace step for each of Location, API key, Year, Month and Date.

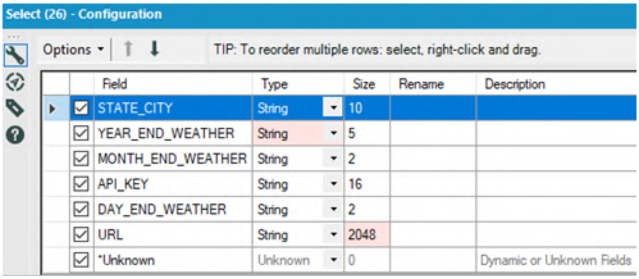


Figure-13-11-Changing datatypes

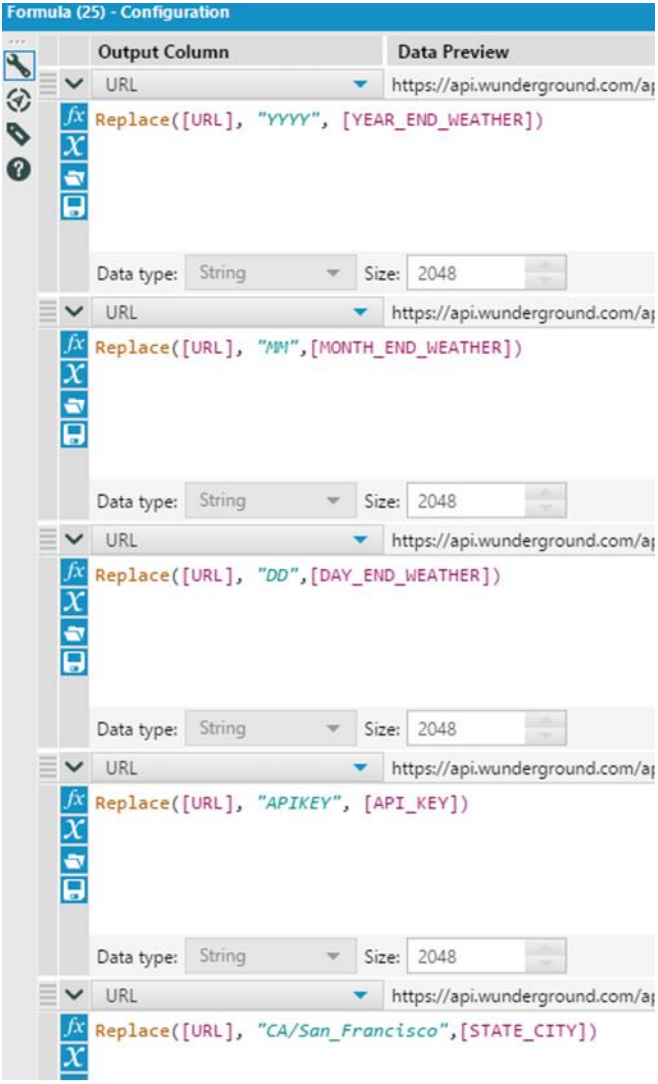


Figure-13-12-Formula Configuration

Now that we have the URL in the correct format, the next step is to import data from the URL. To achieve this, we should use *Download* tool.

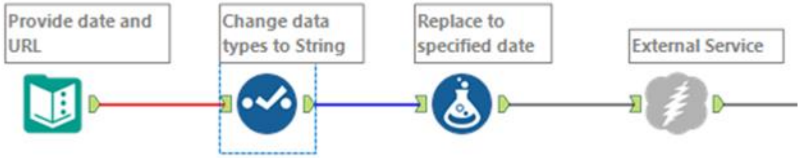


Figure-13-13-Download tool

Best practice is to remove data as soon as it is no longer needed. It makes sense that the first step in filtering would be to bring in the Filter tool; however, if we are not familiar with the data set and we have not run it, we may not have enough information to filter properly. In this case, we want to run the module so that there's data in the Browse section for us to work with.

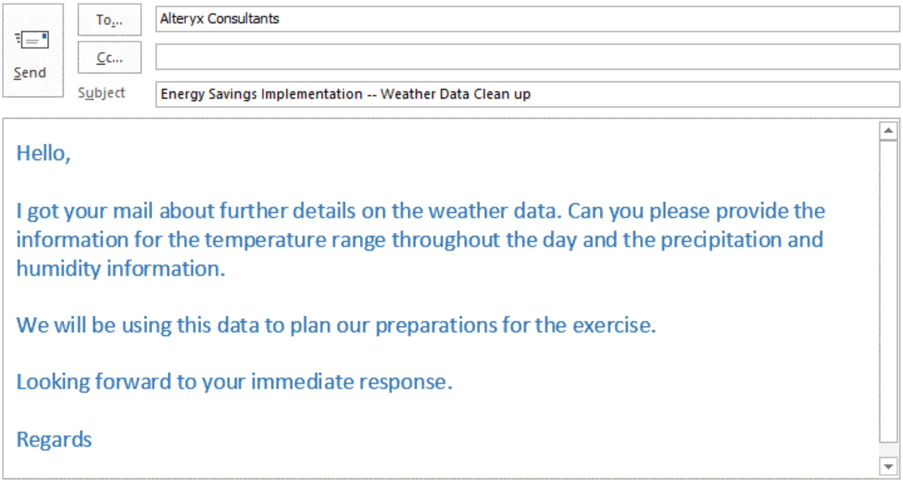
Record #	Y	M	D	URL	DownloadData
1	"	"	"	http://www.wunderground.com/history/airport...	

Figure-13-14-Download tool output readability

We can see that when the module finishes running, we get a pop-up window that lets us know how long it took to run and if there were errors. On browsing the outputs as shown in Figure-13-14 you see a blank field *DownloadData*. Right click and choose *Copy Selected Cells Without Headers* and past into notepad. You will notice data in Key: Value format known as JSON that looks like this:


```
{
  "response": {
  "version":"0.1",
  "termsofService":"http://www.wundergroun
  "features": {
  "history": 1
  }
        }
        ,
        "history": {
                "date": {
                "pretty": "June 1, 2017",
                "year": "2017",
```


Figure-13-15-JSON DownloadData



Send

To... Alteryx Consultants

Cc...

Subject Energy Savings Implementation -- Weather Data Clean up

Hello,

I got your mail about further details on the weather data. Can you please provide the information for the temperature range throughout the day and the precipitation and humidity information.

We will be using this data to plan our preparations for the exercise.

Looking forward to your immediate response.

Regards

We have weather data, but unfortunately, it's not in usable form. We are going to clean the data and employ best practices. So, the process that we are going to follow is:

1. Setup URL as part of Text Input
2. Import data from URL via Download
3. Split the data into Rows and Columns
4. Remove redundant columns via the select tool.
5. Remove data that is not relevant to Temperature
6. Transpose data to switch key and value from separate columns into column headers and rows
7. Write the data for each date as a file.

We have already completed few of the steps listed above. We would be continuing from step 3. The *Select* tool helps to remove all the unnecessary data which were downloaded from the webservice.

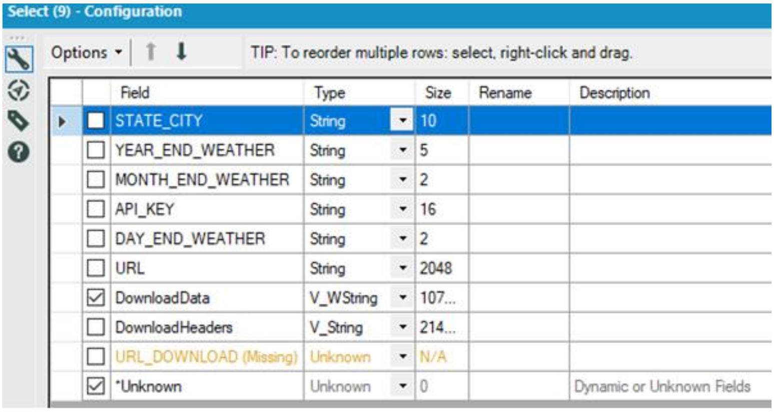


Figure-13-16-Select Only Downloaded Data

The *JSON Parse* tool takes text data and splits it into the hierarchy of fields, arrays and values as shown in Figure-13-18. All values are collated neatly as JSON_ValueString and the fields fall into JSON_Name. If there is a hierarchy, the JSON_Name will follow the Dot Notation and indicate which field is a child of the other as Parent. Child. Grandchild

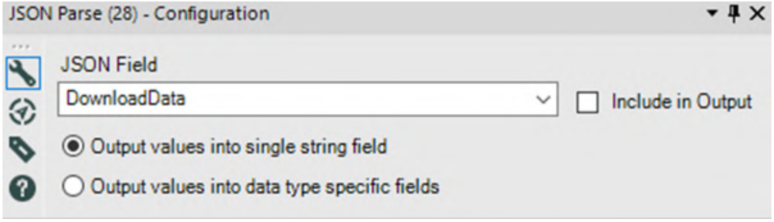


Figure-13-17-JSON Parse of Downloaded data

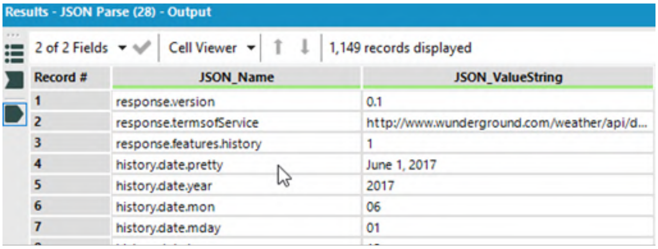


Figure-13-18-JSON Parse Output

We then split the field names based on the delimiter "." into 4 columns using the *Text to columns* tool. We chose 4 levels as we searched for temperature data in the downloaded text that we copied into notepad. A little trial and error helps too.

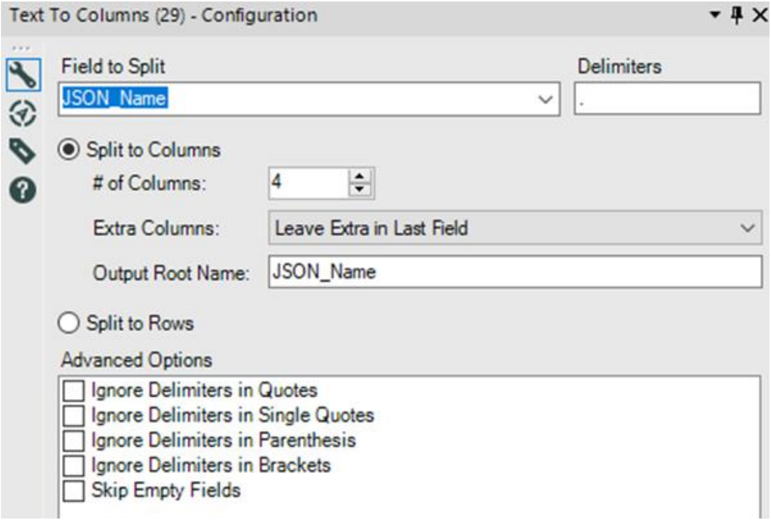


Figure-13-19-Split the field names into parts

We will now include a Filter for temperature observations of the day and ignore the summary data. The formula relies on the field names to contain the letters "temp" and "observations". For good measure we want to keep the hour and min at which the recording was noted. Hence, we add the *date.hour* and *date.min* also into our filter.

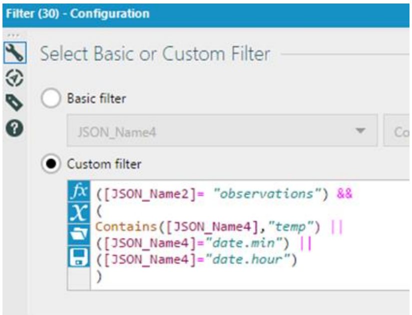


Figure-13-20-Filter down to temperature summary

We then transpose the rows and columns to move the JSON key names into field names. See the Figure-13-24 for the output format

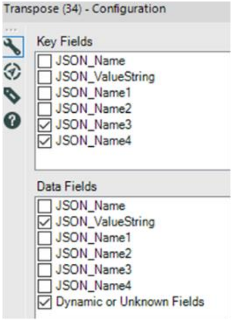


Figure-13-21-Transpose of JSON data

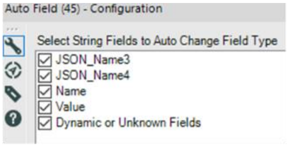


Figure-13-22-Changing datatypes to reduce size

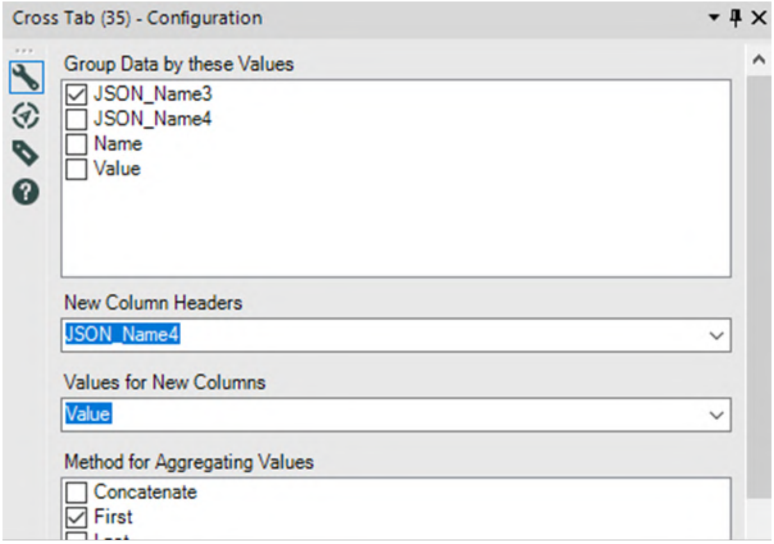
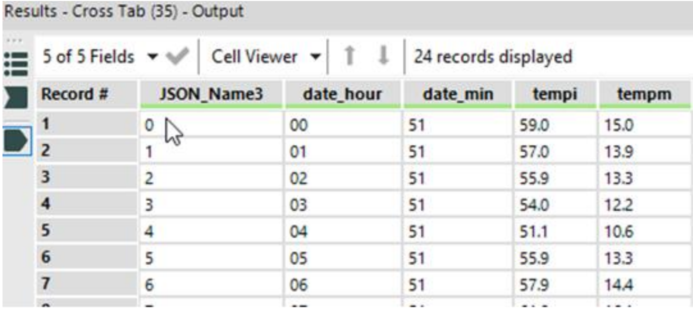


Figure-13-23-Create a crosstab which bubbles up the JSON_Name4 value as field name



Record #	JSON_Name3	date_hour	date_min	tempi	tempm
1	0	00	51	59.0	15.0
2	1	01	51	57.0	13.9
3	2	02	51	55.9	13.3
4	3	03	51	54.0	12.2
5	4	04	51	51.1	10.6
6	5	05	51	55.9	13.3
7	6	06	51	57.9	14.4

Figure-13-24-JSON names switched to Field names

This data looks good but how about using field names that are more end-user friendly like saying tempi is actually temperature in Imperial units. Here we add a Dynamic Rename tool that accepts the field list as an input from the right stream and replaces the names of the fields on the left stream. Note the choice of the Rename Mode on the configuration.

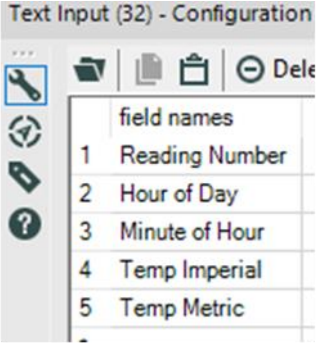


Figure-13-25-Custom Names

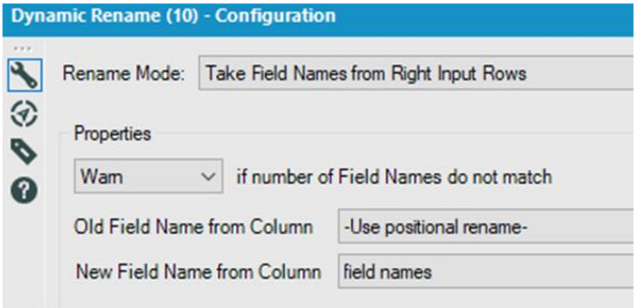


Figure-13-26-Dynamic Rename to Friendly Fieldnames

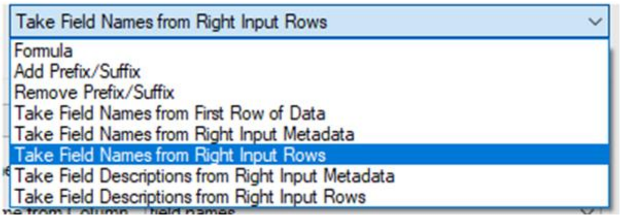


Figure-13-27-Dynamic Rename Mode

Before we use the data stream any further we should make sure that all rows of data were captured; most days it is 24 and more on other days. Hence, we add a *Block Until Done* tool and also add a *Test* tool that ensures that the record count is exact.

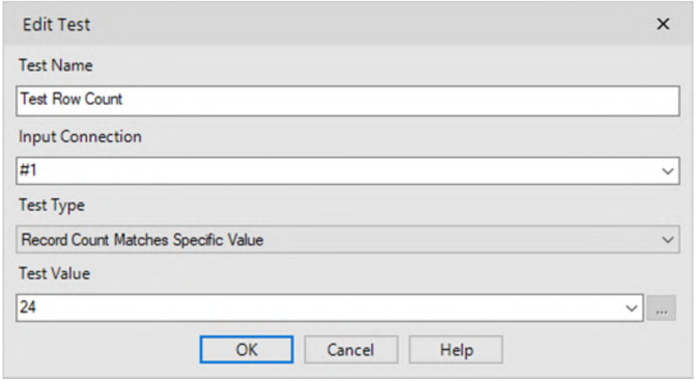


Figure-13-28-Test for number of readings

All this recording is great, but it would be great if we can map the date we specify at the beginning to be a part of this filename to store. To do that we add an Append Field tool which adds the date but don't worry the date will not show up on each record. It will only be a part of the filename.

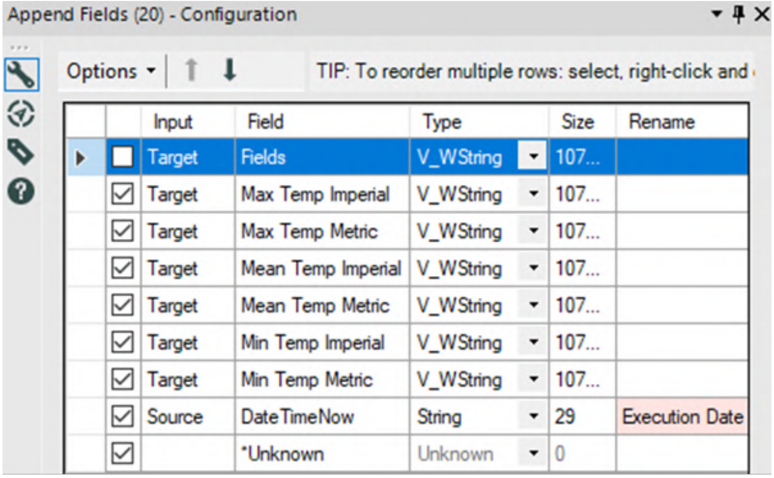


Figure-13-29-Append Field tool to add Execution Date

We can use the Field Info tool to gather the lineage of the data that can be stored as Metadata for future. The actual data is redirected to an output which takes the date suffix field and adds to the filename. We create a single table for each date to use later. Notice that the date itself is not part of the data table saved on the filesystem.

If we run the module, we see what the final dataset looks like. This ensures the information written into the. yxdb was correct.

Alteryx Designer x64 - Tempurature_Chicago_20170401.yxdb

24 records displayed, 5 fields, , 1776 bytes

Record #	Reading Number	Hour of Day	Minute of Hour	Temp Imperial	Temp Metric
1	0	00	51	37.0	2.8
2	1	01	51	37.0	2.8
3	2	02	51	36.0	2.2
4	3	03	51	36.0	2.2
5	4	04	51	37.0	2.8
6	5	05	51	34.0	1.1
7	6	06	51	35.1	1.7
8	7	07	51	39.0	3.9
9	8	08	51	44.1	6.7
10	9	09	51	48.0	8.9
11	10	10	51	52.0	11.1
12	11	11	51	54.0	12.2
13	12	12	51	55.0	12.8
14	13	13	51	57.9	14.4
15	14	14	51	57.9	14.4
16	15	15	51	57.9	14.4
17	16	16	51	57.0	13.9
18	17	17	51	54.0	12.2
19	18	18	51	52.0	11.1
20	19	19	51	46.9	8.3
21	20	20	51	48.0	8.9
22	21	21	51	45.0	7.2
23	22	22	51	45.0	7.2
24	23	23	51	44.1	6.7

Figure-13-30-Contents of Output YXDB

Figure-13-32 shows how the *Weather Reader* data stream looks on completion. We added in a *Throttle* tool Figure-13-31 to ensure underground does not receive more than 60 requests per minute in the event that we setup text input to capture hundreds of days of temperature data.

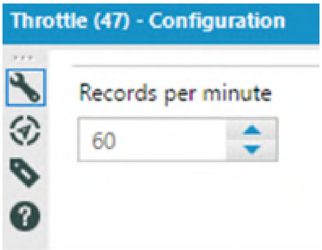


Figure-13-31-Throttle

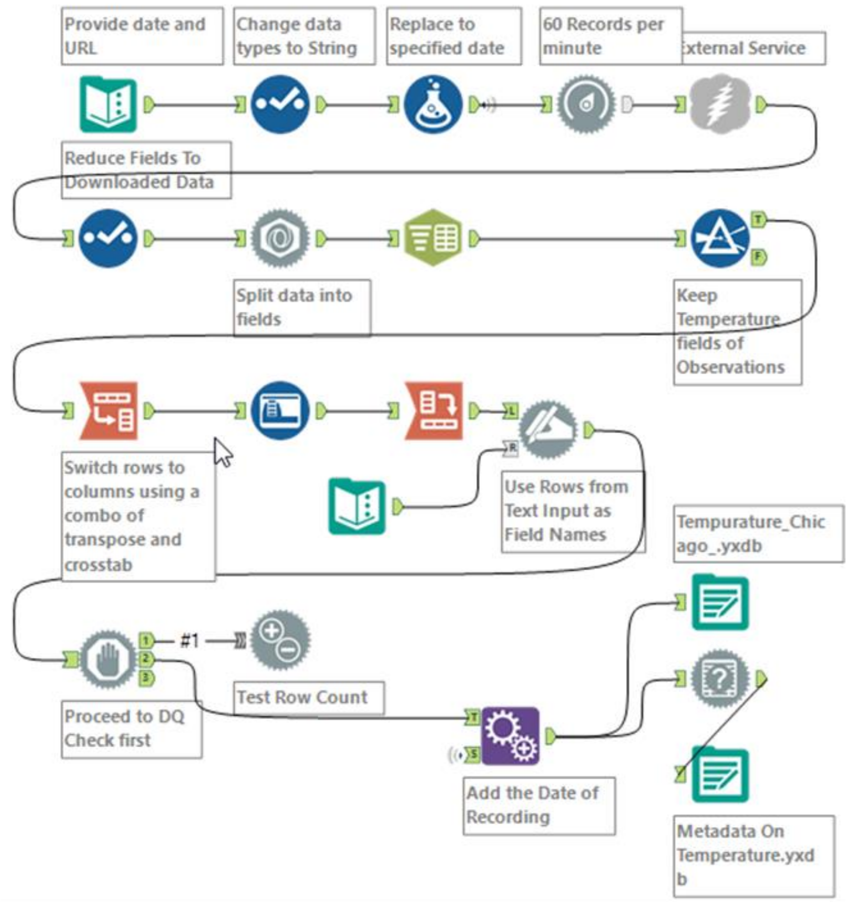
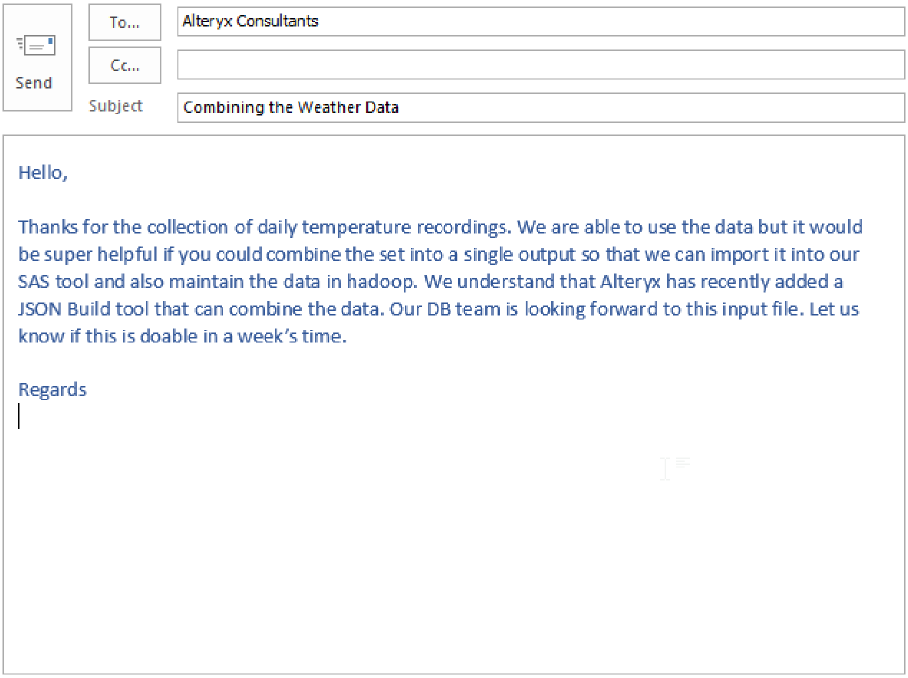


Figure-13-32-Weather Reader

13.3 Fieldnames to JSON tree



Send

To... Alteryx Consultants

Cc...

Subject Combining the Weather Data

Hello,

Thanks for the collection of daily temperature recordings. We are able to use the data but it would be super helpful if you could combine the set into a single output so that we can import it into our SAS tool and also maintain the data in hadoop. We understand that Alteryx has recently added a JSON Build tool that can combine the data. Our DB team is looking forward to this input file. Let us know if this is doable in a week's time.

Regards

This is in itself is an innocuous request and you are pretty sure it does not require the entire week to setup this flow. We start by deciding the specific dates for which we plan to merge the data. The files are named in the format: -

Temperature_Chicago_20170401.yxdb,
Temperature_Chicago_20170501.yxdb,
Temperature_Chicago_20170601.yxdb.

These files are available for readers under *Chapter 13 – ChicagoWeatherInput*. The content of all the files follow the same template. For starters we take 3 dates and setup a *Text Input* tool as shown in Figure-13-33 by renaming Field1 as *ReadDate* and filling in the 3 dates.

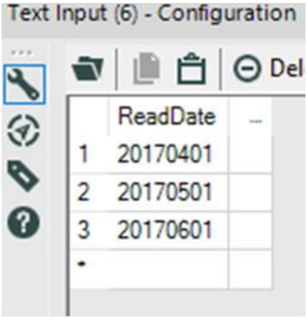


Figure-13-33-Dates to read temperature data

This list can be used in a Dynamic Input tool that can read multiple files and combine the contents of multiple files based on the list of names provided. The fun part is that we don't even have to specify each filename explicitly. If the filenames follow a regular pattern, this tool will generate the suffix dynamically based on the text input or a SQL query. We use the list from the text input to generate the suffixes hence it is necessary to use the *Append Suffix to File/Table Name option.*

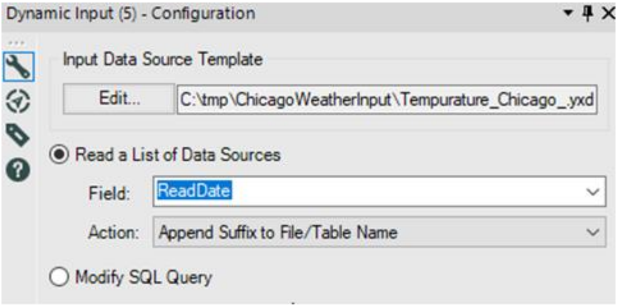


Figure-13-34-Dynamic Input to change suffix

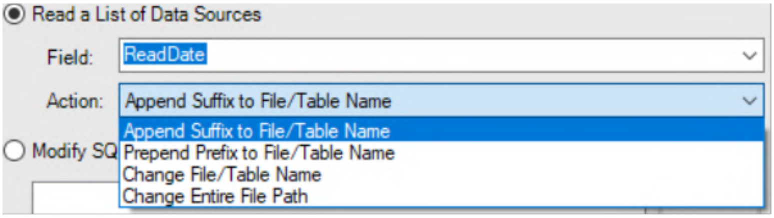


Figure-13-35-Dynamic Input Options

The Dynamic Select Tool as shown on Figure-13-36 allows specifying which fields we would allow downstream based on the type of the field or any number of other formula instead of explicitly selecting fields which an ordinary Select tool from Preparation palette provides. Here we use part of the name and drop the Metric related fields as Imperial measures suffice our study. A quick Autofield passthrough reduces the footprint by optimizing the data types based on the data coming in. Figure-13-37 shows the Autofield configuration.

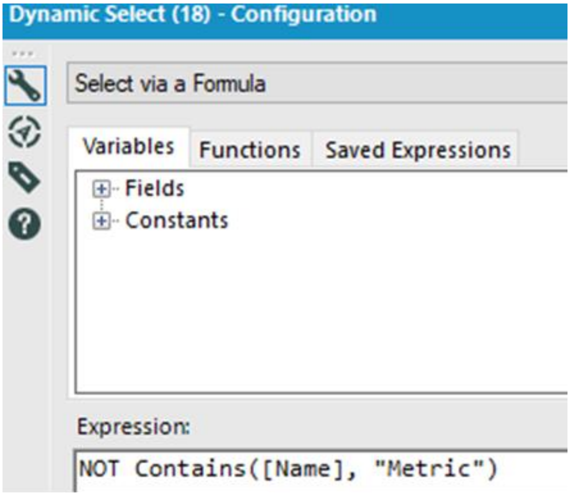


Figure-13-36-Dynamic Select

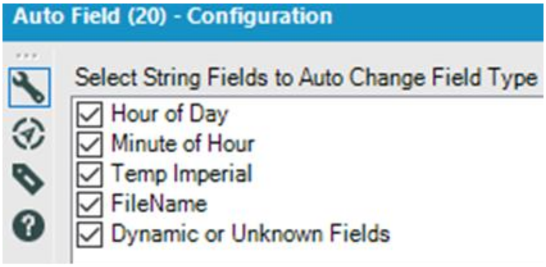


Figure-13-37-AutoField tool helps reduce space in the downstream tools

We then sort the data and send the data to a *Run Command* tool that executes *keepcopy.bat* which is available to readers under *Chapter 13-*

ChicagoWeatherInput. We use this batch file as a simple sample but it can be expanded for complicated requirements such as sending emails through SendMail or other proprietary commands available for you as a Windows shell programmer. This tool expands Alteryx horizons beyond the realm of simple data manipulation and has been used to develop entire workforce process management. In the simplified batch file, we create a copy of the incoming files in order to preserve history of which files have already been processed. This is one of the best practices of ETL industry.

Please note that the Run Command is tricky for 2 reasons.

1) The location of the batch/command file itself. If we try using relative path then the path should be relative the location of Alteryx under C:\Program Files. Hence it is a common practice to provide absolute path for this tool.

2) The exit status within the script. Make sure that command file exits with status 0. We typically add "exit 0" to the last line of the batch file.

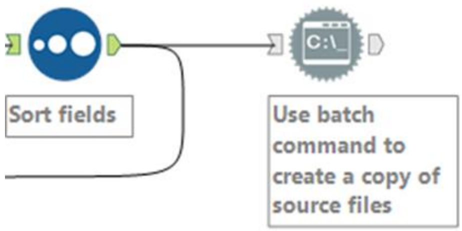


Figure-13-38-Sort tool and Run Command tool

The client did specify that they needed a JSON output and they were well informed that *JSON Build* is now part of Alteryx. Hence, we shall honor their request and use just that to speed up our work. In the past, many workflow designers have created macros for this but now we have a tool on the Developer palette. Figure-13-39 shows the configuration for JSON Build.

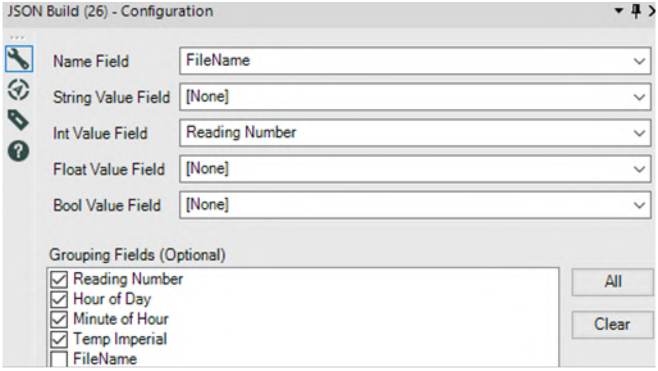


Figure-13-39-JSON Build Configured for Reading Number

The completed flow is depicted in picture 13-40 that reads multiple files, combines all the data and sends the notification as specified by the client.

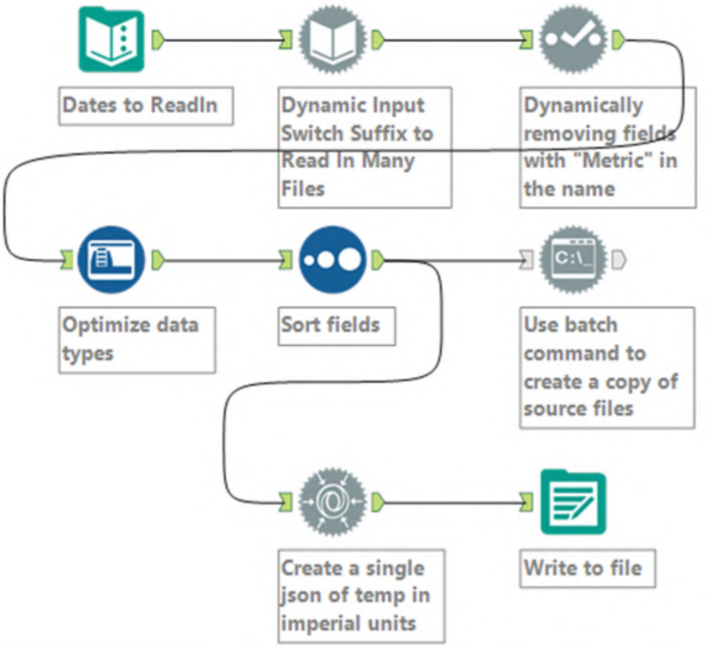


Figure-13-40-Temperature Files Reader

13.4 How's the oil market doing?

Send

To... Alteryx Consultants

Cc...

Subject How's the Oil Market Doing

Hey,

Thanks for your analysis on the Weather Data .

Here is some data from our sources which we would use for Oil prices information . However we are not able to understand anything as this is JSON format.

Can you please put it in a reasonable format for us to use it for our analysis ?

Thanks

Chapter 14

Statistics in Alteryx

14.1 Tools & Concepts

14.1.1 Basic Data Profile

Concept- Metadata Insights

 **Figure 14-1-Basic Data Profile**	**Tool Palette: Data Investigation**
	Outputs basic metadata such as data type, min, max, average, number of missing values, etc. For more details use the link below. bit.ly/2K3lxUW

14.1.2 Field Summary

Concept- Data Insights

 **Figure 14-2-Field Summary**	**Tool Palette: Data Investigation**
	Analyzes data and creates a summary report containing descriptive statistics of data in selected columns. For more details use the link below. bit.ly/2HCB3Zn

14.1.3 Frequency Table

Concept- Descriptive Analysis

<table>
<tr><td rowspan="2">
Figure 14-3-Frequency Table</td><td>Tool Palette: Data Investigation</td></tr>
<tr><td>Produce a frequency analysis for selected fields - output includes a summary of the selected field(s) with frequency counts and percentages for each value in a field.
For more details use the link below.
bit.ly/2Jcze2s</td></tr>
</table>

14.1.4 Pearson Correlation

Concept- Variables Correlation and Interdependencies

<table>
<tr><td rowspan="2">
Figure 14-4-Pearson Correlation</td><td>Tool Palette: Data Investigation</td></tr>
<tr><td>Measures the linear dependence between two variables as well as the covariance.
For more details use the link below.
bit.ly/2qNIClv</td></tr>
</table>

14.1.5 Spearman Correlation

Concept- Variables Correlation and Interdependencies

 **Figure 14-5-Spearman Correlation**	**Tool Palette: Data Investigation**
	Assesses how well an arbitrary monotonic function could describe the relationship between two variables, without making any other assumptions about the particular nature of the relationship between the variables For more details use the link below. bit.ly/2HfTVxY

14.1.6 Decision Tree

Concept- Prediction using Statistical Methods

 **Figure 14-5-Decision Tree**	**Tool Palette: Predictive**
	It is a class of Statistical Methods that predict a target variable using one or more variables that are expected to have an influence on the target variable and are often called Predictor Variables. For more details use the link below. bit.ly/2HMDs1b

14.1.7 Forest Model

Concept- Prediction using Machine Learning Methods

<table>
<tr><td rowspan="2">
Figure 14-7-Forest Model</td><td>Tool Palette: Predictive</td></tr>
<tr><td>It is a class of Machine Learning Methods that predict a target variable using one or more variables that are expected to have an influence on the target variable and are often called predictor variables.
For more details use the link below.
bit.ly/2qOO07T</td></tr>
</table>

14.1.8 Score

Concept- Prediction using R models

<table>
<tr><td rowspan="2">
Figure 14-8-Score</td><td>Tool Palette: Predictive</td></tr>
<tr><td>It takes as inputs an R model object produced by the Logistic Regression, Decision Tree, Forest Model, or Linear Regression macro and a data stream that is consistent with the model object (in terms of field names and the field types) and outputs the data stream with a one (for a model with a continuous target) or two or more (for a model with a categorical target) "Score" (fitted value) field(s) appended to the data stream.
For more details use the link below.</td></tr>
</table>

	bit.ly/2K3muwu

14.1.9 Support Vector Machine

Concept- Prediction using Supervised learning algorithms

 Figure 14-9-Support Vector Machine	**Tool Palette: Predictive**
	Supervised learning algorithms used for classification problems, and are meant to accommodate instances where the data (i.e., observations) are considered linearly non-separable, For more details use the link below. bit.ly/2K49Soz

14.1.10 ARIMA

Concept- Forecasting time series field

<table>
<tr><td rowspan="2">
Figure-14-10 -ARIMA</td><td>Tool Palette: Time Series</td></tr>
<tr><td>Estimates a time series forecasting model, either as a univariate model or one with covariates (predictors), using an autoregressive integrated moving average (or ARIMA) method.
For more details use the link below.
bit.ly/2vtTxqe</td></tr>
</table>

14.1.11 ETS

Concept- Univariate time series forecasting model

<table>
<tr><td rowspan="2">
Figure-14-11 - ETS</td><td>Tool Palette: Time Series</td></tr>
<tr><td>The ETS tool estimates a univariate time series forecasting model using an exponential smoothing method.
For more details use the link below.
bit.ly/2qTHyNQ</td></tr>
</table>

14.1.12 TS Compare

Concept- Compares one or more, time series models

 **Figure-14-12 – TS Compare**	**Tool Palette: Time Series**
	The TS Compare tool analyzes one or more models made with either the ETS or ARIMA macros, including ARIMA models that use covariates. For more details use the link below. bit.ly/2qKWJsx

14.1.13 TS Forecast

Concept- Provides forecasts from either an ARIMA or ETS

 **Figure 14-13 – TS Forecast**	**Tool Palette: Time Series**
	The TS Forecast tool provides forecasts from either an ARIMA or ETS model for a user specified number of future periods For more details use the link below. bit.ly/2HeCEW1

14.1.14 Linear Regression

Concept- Predicts a target variable

 **Figure-14-14 – Linear Regression**	**Tool Palette: Predictive**
	The Linear Regression tool constructs a linear function to create a model that predicts a target variable based on one or more predictor variables. For more details use the link below. bit.ly/2vFMJ8O

14.1.15 Logistic Regression

Concept- Obtains estimated probability for possible responses.

 **Figure-14-15 – Logistic Regression**	**Tool Palette: Predictive**
	The Logistic Regression tool creates a model that relates a target binary variable (such as yes/no, pass/fail) to one or more predictor variables to obtain the estimated probability for each of two possible responses for the target variable For more details use the link below. bit.ly/2vuLGsb

14.1.16 Naïve Bayes Classifier

Concept- Uses binomial or multinomial probabilistic classification model for making predictions.

	Tool Palette: Predictive

 **Figure-14-16 - Naïve Bayes Classifier**	The Naive Bayes Classifier tool creates a binomial or multinomial probabilistic classification model of the relationship between a set of predictor variables and a categorical target variable. For more details use the link below. bit.ly/2HxWHhh

14.1.17 Stepwise

Concept- Determines the best predictor variables to include in a model

 **Figure-14-17 -Stepwise**	**Tool Palette: Predictive**
	The Stepwise tool determines the best predictor variables to include in a model out of a larger set of potential predictor variables for linear, logistic, and other traditional regression models. For more details use the link below. bit.ly/2HwT778

14.1.18 Append Cluster

Concept- Used to create the original cluster solution

<table>
<tr><td rowspan="2">
Figure-14-18 – Append Cluster</td><td>**Tool Palette: Predictive Grouping**</td></tr>
<tr><td>The Append Cluster tool appends the cluster assignments from a K-Centroids Cluster Analysis tool to a data stream. The data stream does not need to be the same one that the cluster solution is based on.
For more details use the link below.
bit.ly/2K2272A</td></tr>
</table>

14.1.19 K-Centroids Cluster Analysis

Concept- Represents a class of algorithms for partitioning cluster analysis

<table>
<tr><td rowspan="2">
Figure-14-19 - K-Centroids Cluster Analysis</td><td>**Tool Palette: Predictive Grouping**</td></tr>
<tr><td>K-Centroids represent a class of algorithms for doing what is known as partitioning cluster analysis. These methods work by taking the records in a database and dividing (partitioning) them into the “best” K groups based on some criteria.
For more details use the link

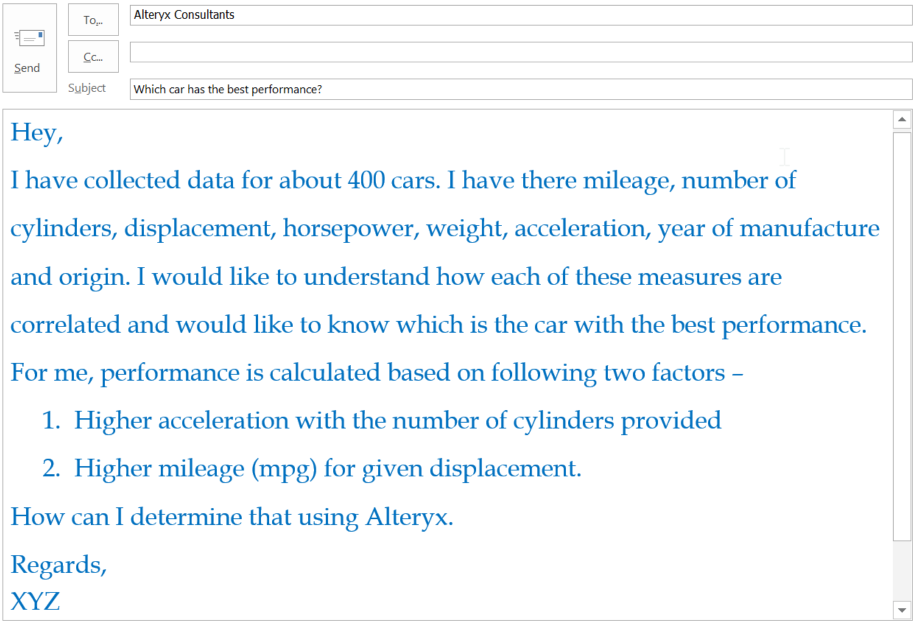
bit.ly/2vpIazo</td></tr>
</table>

14.1.20 K-Centroids Cluster Diagnostics

Concept- Assesses appropriate number of clusters for specific clustering algorithm.

<table>
<tr><td rowspan="2">
Figure-14-20 - K-Centroids Cluster Diagnostics</td><td>Tool Palette: Predictive Grouping</td></tr>
<tr><td>The K-Centroids Diagnostic tool is designed to allow the user to make an assessment of the appropriate number of clusters to specify given the data and the selected clustering algorithm (K-Means, K-Medians, or Neural Gas).
For more details use the link below.
bit.ly/2HcDKNJ</td></tr>
</table>

14.2 Which car has the best performance?



To... Alteryx Consultants

Cc...

Subject Which car has the best performance?

Send

Hey,

I have collected data for about 400 cars. I have there mileage, number of cylinders, displacement, horsepower, weight, acceleration, year of manufacture and origin. I would like to understand how each of these measures are correlated and would like to know which is the car with the best performance.

For me, performance is calculated based on following two factors –

1. Higher acceleration with the number of cylinders provided
2. Higher mileage (mpg) for given displacement.

How can I determine that using Alteryx.

Regards,
XYZ

Let us start with having the Auto dataset as the data source for the Alteryx workflow. The file is *Chapter-14-Auto.csv*. Drag the *Input Data* tool from the *In/Out* group and connect to the file.

Since we need to determine the correlation between the measures to know which car has the best performance.

This would be achieved through the below process:

- **Data Investigation** – To understand the data better using –
  a. Basic Data Profile Tool – To understand the metadata
  b. Field Summary - To view the statistical description
  c. Frequency Table – To understand the contents of the data
- Identifying the **Correlation** between data variables –
  a. Spearman Correlation

b. Pearson Correlation

Basic Data Profile tool from the *Data Investigation* group will help to understand the metadata for each of the columns present in the dataset. Add *Browse* to get a detailed report. Run (Ctrl+R) the workflow and select the *Browse* tool to view the detailed report.

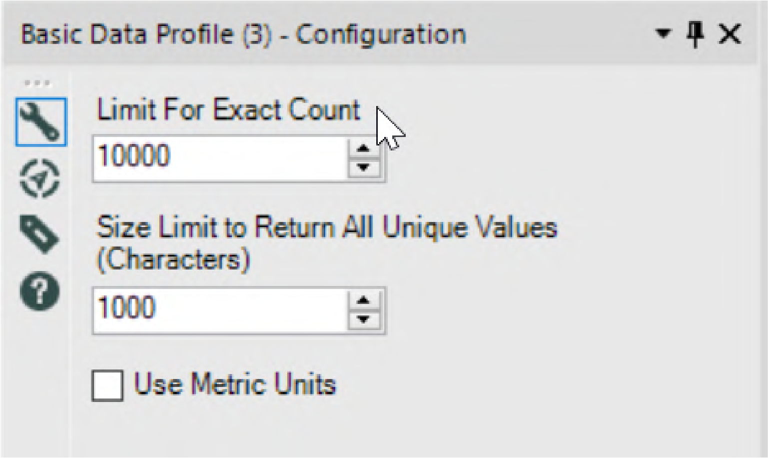
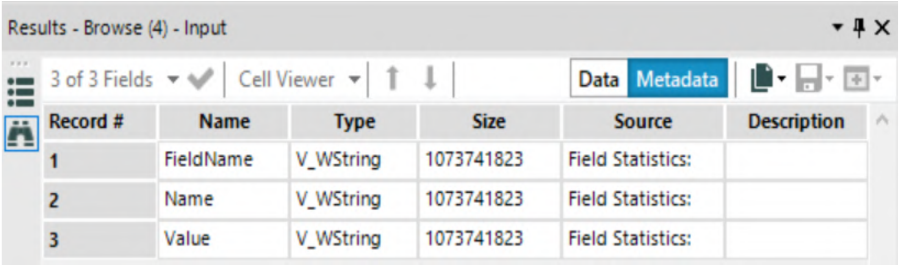


Figure 14-21-Basic Data Profile Configuration

<u>Properties Window</u>:
The *Basic Data Profile Configuration* window has three optional configuration options:

- *Limit for Exact Count*: The default limit is recommended for best performance. Increase the limit to see profile information for more data. Type or click to select the maximum number of unique values that you want Alteryx to identify in the data.
- *Size Limit to Return All Unique Values (Characters)*: The default limit is recommended for best performance. Increase the limit to see profile information for more data. Type or click to select the maximum number of characters you want Alteryx to check in a value to determine if the value
- is unique.
- *Use Metric Units*: Select to use metric units of measure. This option only applies to spatial data.

Figure 14-22-Running the Data through the Basic Data Profile



Record #	Name	Type	Size	Source	Description
1	FieldName	V_WString	1073741823	Field Statistics:	
2	Name	V_WString	1073741823	Field Statistics:	
3	Value	V_WString	1073741823	Field Statistics:	

Figure 14-23-Basic Data Profile Output

From the output, we understand that the Data Type is V_String for all the columns.

Add the Field Summary tool followed by Browse to Output O to view the statistical description for each of the columns in the data set. Select 'All' in configuration window while configuring the Field Summary tool.

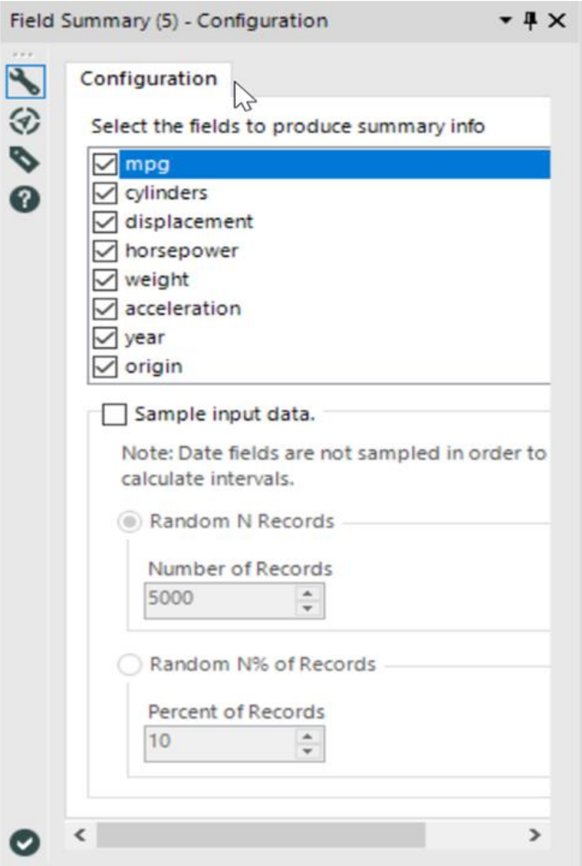


Figure 14-24-Field Summary to know the data types

<u>Properties Window</u>:
The *Field Summary Configuration* window has three optional configuration options:

- *Select the fields to product summary info:* Select the columns to analyze and include in the output.
  - *All:* Click to select all columns in the data.
  - *None:* Click to deselect previously selected columns.
- *Sample input data:* Select to include a sample of columns in the output instead of select all, or specific columns.

- *Random N Records:* Select the number of records to include in the output.
- *Random N% of Records:* Select the percentage of incoming records to include in the output.

Run (Ctrl+R) the workflow and click on *Browse* to view the output.

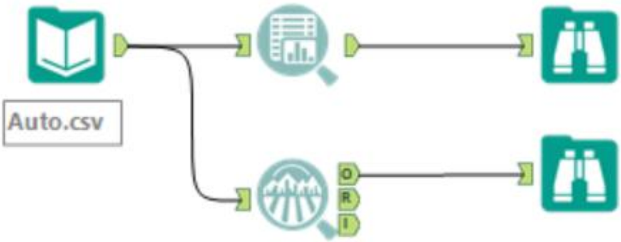
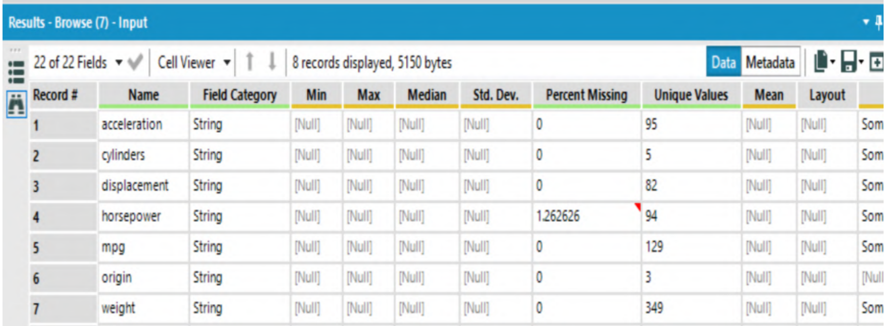


Figure 14-25-Running the Data through the Field Summary

A summary of each of the columns can be studied from the output. For example, we can observe that the percent missing is 0 for all the columns and understand that none of the columns have missing values.


Results - Browse (7) - Input

22 of 22 Fields | Cell Viewer | 8 records displayed, 5150 bytes | Data | Metadata

Record #	Name	Field Category	Min	Max	Median	Std. Dev.	Percent Missing	Unique Values	Mean	Layout	
1	acceleration	String	[Null]	[Null]	[Null]	[Null]	0	95	[Null]	[Null]	Som
2	cylinders	String	[Null]	[Null]	[Null]	[Null]	0	5	[Null]	[Null]	Som
3	displacement	String	[Null]	[Null]	[Null]	[Null]	0	82	[Null]	[Null]	Som
4	horsepower	String	[Null]	[Null]	[Null]	[Null]	1.262626	94	[Null]	[Null]	Som
5	mpg	String	[Null]	[Null]	[Null]	[Null]	0	129	[Null]	[Null]	Som
6	origin	String	[Null]	[Null]	[Null]	[Null]	0	3	[Null]	[Null]	[Null
7	weight	String	[Null]	[Null]	[Null]	[Null]	0	349	[Null]	[Null]	Som

Figure 14-26-Output of Field Summary

Except for Name, Origin and Year, rest are supposed to be numeric. So, we need to change the data type for each of those to fields to a numeric data type like *Double.* Use the *Select* tool from *Preparation* group to change the data types. Click on the dropdown under *Type* to make the appropriate selection.

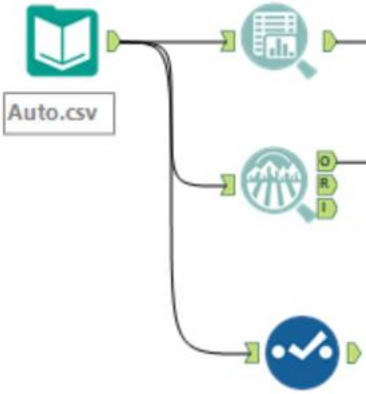


Figure 14-27-Selection of Columns for further analysis

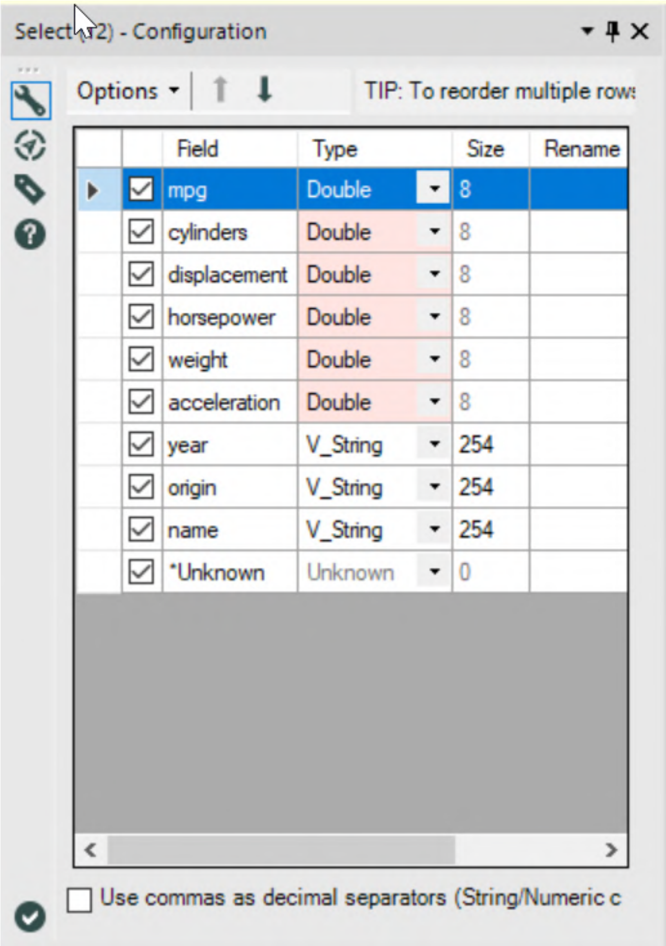


Figure 14-28-Configuration window for Select

Name column has the names of all the cars. We can use the *Formula* tool from the *Preparation* group to get the Brand of each car.

The first word in the *Name* can be considered as the Brand. The formula below can be used to get the Brand of the car.

left([name],FindString([name], " "))

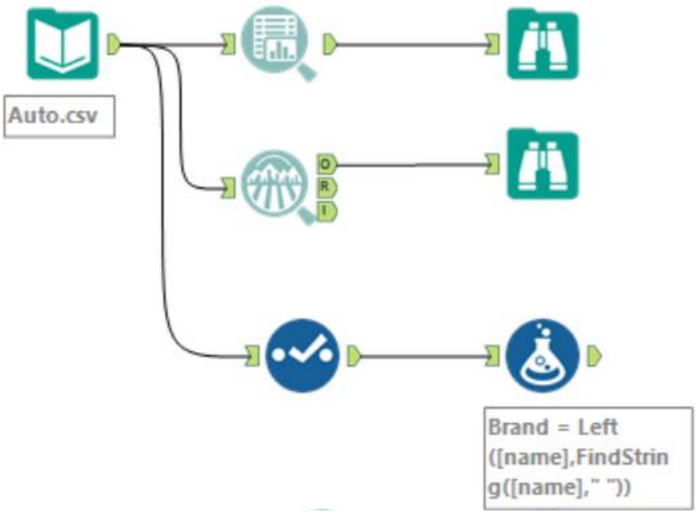


Figure 14-29-Retrieving the Brand Name

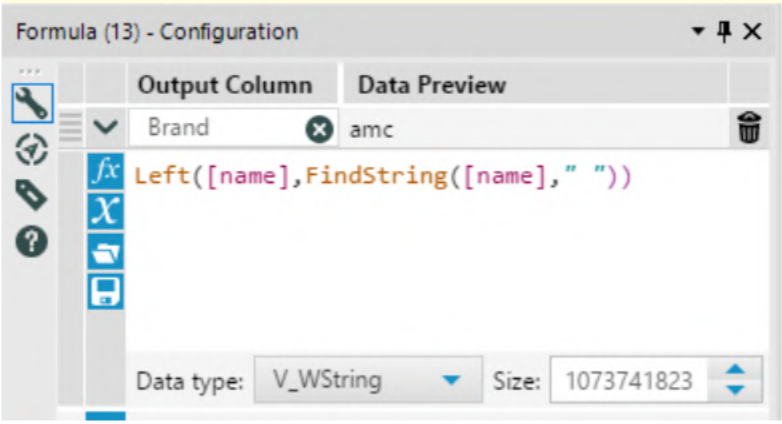


Figure 14-30-Configuration window for Formula

Add the Frequency Table followed by Browse to Report R to understand the frequency table for each categorical field selected. For each categorical field selected, a frequency table is produced. The frequency table is a summary of the data with frequency counts and percentages for each value in a field. Select 'All' in configuration window while configuring the Frequency Table.

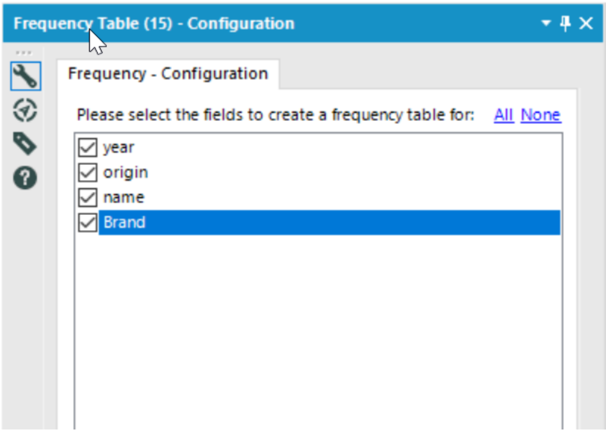


Figure 14-31-Configuration window for Frequency Table

Run (Ctrl+R) the workflow and click on *Browse* to view the output.

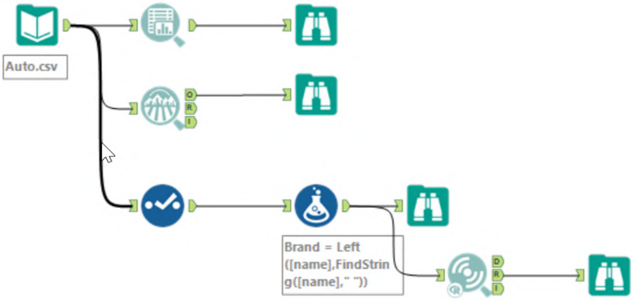


Figure 14-32-Running the Data through the Frequency Table

<u>Properties Window</u>:
The *Frequency Table Configuration* window has one component.

- *Select the fields to produce a frequency analysis for:* Certain field types cannot be selected which include:
  - FixedDecimal
  - Date, Time, DateTime
  - Blob, and
  - SpatialObj

origin

Value	Frequency	Percent	Cumulative Frequency	Cumulative Percent
1	247	62.37	247	62.37
3	79	19.95	326	82.32
2	70	17.68	396	100.00

Figure 14-33-Frequency Table Output

The Frequencies for each of the Categorical Variable can be studied from the output. For example, we can observe that the Origin with value 1 have maximum data with frequency 247 (62%) out of the total of 396 (100%).

Now as per the requirement we would like to understand the correlation between cylinders and acceleration and also between mpg and displacement to conclude on the performance of each car.

We can connect the current output to two *Spearman Correlation* tools present in the Data Investigation group and join these with the output using the *Join Multiple* tool present in the *Join* group.

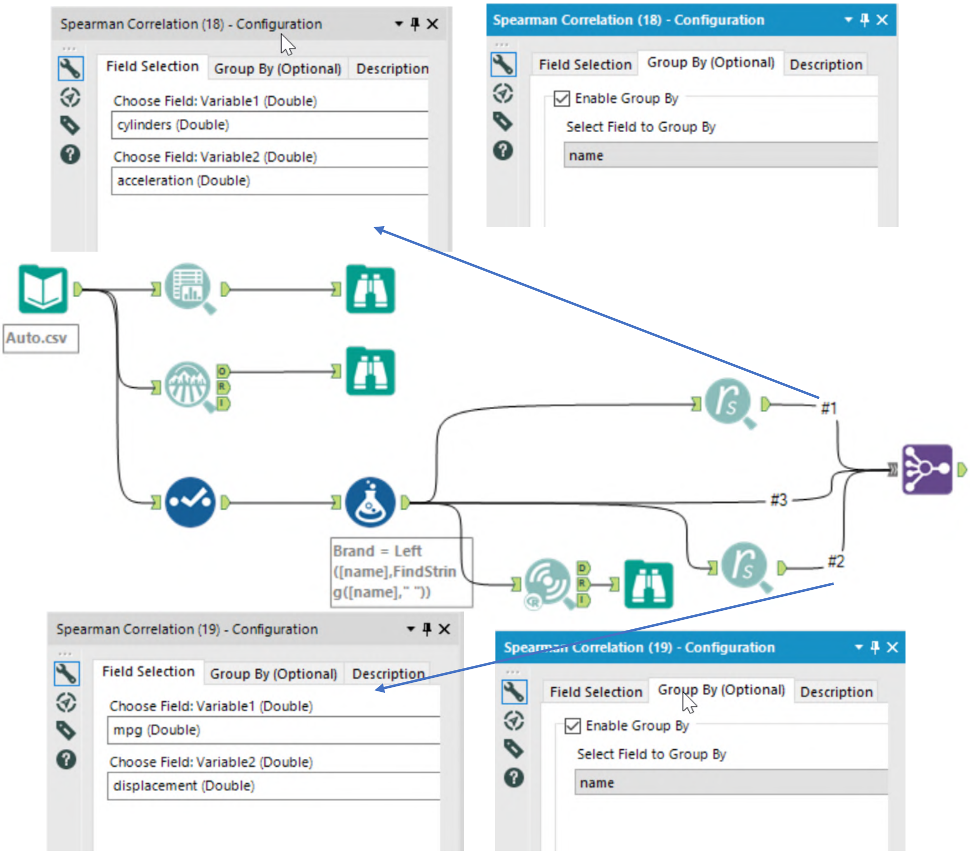


Figure 14-34-Introducing Spearman Correlation based on Grouping

The *Spearman Correlation* tools are configured as shown in the figure above. They are grouped by *Name* so that we get the values for each car.

<u>Properties Window</u>:
Spearman Correlation window have two components.

- *Field Selection tab*: Choose the two fields to determine the Correlation. The two fields must be numeric and the same field cannot be chosen twice.

Columns containing unique identifiers, such as surrogate primary keys and natural primary keys, should not be used in statistical analyses. They have no predictive value and can cause runtime exceptions.

- *Group By (Optional) tab*: To determine the correlation for a group, select Enable Group By and select the appropriate grouping field.

The outputs of this tool are similar to what is shown below.

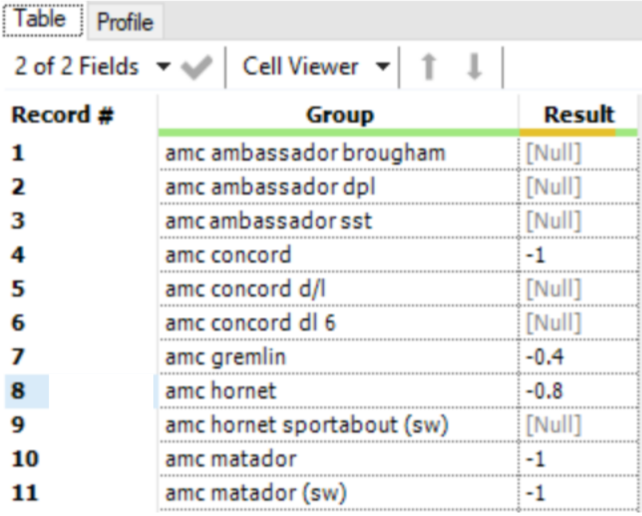


Table Profile
2 of 2 Fields Cell Viewer

Record #	Group	Result
1	amc ambassador brougham	[Null]
2	amc ambassador dpl	[Null]
3	amc ambassador sst	[Null]
4	amc concord	-1
5	amc concord d/l	[Null]
6	amc concord dl 6	[Null]
7	amc gremlin	-0.4
8	amc hornet	-0.8
9	amc hornet sportabout (sw)	[Null]
10	amc matador	-1
11	amc matador (sw)	-1

Figure 14-35-Sample Output of Spearman Correlation

These outputs are joined (using *Multiple Join*) with the previous output (output at *Formula*) using the following condition.

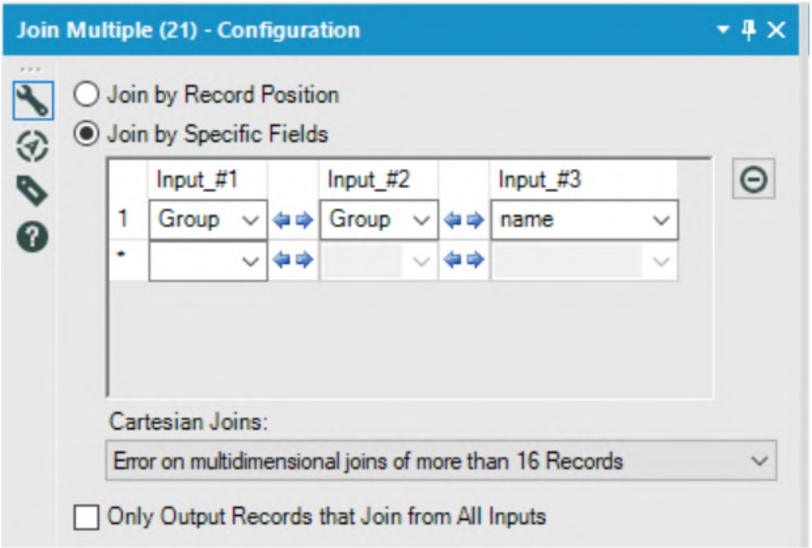


Figure 14-36-Joining the Spearman Correlation Outputs based on Group

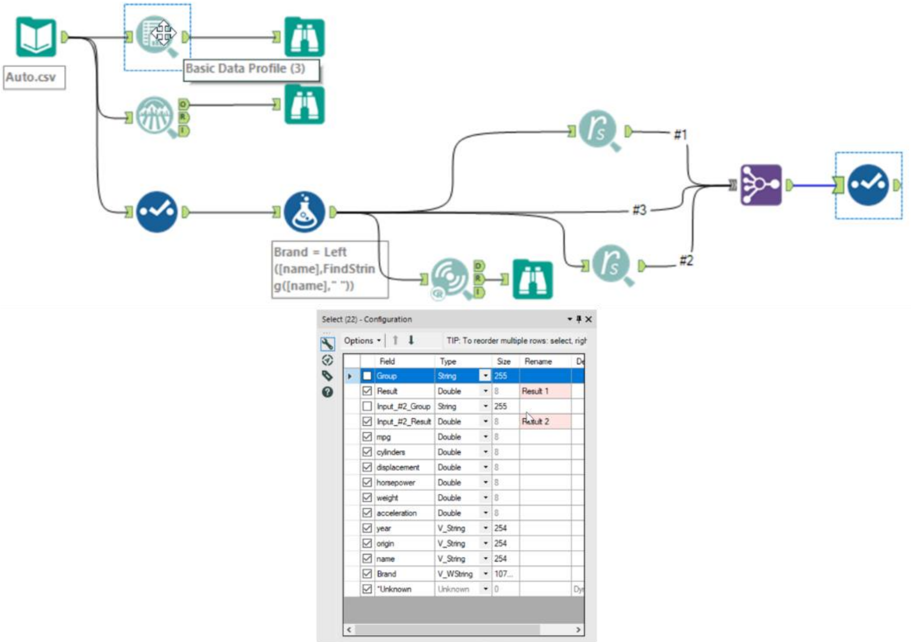


Figure 14-37-Selecting Relevant Columns for further analysis

We can use a *Select* tool after the *Multiple Join* to carry forward only the required fields and rename the ones required.

Using the *Formula* tool, we can create a column to get the priority. This can be followed by a *Sort* tool from *Preparation* group sort the data according to Priority. Add *Browse* to view the data.

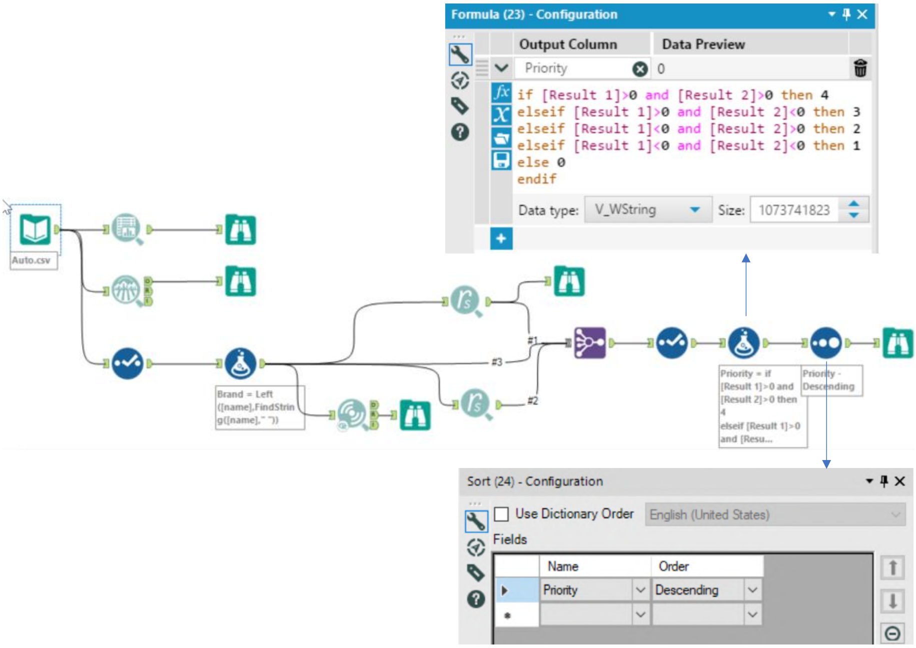


Figure 14-38-Data Transformations to calculate Priority

We get a result similar to the one displayed below.


Results - Browse (25) - Input

13 of 13 Fields | Cell Viewer | 396 records displayed, 20 KB

Record #	Result 1	Result 2	mpg	cylinders	displacement	horsepower	weight	acceleration	year	origin	name	Brand	Priority
1	1	0.4	29	4	85	52	2035	22.2	76	1	chevrolet chevette	chevrolet	4
2	1	0.4	30.5	4	98	63	2051	17	77	1	chevrolet chevette	chevrolet	4
3	1	0.4	30	4	98	68	2155	16.5	78	1	chevrolet chevette	chevrolet	4
4	1	0.4	32.1	4	98	70	2120	15.5	80	1	chevrolet chevette	chevrolet	4
5	1	1	27	4	97	88	2130	14.5	70	3	datsun pl510	datsun	4
6	1	1	27	4	97	88	2130	14.5	71	3	datsun pl510	datsun	4
7	0.5	0.5	26	4	79	67	1963	15.5	74	2	volkswagen dasher	volkswagen	4

Figure 14-39-Data Output Post Priority Calculation

Higher the *Priority* value, better is the performance.

To understand the correlation between each of the fields (only numeric), we connect the data to *Pearson Correlation* tool under *Data Investigation* group. Select all the numeric fields in the configuration window and connect it to *Browse* to view the output.

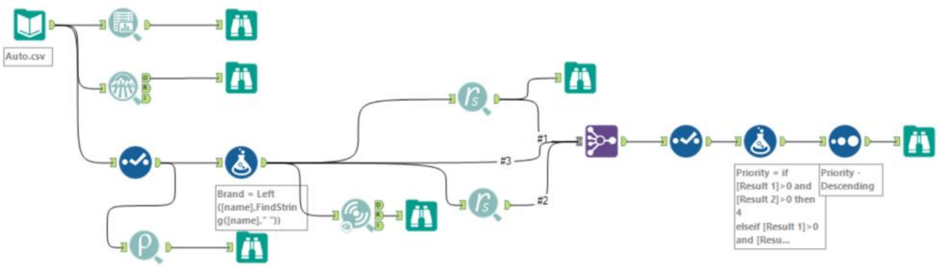


Figure 14-40-Introducing Pearson Correlation

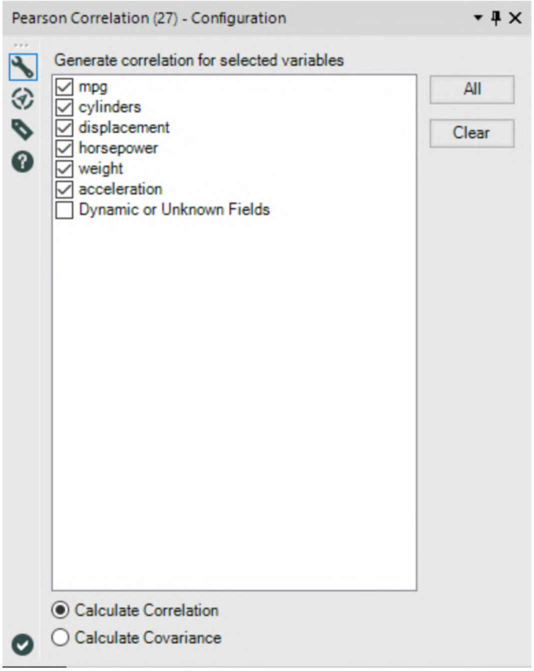


Figure 14-41-Configuration Window for Pearson Correlation

<u>Properties Window</u>:

Pearson Correlation window have two components.

- *Generate correlation for selected variables:* Select two or more fields from the input stream to run the correlation on. Fields must be numeric.

Columns containing unique identifiers, such as surrogate primary keys and natural primary keys, should not be used in statistical analyses. They have no predictive value and can cause runtime exceptions.

- Specify the type of calculation to run. Choices are:
    - *Calculate Correlation*: Measures the Pearson Correlation.
    - *Calculate Covariance*: Measures the Covariance between different fields. The type of covariance is "sample covariance", which is the same as the Excel statistical formula "COVARIANCES".

The figure below shows the output.

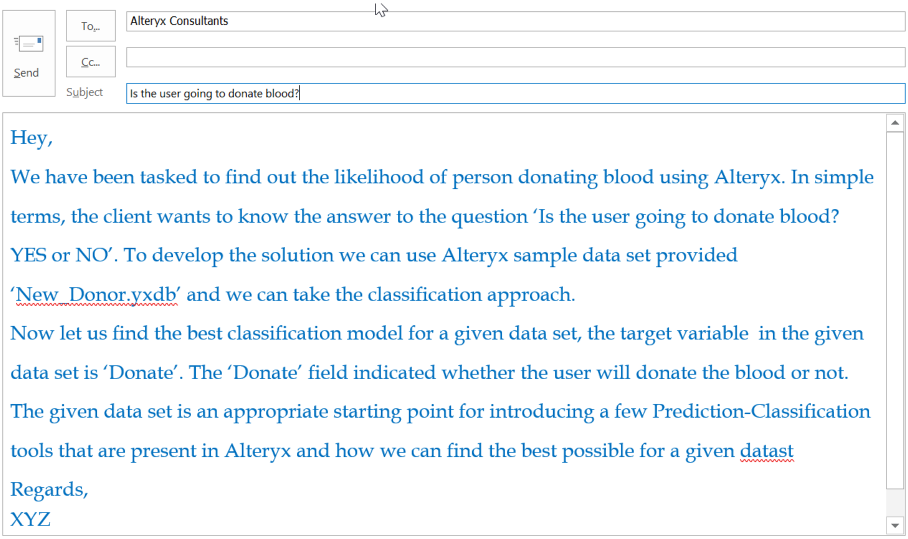
Results - Browse (28) - Input

7 of 7 Fields | Cell Viewer | 6 records displayed, 1979 bytes

Record #	FieldName	mpg	cylinders	displacement	horsepower	weight	acceleration
1	mpg	1	-0.776606	-0.804501	[Null]	-0.831784	0.422462
2	cylinders	-0.776606	1	0.951008	[Null]	0.897083	-0.505084
3	displacement	-0.804501	0.951008	1	[Null]	0.933066	-0.544692
4	horsepower	[Null]	[Null]	[Null]	1	[Null]	[Null]
5	weight	-0.831784	0.897083	0.933066	[Null]	1	-0.419933
6	acceleration	0.422462	-0.505084	-0.544692	[Null]	-0.419933	1

Figure 14-42-Pearson Correlation Output to show how variables are related to each other

14.3 Blood Donation


To... Alteryx Consultants

Cc...

Subject: Is the user going to donate blood?

Send

Hey,

We have been tasked to find out the likelihood of person donating blood using Alteryx. In simple terms, the client wants to know the answer to the question 'Is the user going to donate blood? YES or NO'. To develop the solution we can use Alteryx sample data set provided 'New_Donor.yxdb' and we can take the classification approach.

Now let us find the best classification model for a given data set, the target variable in the given data set is 'Donate'. The 'Donate' field indicated whether the user will donate the blood or not. The given data set is an appropriate starting point for introducing a few Prediction-Classification tools that are present in Alteryx and how we can find the best possible for a given datast

Regards,
XYZ

The data to be used here for the analysis is from New donor score sample example in predictive analytics since they wanted us to build it on generic data which they would then use with their dataset. The app is an extension of the existing New donor score sample example - sample workflow. This dataset is available at *Chapter-14-New_Donor*. The first step to solving this is to load the data in Alteryx and take a look at it using the browse tool.

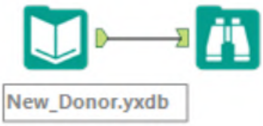


Figure 14-43-Blood Donation Input Data

From the Browse tool, we understand that the data is about all the details of various students and their activities in the college. The next step is to create sample data from this dataset to be used for the prediction, so we use a *Create Samples* tool to do that.

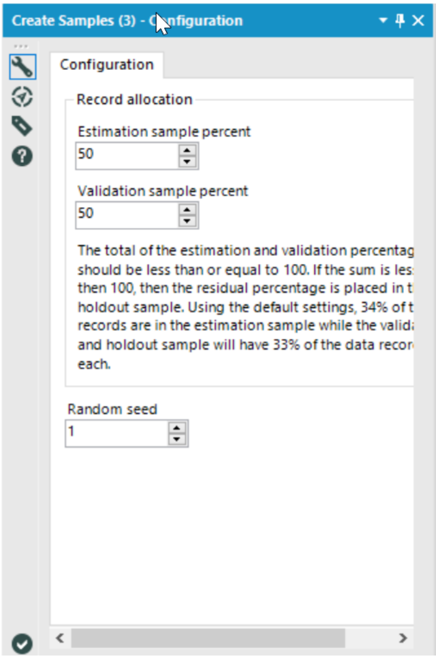


Figure 14-44-Blood Donation-Samples Configuration

<u>Properties Window</u>:

Create Samples Tool have three components.

- *Estimation sample percent:* The percentage of the data to be placed in the estimation sample (between 1% and 99%).
- *Validation sample percent:* The percentage of the data to be placed in the validation sample (between 1% and 99%).
- *Random seed:* An integer value between 1 and 1000. Changing this value will alter the sample that an individual row of the data is placed in. Unless there is a specific reason to change this value, the default value of 1 is the recommended choice.

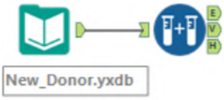


Figure 14-45-Blood Donation-Samples

We use a 50-50 split to split it right in the center and have reasonably same data.

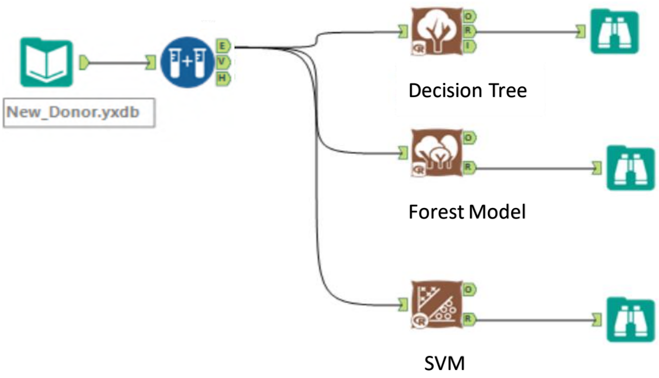


Figure 14-46-Blood Donation-Predictive Analytics Tools

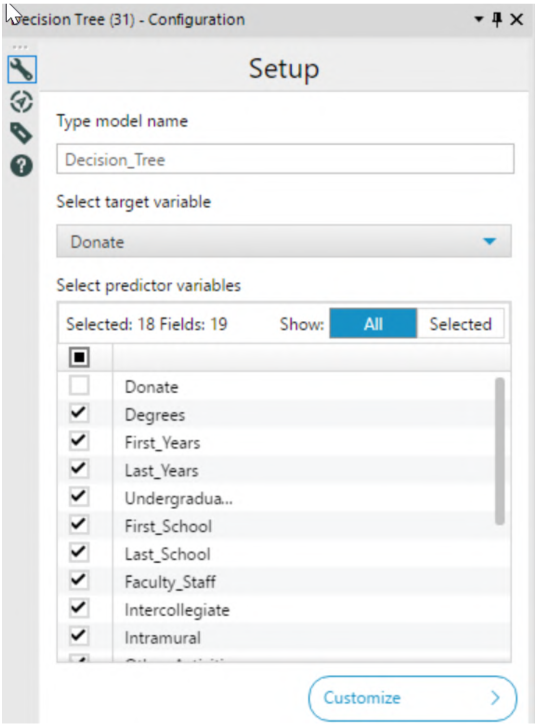


Figure 14-47-Decision Tree Configuration Window

Properties Window:
In *Decision Tree Tool,* below options are required to generate a decision.

- *Type model name*: A name for the model that can be referenced by other tools. The model name or prefix must start with a letter and may contain letters, numbers, and the special characters period (".") and underscore ("_"). R is case sensitive.
- *Select target variable*: The data field to be predicted, also known as a response or dependent variable.
- *Select predictor variables*: The data fields used to influence the value of the target variable, also known as a feature or independent variable. One predictor field is required at a minimum, but there is no upper limit on the number of predictor fields selected. The target variable itself should not be used in calculating the target value, so the target field should not be included with the predictor fields.

Columns containing unique identifiers, such as surrogate primary keys and natural primary keys, should not be used in statistical analyses. They have no predictive value and can cause runtime exceptions.
Click Customize to adjust additional settings

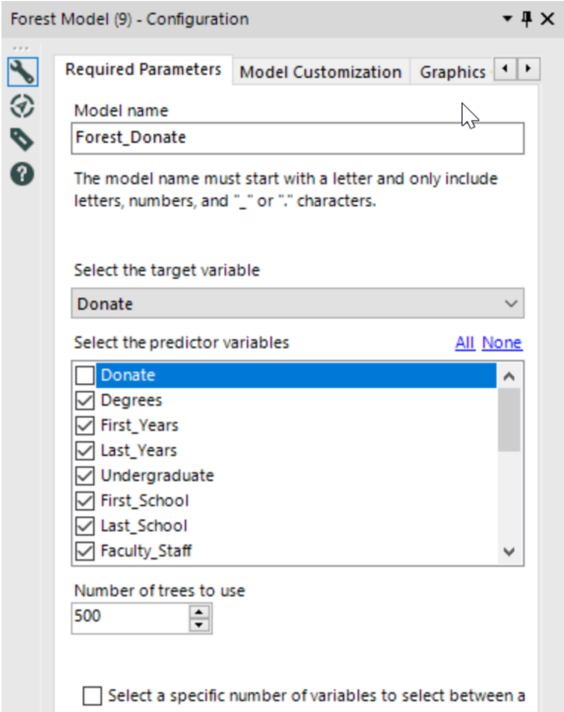


Figure 14-48-Forest Model Configuration Window

Properties Window:

Forest Model Tool have below required components.

- *Model name*: Type a name for the model to identify the model when it is referenced in other tools. Model names must start with a letter and may contain letters, numbers, and the special characters period (.) and underscore (_). No other special characters are allowed, and R is case sensitive.
- *Select the target variable:* Select the data to be predicted. A target variable is also known as a response or dependent variable.
- *Select the predictor variables*: Select the data to use to influence the value of the target variable. A predictor variable is also known as a feature or an independent variable. Any number of predictor variables can be selected, but the target variable

should not also be a predictor variable. Each categorical predictor variable can have a maximum of 32 classes.

Columns containing unique identifiers, such as surrogate primary keys and natural primary keys, should not be used in statistical analyses. They have no predictive value and can cause runtime exceptions.

- *Number of trees to use*: Select the number of tree models to include in the forest. The default is 500 based on the finding of Breiman. Decrease the value with an XDF metadata stream if the length of model runtime is a concern.
- *Select a specific number of variables to select between at each split*: Select the number of variables to be considered at each split.

Click Model Customization to modify the model settings.

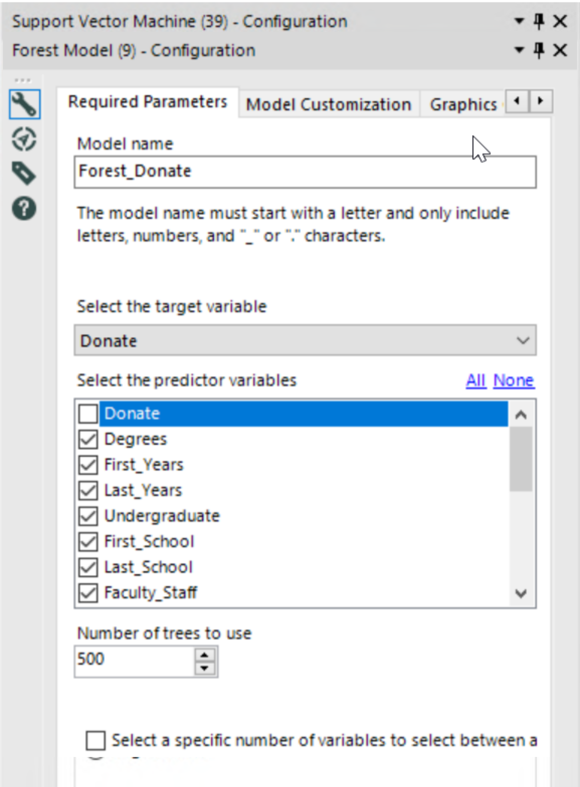


Figure 14-49-Forest Model Configuration Window

<u>Properties Window</u>:

Support Vector Machine have below required parameters.

- *Model Name:* Each model needs a name so it can later be identified. Model names must start with a letter and may contain letters, numbers, and the special characters period (".") and underscore ("_"). No other special characters are allowed, and R is case sensitive.
- *Select the Target Field*: Select the field from the data stream you want to predict.

- *Select the Predictor Fields:* Choose the fields from the data stream you believe "cause" changes in the value of the target variable.

Columns containing unique identifiers, such as surrogate primary keys and natural primary keys, should not be used in statistical analyses. They have no predictive value and can cause runtime exceptions.

- Choose the *Method* of classification or regression based on the target variable you want to predict. Generally, if the target variable you choose is string or Boolean type, it is probably a classification problem. If it is numeric type, chances are it is a regression problem.
  - *Classification*
    - C-classification: Optimizes the decision plane while allowing for some amount of error
    - nu-classification: Similar to C-classification but enables the user to limit the amount of error by selecting the value of nu.
  - *Regression*
    - *epsilon regression*
    - *nu regression*: Similar to epsilon regression but enables the user to limit the amount of error by selecting The value of nu.

Let us start with the idea that, we have to classify donor and say whether the donate field is 'YES' or 'NO'. For the purpose of this case the tools used are *Decision Tree, Random Forest,* and *Support Vector Machine* tool. *Note*: The use of Random forest will help us negate the overfitting that can happen with a decision tree. The sampling of data will split the data into estimation and validation sets.

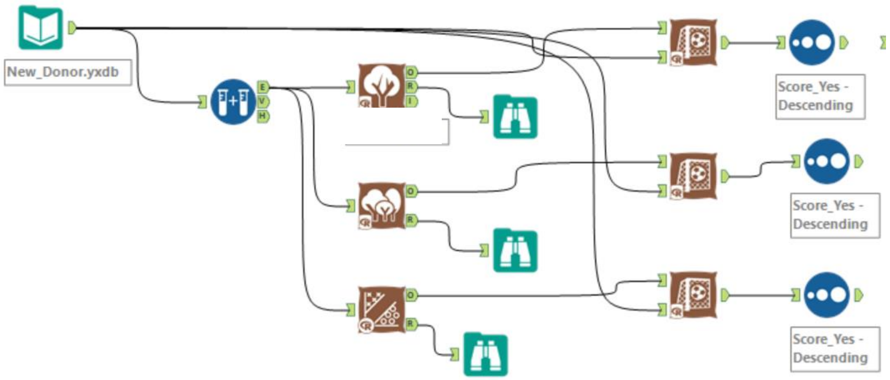


Figure 14-50-Blood Donation – Predictive Analytics Tools Scoring

The output from the models is scored against the input data and then fed into a sorting tool. We can compare the outputs from the sorted data to see how the models perform.

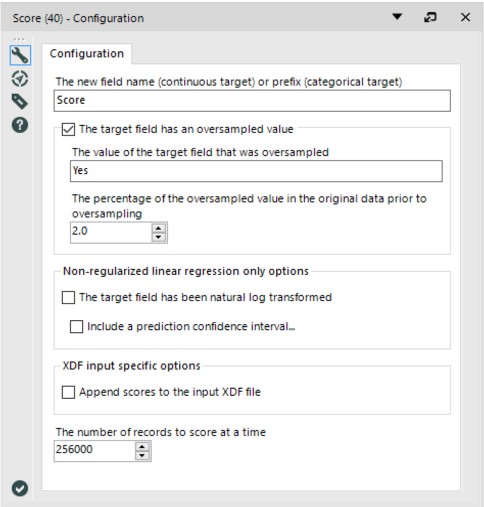
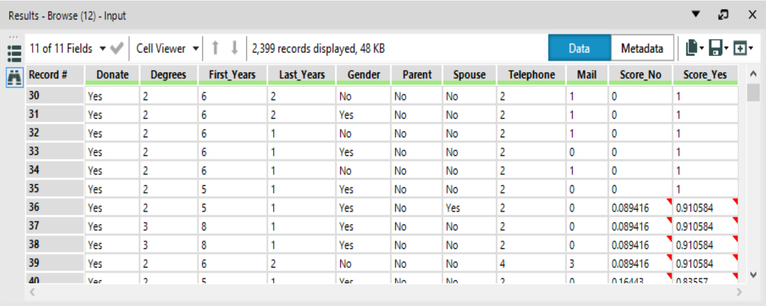


Figure 14-51-Blood Donation –Score Tool Configuration

We have set up the configuration for the scoring model as shown above to get the appropriate solution. We can use this technique to determine the best prediction model.

Here is a sample of the output generated from one model's score.


Results - Browse (12) - Input

11 of 11 Fields | Cell Viewer | 2,399 records displayed, 48 KB | Data | Metadata

Record #	Donate	Degrees	First_Years	Last_Years	Gender	Parent	Spouse	Telephone	Mail	Score_No	Score_Yes
30	Yes	2	6	2	No	No	No	2	1	0	1
31	Yes	2	6	2	Yes	No	No	2	1	0	1
32	Yes	2	6	1	No	No	No	2	1	0	1
33	Yes	2	6	1	Yes	No	No	2	0	0	1
34	Yes	2	6	1	No	No	No	2	1	0	1
35	Yes	2	5	1	Yes	No	No	2	0	0	1
36	Yes	2	5	1	Yes	No	Yes	2	0	0.089416	0.910584
37	Yes	3	8	1	Yes	No	No	2	0	0.089416	0.910584
38	Yes	3	8	1	Yes	No	No	2	0	0.089416	0.910584
39	Yes	2	6	2	No	No	No	4	3	0.089416	0.910584

Figure 14-52-Blood Donation –Score Tool Sample Output

The final output shows us that the higher the *Score_Yes* value is, higher is the chance of the person donating blood.

The *Blood Donation* data stream should look like this when it's complete.

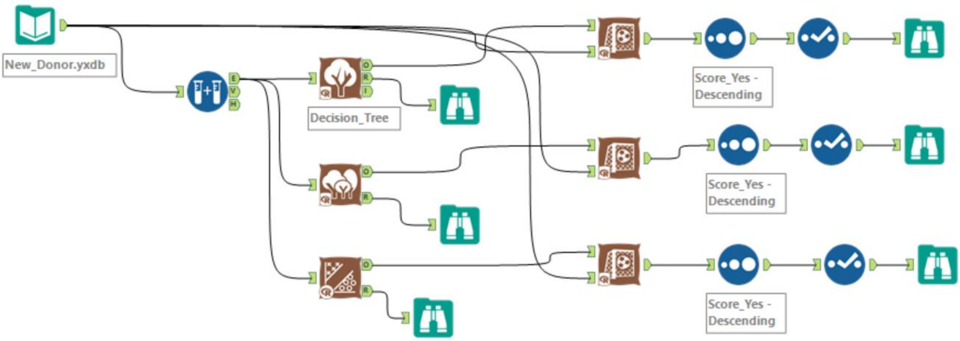
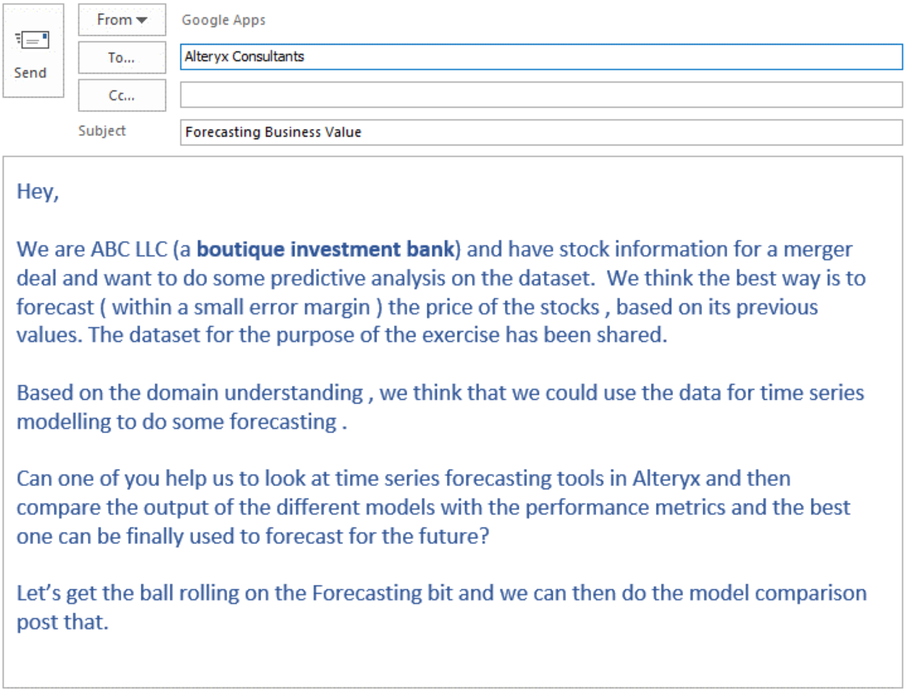


Figure 14-53-Blood Donation –Final Workflow when complete

14.4 Boutique Investment Bank


Send
From
Google Apps
To... Alteryx Consultants
Cc...
Subject Forecasting Business Value

Hey,

We are ABC LLC (a **boutique investment bank**) and have stock information for a merger deal and want to do some predictive analysis on the dataset. We think the best way is to forecast (within a small error margin) the price of the stocks , based on its previous values. The dataset for the purpose of the exercise has been shared.

Based on the domain understanding , we think that we could use the data for time series modelling to do some forecasting .

Can one of you help us to look at time series forecasting tools in Alteryx and then compare the output of the different models with the performance metrics and the best one can be finally used to forecast for the future?

Let's get the ball rolling on the Forecasting bit and we can then do the model comparison post that.

To start working on this deliverable, we will need to keep Predictive analytics sample workflow 14 as a reference and use any stock data from the public forum, here we have used a dataset from google. We can get the dataset from *Chapter 14-Google.csv*.

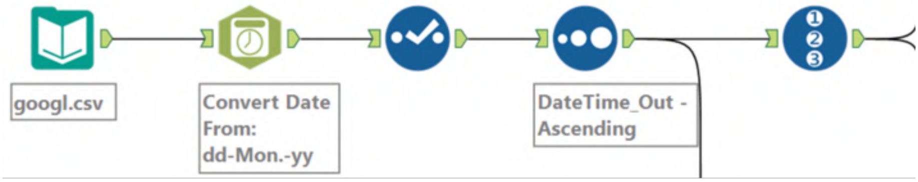


Figure 14-54 – Boutique Investment Firm Input Section

Once we input the data, we would need to convert the date string to date in the required format since that is the most important metric for the analysis. We then select the appropriate columns which are Date Time_Out and Close and sort it in ascending order of the Dates. Post that we add a column for record count using *Record ID* tool. This is relevant when have to split the input into 90% and 10% for training and verification purposes to make it ready for the Predictive Analysis.

Using the 90% of the data train the time series models you want to test.

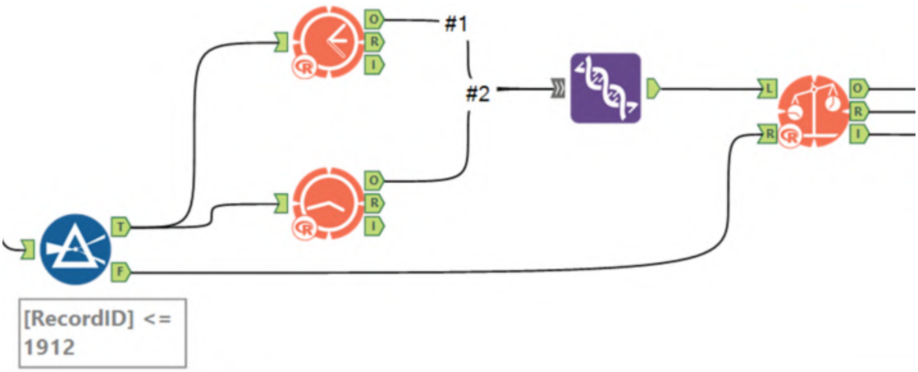


Figure 14-55 – Boutique Investment Firm -ETS Model and ARIMA

The Model1 is an *ETS model* and Model 2 is an *ARIMA Model*. For the *ETS Model*, we need to make sure we select the appropriate target field and target field frequency in the tool configuration. A union of these two is fed to a TS compare tool that returns various parameters than can be used to determine a suitable model for the given data.

Suppose we use the *ARIMA Model* with the given data set, the output from this tool is fed to a TS forecast tool with the number of periods into the future the forecast is necessary(configuration). At the end of the workflow, we would have tested out the time series models and then used the best one for forecasting.

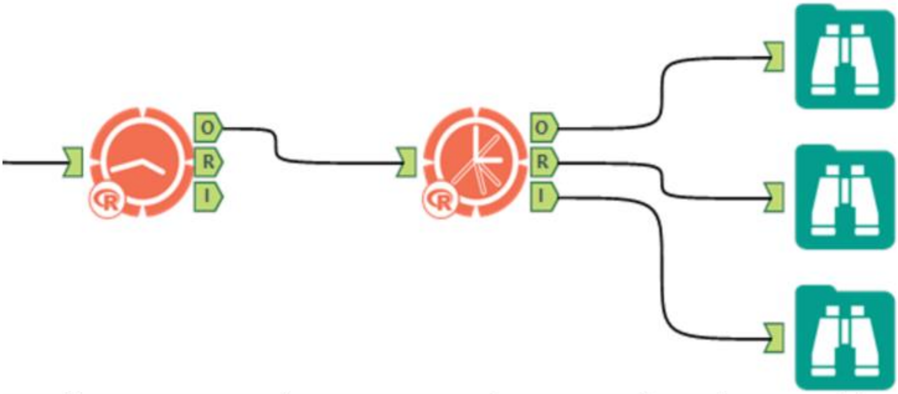


Figure 14-56 – Boutique Investment Firm -Comparison of ETS Model and ARIMA using TS Compare

When we compare the model outputs using the TS compare we get three outputs – *Output*, *Report,* and *Interactive.* The output is of highest importance since it provides us the complete statistics of the models' performance using the training set. The two main components to understand here are the mean absolute percentage forecast error (MAPE), and the mean absolute scale error (MASE). The MASE is a better metric to compare the numbers and the lower the value, the better is the fit. In our case, we can see that the MASE for ARIMA is lower than ETS.

Results - Browse (12) - Input

8 of 8 Fields | Cell Viewer | 2 records displayed, 1732 bytes

Record #	Model	ME	RMSE	MAE	MPE	MAPE	MASE	NA
1	Stock_Close_ETS	93.4274	117.6477	96.5755	8.5591	8.897	19.6321	[Null]
2	Stock_Close_ARIMA	65.4832	88.6906	70.8023	5.9423	6.5114	14.3929	[Null]

Figure 14-57 – Boutique Investment Firm -Statistics of ETS Model and ARIMA using TS Compare

The ARIMA model hence is a better fit to the current stock data. The report generated from the tool is very important to note the exact values about the Model.

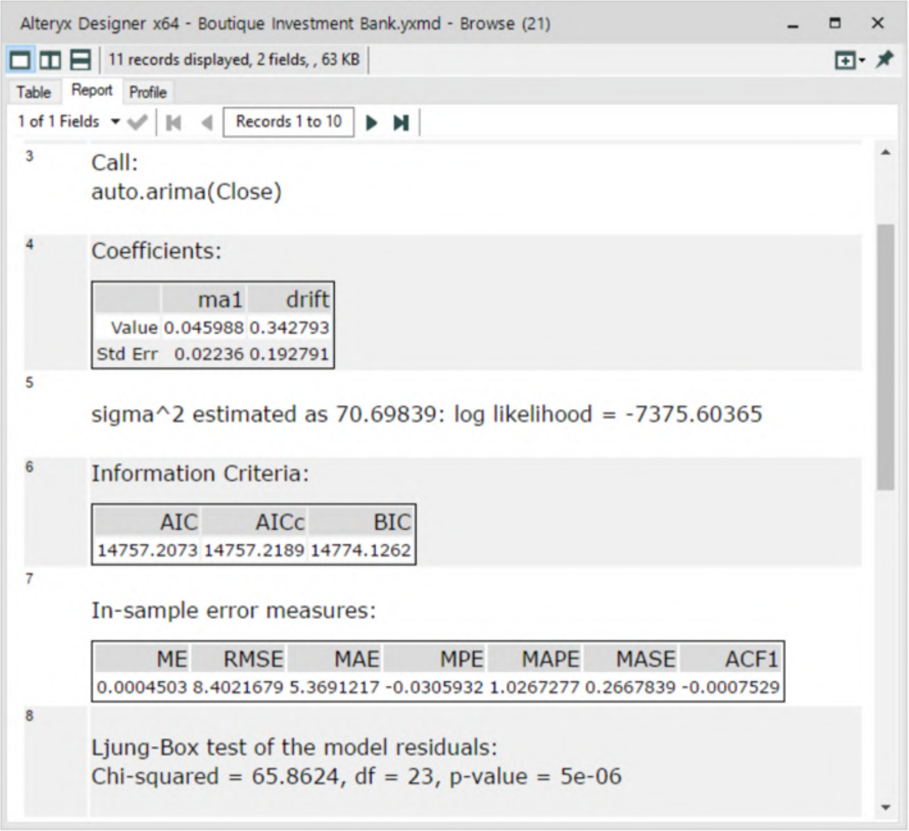


Figure 14-58 – Boutique Investment Firm -ARIMA Model Statistics

This report has all the details about the model and also contains forecasted values for a given date range. The final workflow would be in the below format

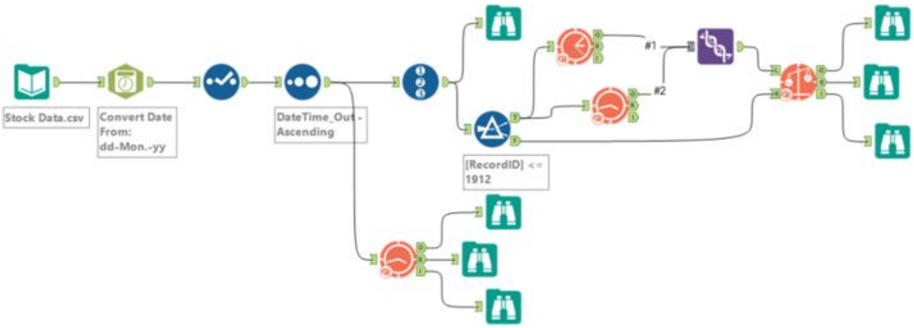


Figure 14-59 – Boutique Investment Firm stream on completion

14.5 Super Store Sales

Send

To... Alteryx Consultants

Cc...

Subject How is Discounts affecting the Sales and Profit

Hey,

I have sales data from my Super Stores' POS. I have the following data points –

- Geographic Data: Country, State, City
- Customer Segment
- Product Category
- Sales ($), Sales (Quantity), Discount, Profit.

I would like to understand how each of these measures are correlated and would like to know which is State have star performer Super Stores. For me, performance is calculated based on following two factors –

1. Less Discount Higher Sales
2. Less Discount Higher Profit

How can I determine that using Alteryx.

Regards,

14.6 Property Price Prediction

To... Alteryx Consultants

Subject Property Price Prediction

Hey,

I have data for house properties on sale, I need you to use Alteryx predictive analysis to estimate the property price.

Additionally, design a workflow that will compare the actual price Vs predicted price.

Thanks.

CHAPTER 15

Self-Guided Solutions

Readers are strongly encouraged not to use these solutions when solving the self-guided examples. There are many approaches to answering these problems and figuring out how to do it on your own is the goal of these exercises. These screenshots of solutions are provided so that if you're stuck, you can refer to the logic of how the examples direct you to a solution.

15.1 The Games: How are we doing?

Since the complete workflow is large, it is broken down into 2 parts for better understanding.

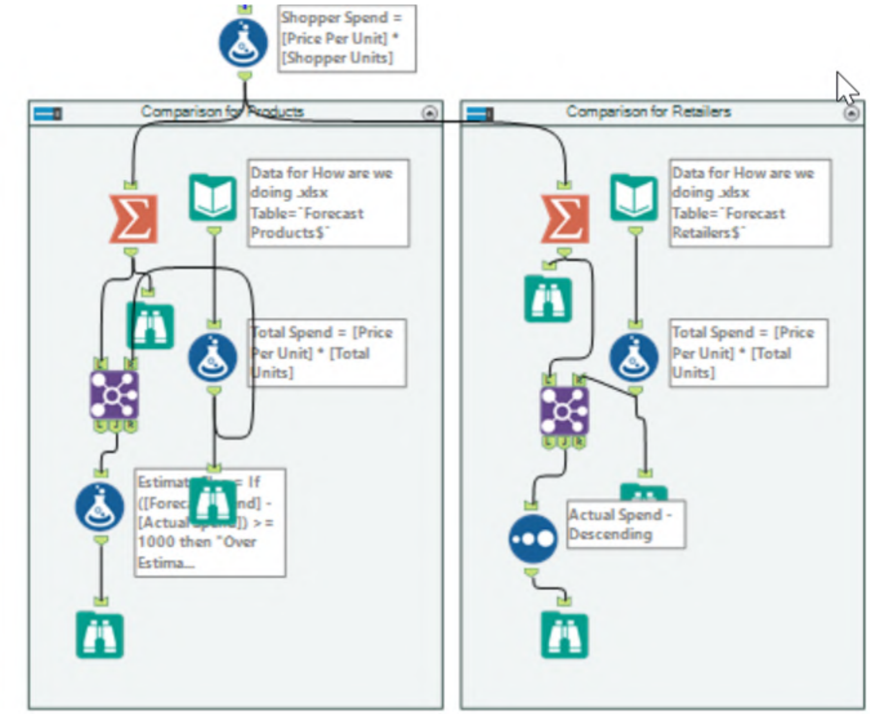


Figure 15-1 How are we doing - part1

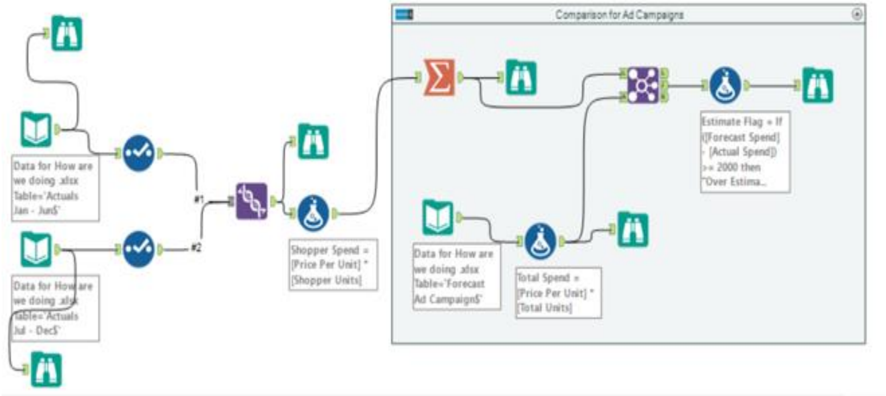


Figure 15-1 How are we doing - part2

15.2 Unisex Baby Names: What's In a Name?

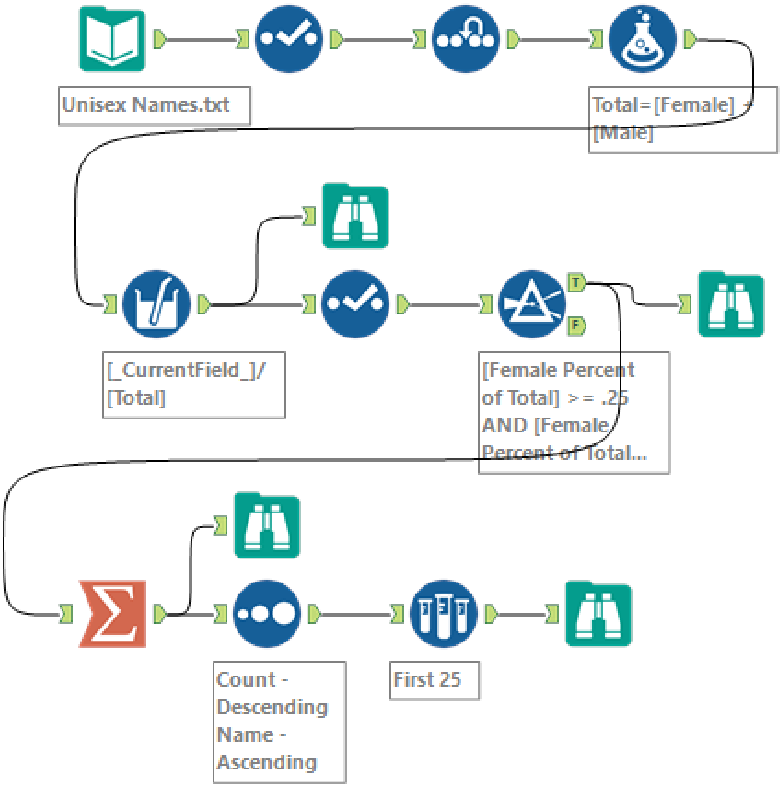


Figure 15-2 What's in a Name

15.3 The Direct Approach: Where, Oh Where Have My Three Files Gone?

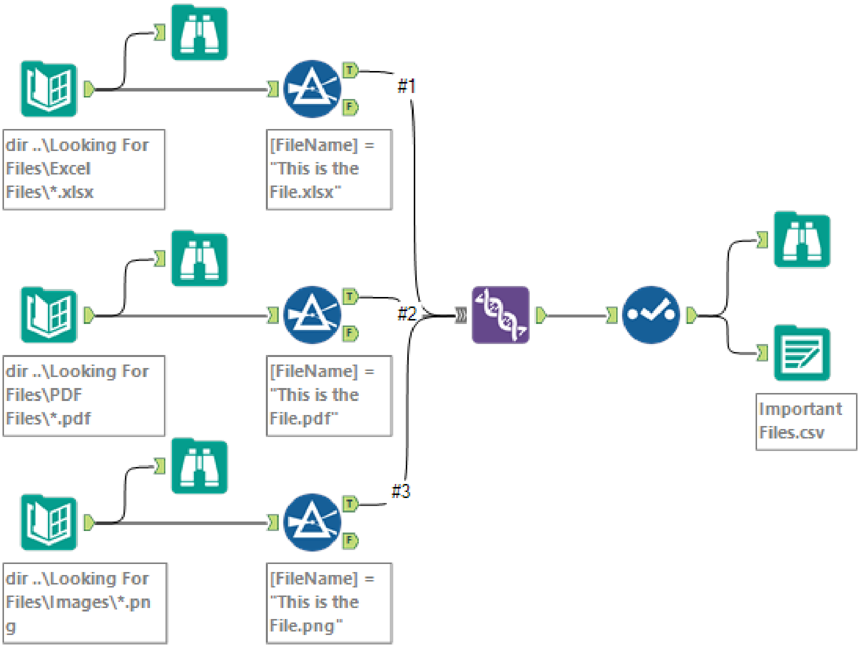


Figure 15-3 Where Have my Three Files Gone

15.4 Cultural Musing: Culturally Divided

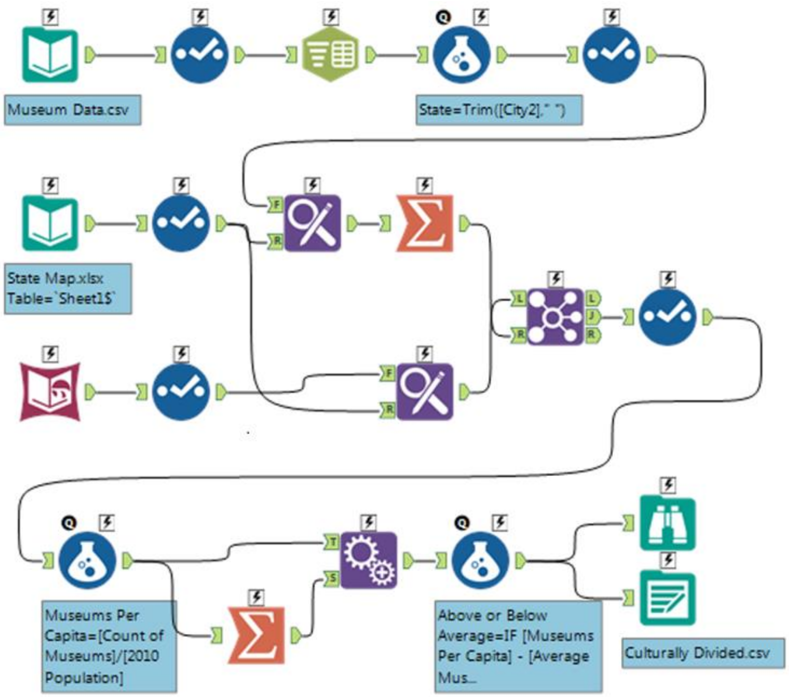


Figure 15-4 – Culturally Divided

15.5 The Sport Report: FOOTBALL!

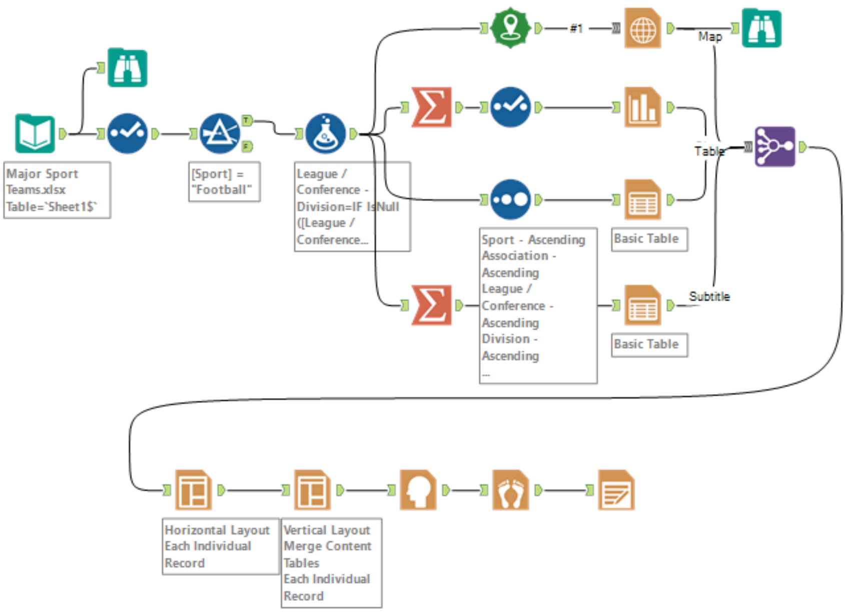


Figure 15-5 Football!

15.6 Expensive Beauty Products: More Flags

Since the complete workflow is large, it is broken down into segments for better understanding.

Figure 15-7 More Flags

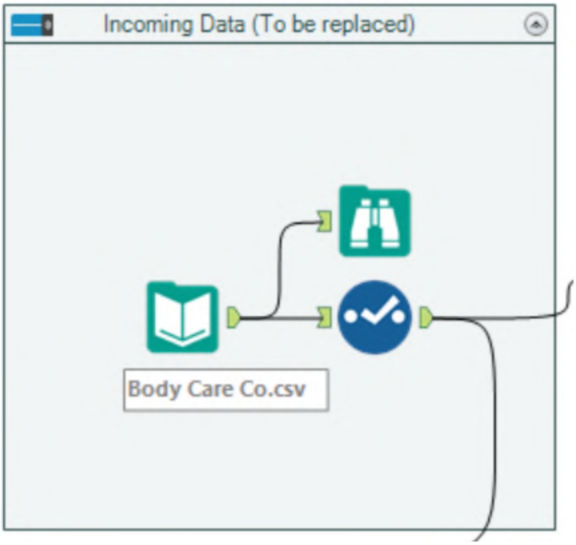


Figure 15-6a – Incoming Data

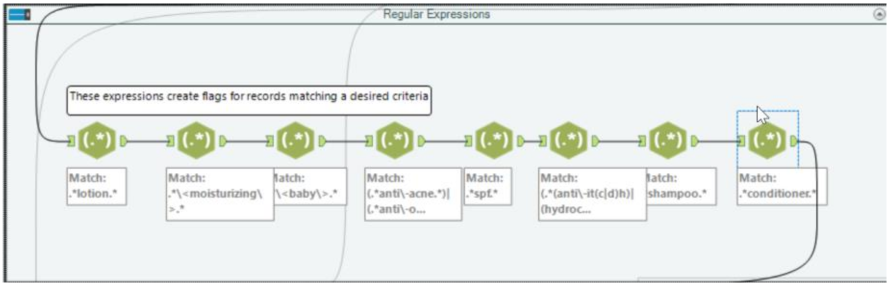


Figure 15-6b -Regular Expressions

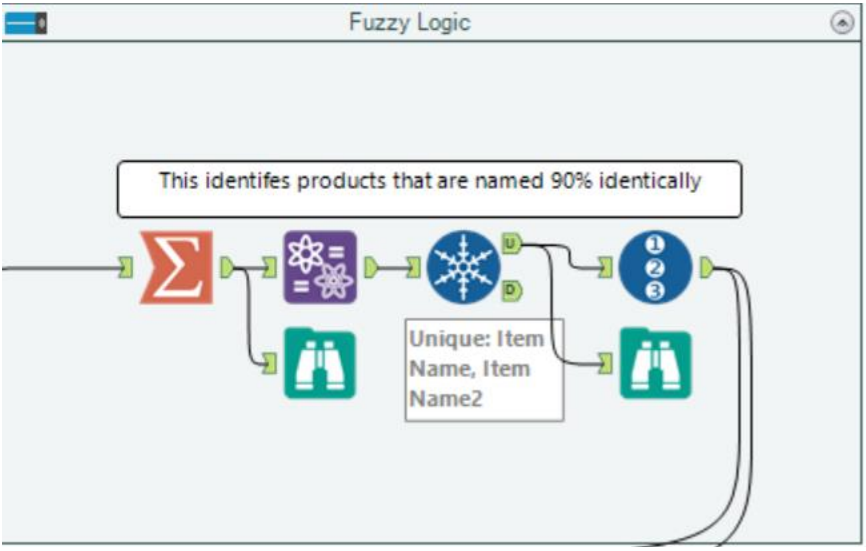


Figure 15-6c – Fuzzy Logic

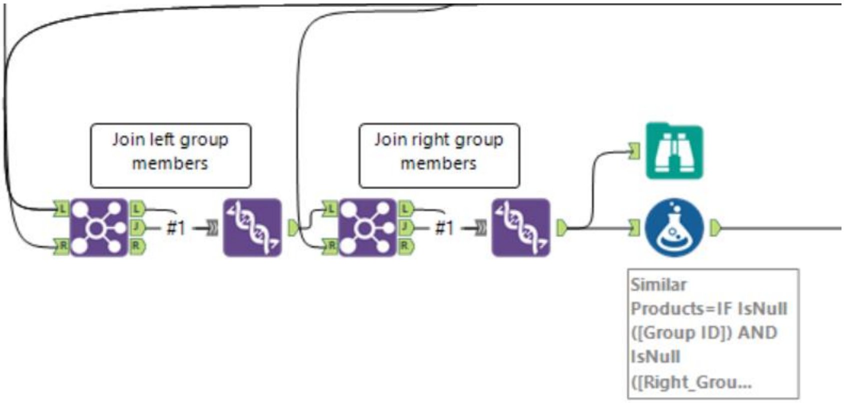


Figure 15-6d – Join & Group

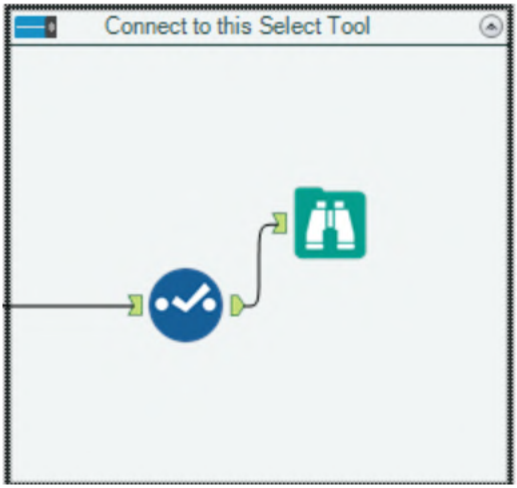


Figure 15-6e - Output

15.7 Applications Wanted: To Summarize or Not to Summarize

Since the complete workflow is large, it is broken down into segments for better understanding.

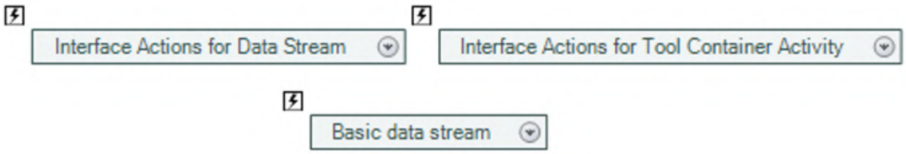


Figure 15-7 – To Summarize or Not to Summarize

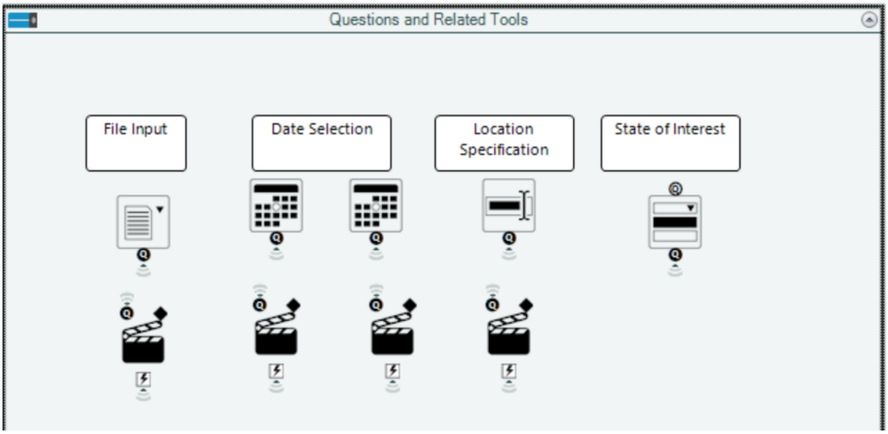


Figure 15-7a -Interface Actions for Data Stream

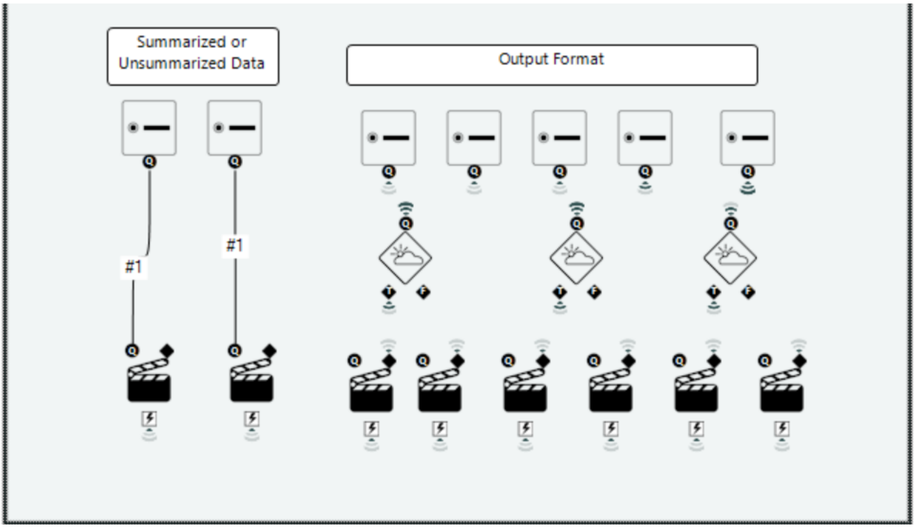


Figure 15-7b – Interface Actions for Tool Container Activity

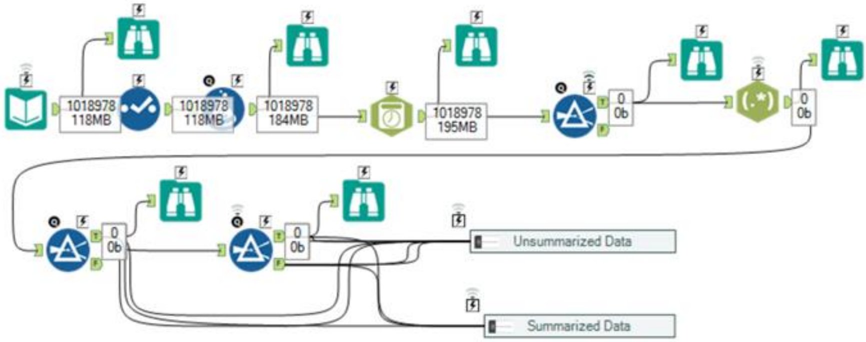


Figure 15-7c – Basic Data Stream

15.8 Where's the Joe?

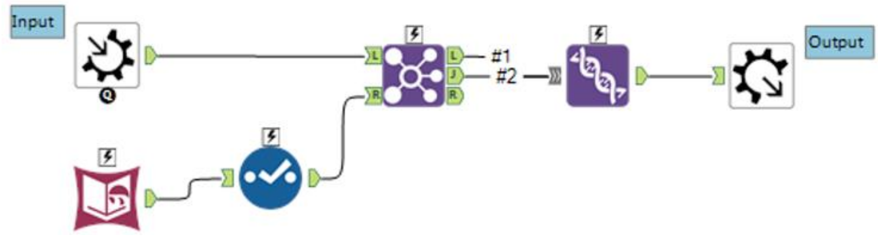


Figure 15-8a – *Who Wants Coffee?* – Macro

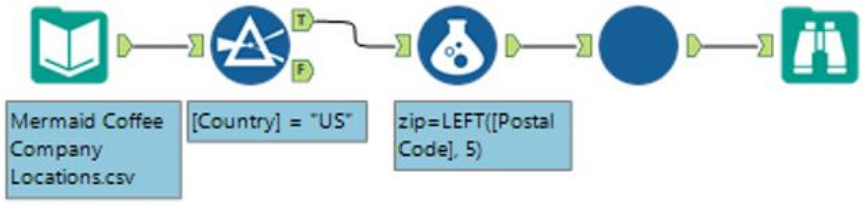


Figure 15-8b – *Who wants Coffee?* - Workflow

15.9 Metamorphosis: Narrowing Down the Search

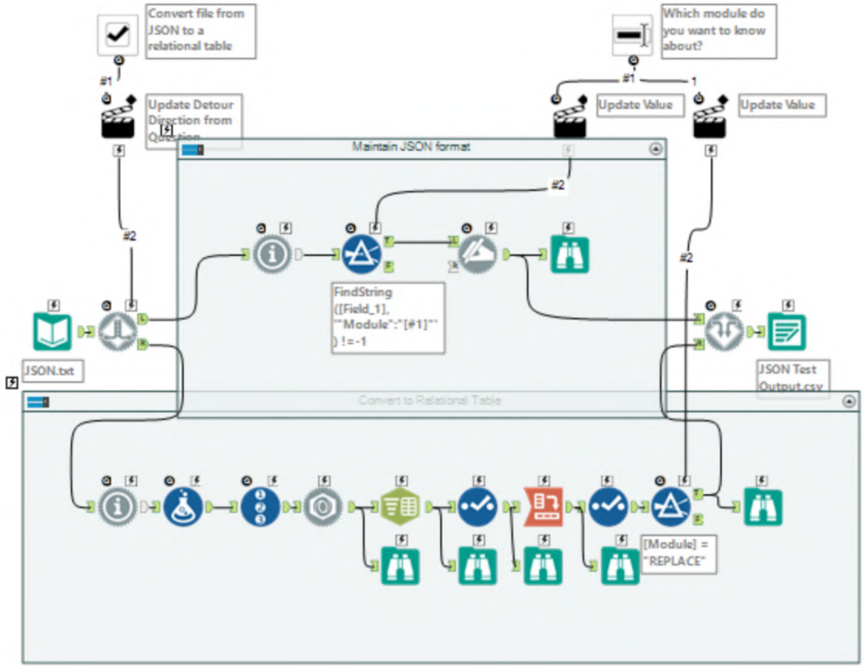


Figure 15-9 -Metamorphosis

15.10 How's the oil market doing?

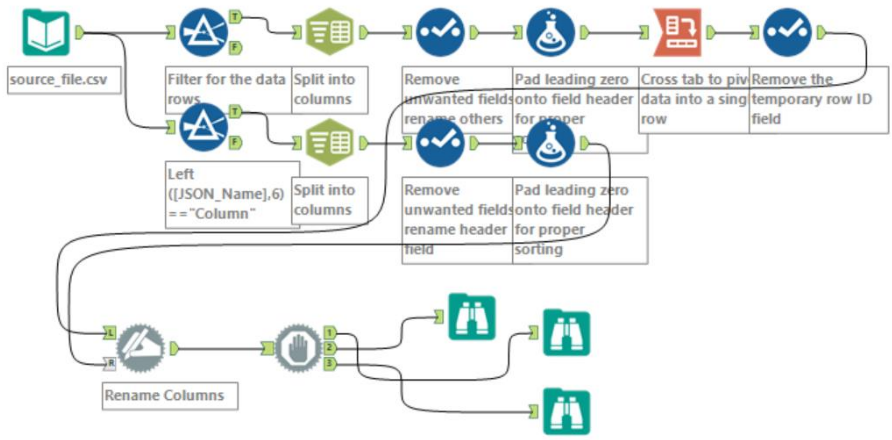


Figure 15-10 -How's the oil Market Doing?

15.11 Super Store Sales

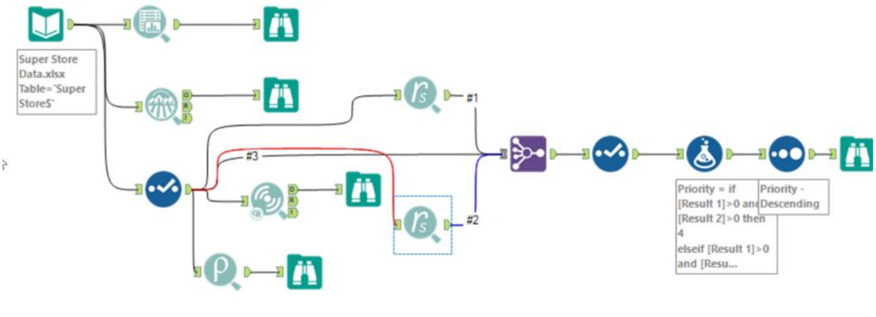


Figure 15-11 -Super Store Sales

15.12 Property Price Prediction

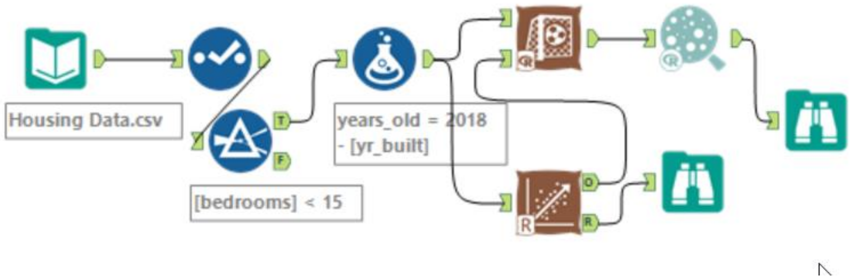


Figure 15-12 -Property Price Prediction

Appendices

Appendix A – File Types

.aws	Allocate Workspace
.cydb	Calgary database
.pcxml	XML data output type that allows the end user to select from multiple file types for a final output when using Alteryx Gallery
.yxdb	Alteryx database
.yxft	Alteryx Header File
.yxlc	Alteryx License File
.yxmc	Alteryx Macro File
.yxmd	Alteryx Workflow File
.yxwv	Alteryx Analytic App Values File which allows predefined selections to be made.
.yxwz	Alteryx Analytic App File
.yxzp	Packaged Alteryx File

Appendix B – Hot Keys

Ctrl+Alt+B	Show/Hide Toolbar
Ctrl+Alt+D	Show/Hide Interface Designer
Ctrl+Alt+O	Show/Hide Output
Ctrl+Alt+P	Show/Hide Properties
Ctrl+Alt+T	Show/Hide Tool Palette
Ctrl+Alt+V	Show/Hide Overview
Ctrl+C	Copy
Ctrl+F	Fine
Ctrl+F4	Close workflow
Ctrl+N	New workflow
Ctrl+O	Open workflow
Ctrl+R	Run/Cancel Workflow
Ctrl+S	Save workflow
Ctrl+Shift+B	Add a Browse tool after all selected tools that a Browse tool can be connected to
Ctrl+V	Paste
Ctrl+X	Cut
Ctrl+Y	Redo
Ctrl+Z	Undo
Ctrl++	Vertically align selected tools
Ctrl+-	Horizontally align selected tools
Del (Delete)	Deletes selected part of workflow
F1	Open Help menu
F5	Refresh Configuration

Appendix C – Downloads/Content

US 2010 Census SF1
http://downloads.alteryx.com/data.html

USGS North America Map
http://downloads.alteryx.com/data.html

Visual Analytics Kit (Tableau)
http://alteryx.com/kit

Visual Analytics Kit (Qlik)
http://pages.alteryx.com/VisualAnalyticsKitforQlik_Reg-LP.html

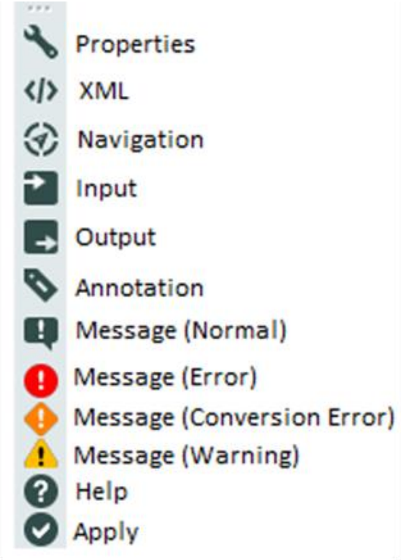
Alteryx Gallery
https://gallery.alteryx.com

Appendix D – Field Types

Type	Class	Description
Bool	Boolean	A field with two values, True and False
Byte	Numeric	A field with integers between 0 and 255
Int16	Numeric	A field with integers between -32,768 and 32,767
Int32	Numeric	A field with integers between -2,147,483,648 and 2,147,483,647
Int64	Numeric	A field with integers between –9,223,372,036,854,775,808 and 9,223,372,036,854,775,807
Fixed Decimal	Numeric	A field with a specific width including the number of decimal places
Float	Numeric	A field with up to 7 digits of accuracy between -3.4^38 and 3.4^38
Double	Numeric	A field with up to 15 digits of accuracy between -1.7^308 and 1.7^308
String	String	A field with up to 8192 characters
WString	String	A field with up to 8192 characters that will accept Unicode characters
V_String	String	A field that will adjust the size depending on the length of the strings in it

V_WString	String	A field that will adjust the size depending on the length of the strings in it and will accept Unicode characters
Date	String	A field in the format "yyyy-mm-dd"
Time	String	A field in the format "hh:mm:ss"
DateTime	String	A field in the format "yyyy-mm-dd hh:mm:ss"
Blob	Blob	A field with image or sound files
SpatialObj	Blob	A field with points, lines, polylines, or polygons.

Appendix E- Properties Window



Appendix F – Boolean Expressions

Expression	Meaning
A = B	Expression A equals expression B
A != B	A does not equal B
A > B	A is greater than B
A >= B	A is greater than or equal to B
A < B	A is less than b
A <= B	A is less than or equal to B
A	A is True
NOT A	A is False
A IN B	A is in the set of B things
A NOT IN B	A is not in the set of B things
A OR B	Expression A is True or expression B is True or both expressions are True
A AND B	Expression A is True and expression B is True
A OR (B AND C)	Expression A is True or expression B and expression C are both True or expression A and expression B and C are True
A AND (B OR C)	Expression A is True and expression B, expression C, or expressions B and C are True

Basic Boolean expressions can be combined to create more complex logic by replacing A, B or C with any basic Boolean Expression.

Appendix G – Data Components

yyyy-MM-dd
yyyyMMdd
MM/dd/yy
MM/dd/yyyy
MM-dd-yy
MM-dd-yyyy
Month dd, yyyy
Month, yyyy
Mon dd
dd/MM/yy
dd/MM/yyyy
dd-MM-yy
dd-MM-yyyy
dy., Month dd, yyyy
day, dd Month, yyyy
dd Month, yyyy
yyyy-MM-dd hh:mm:ss
MM/dd/yyyy hh:mm:ss
MM/dd/yy hh:mm:ss
dd/MM/yyyy hh:mm:ss
dd/MM/yy hh:mm:ss

day - Spelled out weekday

dd – 2 digit day of month

dy – Abbreviated day of week

hh – 2 digit hour

mm – 2 digit minute

MM – 2 digit month of year

Mon – Abbreviated month of year

Month – Spelled out month

ss – 2 digit second

yy – 2 digit year

yyyy – 4 digit year

Figure 12-12 – Date Components

Appendix H – Date/Time Units

Keywords
Years
Months
Days
Hours
Minutes
Seconds

Appendix I – RegEx Cheat Sheet

.	Any Character
$	End of a line
()	Marked Group
(?:)	Unmarked Group
*	Repeat the previous thing 0 or more times
+	Repeat the previous thing 1 or more times
[ ]	A set to be used for optional lists of single characters
[^]	A set to be used for optional lists of single characters to be excluded
[[:alpha:]]	Any letter
\	Escape the following symbol
\<	Beginning of a word
\>	End of a word
\d	Digit
\l	Lowercase letter
\n	New line character
\s	Space
\t	Tab character
\u	Uppercase letter
\w	Word characters
\W	Non-word characters
^	Beginning of a new line or "not" depending on context
|	Or

Appendix J – Action Tool Sets

These tools do not receive connections from an Action tool.

Action	Date	Folder Browse	Numeric Up Down
Checkmark	Drop Down	List Box	Radio Button
Comment	Error Message	Macro Input	Text Box
Condition	Explorer Box	Macro Output	Tree
Control Parameter	File Browse	Map	

All other tools have at least these basic methods

Delete Tool from Workflow Update Field List Update Raw XML with Formula Update Select with Reverse Field Map Update Value Update Value with Formula

These tools only have the basic methods.

AB Analysis	Data Stream In	Join In-DB	Report: Comparison
AB Controls	Data Stream Out	Join Multiple	Report: Detail
AB Treatments	DataSift Connector	JSON Build	Report: Rank

AB Trend	Date Filter	JSON Parse	Run Command
Allocate Append	Date Time Now	K-Centroids Cluster Analysis	Running Total
Allocate Input	DateTime	K-Centroids Diagnostics	Sample
Allocate Metainfo	Decision Tree	Layout	Sample In-DB
Allocate Report	Detail Fields	Lift Chart	Scatterplot
Amazon S3 Download	Detour End	Linear Regression	Score
Amazon S3 Upload	Distance	Logistic Regression	Select In-DB
API Output	Distribution Analysis	Make Columns	Select Records
Append Cluster	Dynamic Input	Make Grid	Smooth
Append Fields	Dynamic Rename	Make Group	Spatial Info
ARIMA	Dynamic Replace	Map Legend Builder	Spatial Match
Arrange	Dynamic Select	Map Legend Splitter	Spatial Process
Association Analysis	Email	Marketo Append	Spearman Correlation
Auto Field	ETS	Marketo Input	Spline Model
Base 64 Encoder	Field Info	Marketo Output	Stepwise
Behavior Metainfo	Field Summary	MB Inspect	Street Geocoder

Blob Convert	Filter	MB Rules	Summarize
Blob Output	Filter In-DB	Message	Summarize In-DB
Block Until Done	Find Nearest	Multi-Field Binning	Support Vector Machine
Boosted Model	Find Nearest Neighbors	Multi-Field Formula	Table
Browse	Find Replace	Multi-Row Formula	Tableau Workbook Macro
Browse Data In-DB	Footer	Naïve Bayes Classifier	Test
Buffer	Forest Model	Nested Test	Test of Means
Business Match (US)	Formula	Neural Network	Text To Columns
Calgary Input	Formula In-DB	Non Overlap Drivetime	Throttle
Calgary Join	Foursquare Search	Overlay	Tile
Calgary Loader	Frequency Table	Oversample Field	Transpose
Canada Geocoder	Fuzzy Match	Parse Address	TS Compare
CASS	Gamma Regression	Pearson Correlation	TS Covariate Forecast
Charting	Generalize	Plot of Means	TS Filler
Cluster Code	Generate Rows	Poly-Build	TS Forecast
Compare Behavior	Gnip Search	Poly-Split	TS Plot

Connect In-DB	Google Analytics	Principal Components	Twitter Search
ConsumerView Matching	HDFS Input	Profile Input	Union
Contingency Table	HDFS Output	Profile Output	Union In-DB
Count Records	Header	R	Unique
Count Regression	Heat Map	Random % Sample	US Geocoder
Create Points	Heat Plot	Record ID	US Zip 9 Coder
Create Profile	Histogram	RegEx	Violin Plot
Create Samples	Household File Matching	Render	Weighted Average
Cross Count	Image	Report Map	Write In-DB
Cross Count Append	Imputation	Report Text	XML Parse
Cross Tab	Join		

The following tools have the basic tools plus

Update Password

Download	MongoDB Input	Salesforce Input	SharePoint List Input
	MongoDB Output	Salesforce Output	SharePoint List Output

The following tools have the basic tools plus

Update Input Data Tool

Blob Input	Input Data

The following tools have the basic tools plus unique methods

Detour	Update Detour Direction from Condition Update Detour Direction from Question
Directory	Update Directory with Folder Browse
Map Input	Update Map Input with Map Question
Output Data	Update Output Data Tool
Select	Update Select with multi-select ListBox
Sort	Update Sort Order
Text Input	Update Cell
Tool Container	Enable/Disable Container From Condition Enable/Disable Container From Question
Trade Area	Update Trade Area Radius

Appendix K – Directory Tool Data

FullPath	The direct path to open the file.
Directory	The direct path to the folder the file is in.
FileName	The actual name and extension of the file.
ShortFileName	A short file name provided for compatibility with legacy software.
CreationTime	This indicates the first time the file was created.
LastAccessTime	This indicates the last time the file was opened.
LastWriteTime	This indicates the last time the file was saved.
Size	This is the total size of the file in bytes.
AttributeArchive	This indicates if the file is archived.
AttributeCompressed	This indicates if the file is compressed.
AttributeEncrypted	This indicates if the file is encrypted.
AttributeHidden	This indicates if the file is hidden.
AttributeNormal	This indicates if the file has normal attributes.
AttributeOffline	This indicates if the data is unavailable.
AttributeReadonly	This indicates if the file is read-only.
AttributeResparsePoint	This indicates that the file is associated with a reparse point.
AttributeSparseFile	This indicates that the file is sparse.
AttributeSystem	This indicates that the file is a system file.
AttributeTemporary	This indicates that the file is a temporary file.

Credited Original Data Sources

The Games

- Medals:
  - Access through Tableau Public and assumed to be in the public domain https://public.tableau.com/profile/a.m.5517#!/vizhome/OlympicMedals2000-2012_0/Dashboard2 published by A.M.
- Nobel Laureates:
  - Access through Tableau Public and assumed to be in the public domain: http://tableausoftware.co.nz/public/gallery/nobelprizes published by Ross Perez
- GDP per Capita:
  - Accessed through GitHub https://github.com/cschin/ipython_d3_mashup/blob/master/ipython_13_vis_example/gm_data/GDPpercapitaconstant2000US.csv published by **Jason Chin**

Unisex Baby Names

- Unisex Names:
  - Access through Tableau Public and assumed to be in the public domain http://tableausoftware.co.nz/public/gallery/unisex-baby-names published by Steve Ruble

The Direct Approach

- Created by Author
- Special Thanks to http://watchout4snakes.com/ for use of their random word generator in creating the names of the folders in the exercise Where, Oh Where Have The Three Files Gone?

Cultural Musings

- GDP per Capita:
  - List of U.S. states by GDP per capita. (2015, March 28). In *Wikipedia, The Free Encyclopedia*. Retrieved 00:01, May 12, 2015, fromhttp://en.wikipedia.org/w/index.php?title=List_of_U.S._states_by_GDP_per_capita&oldid=653869588
- Museum Data:
  - Data retrieved from Data.Gov (http://data.gov) Direct Link: http://catalog.data.gov/dataset/museums-universe-data-file-mudf-fy-2014-3rd-quarter/resource/d4b36e0f-1d9d-4c2a-8882-138c15763717
- State Map:
  - Created by Author

Sport Report

- Major US Sport Teams:
  - The author created this data using Wikipedia pages and tools
    - Primary Wikipedia Article

- List of professional sports teams in the United States and Canada. (2015, March 8). In *Wikipedia, The Free Encyclopedia*. Retrieved 14:34, January 7, 2015, from http://en.wikipedia.org/w/index.php?title=List_of_professional_sports_teams_in_the_United_States_and_Canada&oldid=650502141

- The Geocoded locations of each of the sites were accessed by using the coordinates link embedded in their web pages with utilizes a Wikimedia Tool called GeoHack which can be accessed directly at http://tools.wmflabs.org/geohack/

- Turner Field. (2015, March 22). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:02, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Turner_Field&oldid=653079052
- Marlins Park. (2015, March 2). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:03, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Marlins_Park&oldid=649550406
- Citi Field. (2015, March 25). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:03, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Citi_Field&oldid=653482419
- Citizens Bank Park. (2015, March 1). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:03, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Citizens_Bank_Park&oldid=649435153
- Nationals Park. (2015, April 7). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:03, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Nationals_Park&oldid=655186324
- Wrigley Field. (2015, April 7). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:04, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Wrigley_Field&oldid=655275853
- Great American Ball Park. (2015, April 7). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:04, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Great_American_Ball_Park&oldid=655285631
- Miller Park (Milwaukee). (2015, April 6). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:04, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Miller_Park_(Milwaukee)&oldid=655121188
- PNC Park. (2015, April 3). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:05, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=PNC_Park&oldid=654838712
- Busch Stadium. (2015, March 2). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:05, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Busch_Stadium&oldid=649550165
- Chase Field. (2015, April 5). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:05, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Chase_Field&oldid=655057408
- Coors Field. (2015, April 1). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:09, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Coors_Field&oldid=654538824
- Dodger Stadium. (2015, April 4). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:09, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Dodger_Stadium&oldid=654869845
- Petco Park. (2015, April 7). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:10, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Petco_Park&oldid=654538750
- AT&T Park. (2015, April 1). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:10, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=AT%26T_Park&oldid=654538692
- Oriole Park at Camden Yards. (2015, March 18). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:11, April 7, 2015,

from http://en.wikipedia.org/w/index.php?title=Oriole_Park_at_Camden_Yards&oldid=651898983

- Fenway Park. (2015, March 25). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:11, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Fenway_Park&oldid=653504103
- Yankee Stadium. (2015, April 4). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:11, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Yankee_Stadium&oldid=654863343
- Tropicana Field. (2015, April 6). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:11, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Tropicana_Field&oldid=655160166
- Rogers Centre. (2015, April 4). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:12, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Rogers_Centre&oldid=654913445
- U.S. Cellular Field. (2015, March 4). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:12, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=U.S._Cellular_Field&oldid=649856024
- Progressive Field. (2015, March 25). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:12, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Progressive_Field&oldid=653499953
- Comerica Park. (2015, March 17). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:13, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Comerica_Park&oldid=651832995
- Kauffman Stadium. (2015, March 8). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:13, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Kauffman_Stadium&oldid=650377832
- Target Field. (2015, February 19). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:13, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Target_Field&oldid=647933510
- Minute Maid Park. (2015, April 7). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:13, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Minute_Maid_Park&oldid=655303381
- Angel Stadium of Anaheim. (2015, April 1). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:13, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Angel_Stadium_of_Anaheim&oldid=654460841
- O.co Coliseum. (2015, March 26). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:13, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=O.co_Coliseum&oldid=653612043
- Safeco Field. (2015, April 6). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:14, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Safeco_Field&oldid=655148228
- Globe Life Park in Arlington. (2015, March 22). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:14, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Globe_Life_Park_in_Arlington&oldid=652959003
- TD Garden. (2015, March 25). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:14, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=TD_Garden&oldid=653403579
- Barclays Center. (2015, April 2). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:14, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Barclays_Center&oldid=654631943
- Madison Square Garden. (2015, April 3). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:14, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Madison_Square_Garden&oldid=654828742
- Wells Fargo Center (Philadelphia). (2015, March 3). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:15, April 7, 2015, fromhttp://en.wikipedia.org/w/index.php?title=Wells_Fargo_Center_(Philadelphia)&oldid=649738895
- Air Canada Centre. (2015, April 6). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:15, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Air_Canada_Centre&oldid=655130412

- United Center. (2015, April 7). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:15, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=United_Center&oldid=655268263
- Quicken Loans Arena. (2015, April 7). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:15, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Quicken_Loans_Arena&oldid=655272955
- The Palace of Auburn Hills. (2015, February 22). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:16, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=The_Palace_of_Auburn_Hills&oldid=648272487
- Bankers Life Fieldhouse. (2014, December 29). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:16, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Bankers_Life_Fieldhouse&oldid=640160898
- BMO Harris Bradley Center. (2015, February 4). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:16, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=BMO_Harris_Bradley_Center&oldid=645646883
- Philips Arena. (2015, April 1). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:16, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Philips_Arena&oldid=654538478
- Time Warner Cable Arena. (2015, March 24). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:16, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Time_Warner_Cable_Arena&oldid=653373673
- American Airlines Arena. (2015, February 10). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:16, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=American_Airlines_Arena&oldid=646426983
- Amway Center. (2015, April 7). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:17, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Amway_Center&oldid=655268871
- Verizon Center. (2015, March 14). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:17, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Verizon_Center&oldid=651299892
- Pepsi Center. (2015, March 26). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:17, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Pepsi_Center&oldid=653560944
- Target Center. (2015, April 5). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:17, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Target_Center&oldid=655045980
- Chesapeake Energy Arena. (2015, March 5). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:18, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Chesapeake_Energy_Arena&oldid=650053351
- Moda Center. (2015, March 21). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:18, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Moda_Center&oldid=652902770
- EnergySolutions Arena. (2015, March 30). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:18, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=EnergySolutions_Arena&oldid=654209905
- Oracle Arena. (2015, February 26). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:18, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Oracle_Arena&oldid=648884284
- Staples Center. (2015, April 4). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:18, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Staples_Center&oldid=654943633

- Talking Stick Resort Arena. (2015, April 3). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:19, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Talking_Stick_Resort_Arena&oldid=654788366
- Sleep Train Arena. (2015, February 20). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:19, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Sleep_Train_Arena&oldid=648040550
- AT&T Center. (2015, March 30). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:40, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=AT%26T_Center&oldid=654184289
- Smoothie King Center. (2015, March 25). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:40, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Smoothie_King_Center&oldid=653411690
- FedExForum. (2015, March 3). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:40, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=FedExForum&oldid=649655797
- Toyota Center. (2015, February 28). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:41, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Toyota_Center&oldid=649249468
- American Airlines Center. (2015, March 29). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:41, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=American_Airlines_Center&oldid=654000621
- Sleep Train Arena. (2015, February 20). In *Wikipedia, The Free Encyclopedia*. Retrieved 15:19, April 7, 2015, from http://en.wikipedia.org/w/index.php?title=Sleep_Train_Arena&oldid=648040550
- Philips Arena. (2015, April 1). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:43, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Philips_Arena&oldid=654538478
- Allstate Arena. (2015, April 3). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:43, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Allstate_Arena&oldid=654835689
- Mohegan Sun Arena. (2015, March 6). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:43, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Mohegan_Sun_Arena&oldid=650179511
- Bankers Life Fieldhouse. (2014, December 29). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:43, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Bankers_Life_Fieldhouse&oldid=640160898
- Madison Square Garden. (2015, April 3). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:43, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Madison_Square_Garden&oldid=654828742
- Verizon Center. (2015, March 14). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:43, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Verizon_Center&oldid=651299892
- Staples Center. (2015, April 4). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:43, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Staples_Center&oldid=654943633
- Target Center. (2015, April 5). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:44, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Target_Center&oldid=655045980
- Talking Stick Resort Arena. (2015, April 3). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:44, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Talking_Stick_Resort_Arena&oldid=654788366

- AT&T Center. (2015, March 30). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:44, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=AT%26T_Center&oldid=654184289
- KeyArena. (2015, March 24). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:44, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=KeyArena&oldid=653237504
- BOK Center. (2015, March 31). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:44, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=BOK_Center&oldid=654259841
- Santander Arena. (2015, February 23). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:44, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Santander_Arena&oldid=648485678
- Odeum Expo Center. (2015, February 21). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:44, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Odeum_Expo_Center&oldid=648219741
- The Gardens Ice House. (2014, December 2). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:44, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=The_Gardens_Ice_House&oldid=636303525
- Tiger Arena. (2015, March 25). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:44, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Tiger_Arena&oldid=653415784
- Dow Event Center. (2015, March 30). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:44, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Dow_Event_Center&oldid=654204684
- York City Ice Arena. (2014, October 10). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:44, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=York_City_Ice_Arena&oldid=629018943
- Talking Stick Resort Arena. (2015, April 3). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:44, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Talking_Stick_Resort_Arena&oldid=654788366
- Honda Center. (2015, April 3). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:44, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Honda_Center&oldid=654822150
- Alamodome. (2015, March 30). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:44, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Alamodome&oldid=654207806
- Moda Center. (2015, March 21). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:44, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Moda_Center&oldid=652902770
- SAP Center at San Jose. (2015, April 2). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:44, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=SAP_Center_at_San_Jose&oldid=654649938
- Spokane Veterans Memorial Arena. (2015, March 29). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:44, April 8, 2015, fromhttp://en.wikipedia.org/w/index.php?title=Spokane_Veterans_Memorial_Arena&oldid=654081587
- Jacksonville Veterans Memorial Arena. (2015, March 21). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:44, April 8, 2015, fromhttp://en.wikipedia.org/w/index.php?title=Jacksonville_Veterans_Memorial_Arena&oldid=652941155
- CFE Arena. (2015, February 12). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:44, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=CFE_Arena&oldid=646752542
- Amalie Arena. (2015, February 12). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:44, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Amalie_Arena&oldid=646850271

- Quicken Loans Arena. (2015, April 7). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:44, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Quicken_Loans_Arena&oldid=655272955
- Smoothie King Center. (2015, March 25). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:45, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Smoothie_King_Center&oldid=653411690
- Wells Fargo Center (Philadelphia). (2015, March 3). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:45, April 8, 2015, fromhttp://en.wikipedia.org/w/index.php?title=Wells_Fargo_Center_(Philadelphia)&oldid=649738895
- Tim Hortons Field. (2015, March 17). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:45, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Tim_Hortons_Field&oldid=651817295
- Percival Molson Memorial Stadium. (2015, January 16). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:45, April 8, 2015, fromhttp://en.wikipedia.org/w/index.php?title=Percival_Molson_Memorial_Stadium&oldid=642808624
- TD Place Stadium. (2015, February 25). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:45, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=TD_Place_Stadium&oldid=648816970
- Rogers Centre. (2015, April 4). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:45, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Rogers_Centre&oldid=654913445
- BC Place. (2015, March 26). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:45, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=BC_Place&oldid=653675532
- McMahon Stadium. (2015, February 28). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:45, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=McMahon_Stadium&ol
  Gillette Stadium. (2015, April 2). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:45, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Gillette_Stadium&oldid=654585813did=649207077
- Commonwealth Stadium (Edmonton). (2015, March 4). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:45, April 8, 2015, fromhttp://en.wikipedia.org/w/index.php?title=Commonwealth_Stadium_(Edmonton)&oldid=649791012
- Mosaic Stadium at Taylor Field. (2015, February 15). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:45, April 8, 2015, fromhttp://en.wikipedia.org/w/index.php?title=Mosaic_Stadium_at_Taylor_Field&oldid=647219959
- Investors Group Field. (2015, April 7). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:45, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Investors_Group_Field&oldid=655386188
- Ralph Wilson Stadium. (2015, April 5). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:45, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Ralph_Wilson_Stadium&oldid=655091171
- Sun Life Stadium. (2015, March 21). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:45, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Sun_Life_Stadium&oldid=652918431

- MetLife Stadium. (2015, April 8). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:45, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=MetLife_Stadium&oldid=655501155
- M&T Bank Stadium. (2014, December 16). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:45, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=M%26T_Bank_Stadium&oldid=638418974
- Paul Brown Stadium. (2014, December 7). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:46, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Paul_Brown_Stadium&oldid=637052857
- FirstEnergy Stadium (Cleveland). (2015, April 1). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:46, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=FirstEnergy_Stadium_(Cleveland)&oldid=654531854
- Heinz Field. (2015, March 25). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:47, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Heinz_Field&oldid=653482405
- NRG Stadium. (2015, April 7). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:47, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=NRG_Stadium&oldid=655407096
- Lucas Oil Stadium. (2015, April 7). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:47, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Lucas_Oil_Stadium&oldid=655376758
- EverBank Field. (2015, March 23). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:47, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=EverBank_Field&oldid=653161581
- LP Field. (2014, November 21). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:47, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=LP_Field&oldid=634894166
- Sports Authority Field at Mile High. (2015, March 23). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:48, April 8, 2015, fromhttp://en.wikipedia.org/w/index.php?title=Sports_Authority_Field_at_Mile_High&oldid=653126660
- Arrowhead Stadium. (2015, March 29). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:48, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Arrowhead_Stadium&oldid=654006833
- O.co Coliseum. (2015, March 26). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:48, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=O.co_Coliseum&oldid=653612043
- Qualcomm Stadium. (2015, March 27). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:48, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Qualcomm_Stadium&oldid=653741585
- AT&T Stadium. (2015, April 3). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:48, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=AT%26T_Stadium&oldid=654731734
- MetLife Stadium. (2015, April 8). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:48, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=MetLife_Stadium&oldid=655501155
- Lincoln Financial Field. (2015, April 6). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:48, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Lincoln_Financial_Field&oldid=655256338
- FedExField. (2015, April 1). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:48, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=FedExField&oldid=654429456
- Soldier Field. (2015, March 12). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:48, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Soldier_Field&oldid=651010515
- Ford Field. (2015, March 28). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:48, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Ford_Field&oldid=653890809

- Lambeau Field. (2015, March 10). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:48, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Lambeau_Field&oldid=650772069
- TCF Bank Stadium. (2015, March 3). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:48, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=TCF_Bank_Stadium&oldid=649686714
- Georgia Dome. (2015, March 24). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:48, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Georgia_Dome&oldid=653308324
- Bank of America Stadium. (2015, March 24). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:48, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Bank_of_America_Stadium&oldid=653374205
- Mercedes-Benz Superdome. (2015, March 30). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:48, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Mercedes-Benz_Superdome&oldid=654152346
- Raymond James Stadium. (2015, February 28). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:49, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Raymond_James_Stadium&oldid=649202435
- University of Phoenix Stadium. (2015, March 26). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:49, April 8, 2015, fromhttp://en.wikipedia.org/w/index.php?title=University_of_Phoenix_Stadium&oldid=653559579
- Edward Jones Dome. (2015, March 30). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:49, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Edward_Jones_Dome&oldid=654167126
- Levi's Stadium. (2015, April 6). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:49, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=Levi%27s_Stadium&oldid=655210318
- CenturyLink Field. (2015, March 20). In *Wikipedia, The Free Encyclopedia*. Retrieved 13:49, April 8, 2015, from http://en.wikipedia.org/w/index.php?title=CenturyLink_Field&oldid=652787982
- Toyota Park. (2015, March 30). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:30, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Toyota_Park&oldid=654161546
- Mapfre Stadium. (2015, March 19). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:31, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Mapfre_Stadium&oldid=652025679
- Robert F. Kennedy Memorial Stadium. (2015, March 20). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:31, April 9, 2015, fromhttp://en.wikipedia.org/w/index.php?title=Robert_F._Kennedy_Memorial_Stadium&oldid=652754664
- Saputo Stadium. (2015, March 23). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:31, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Saputo_Stadium&oldid=653113798
- Gillette Stadium. (2015, April 2). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:31, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Gillette_Stadium&oldid=654585813
- Yankee Stadium. (2015, April 8). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:31, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Yankee_Stadium&oldid=655513009
- Red Bull Arena (New Jersey). (2015, March 23). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:31, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Red_Bull_Arena_(New_Jersey)&oldid=653149464

- Orlando Citrus Bowl Stadium. (2015, March 21). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:31, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Orlando_Citrus_Bowl_Stadium&oldid=652829636
- PPL Park. (2015, March 18). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:31, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=PPL_Park&oldid=651972956
- BMO Field. (2015, March 24). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:31, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=BMO_Field&oldid=653369346
- Dick's Sporting Goods Park. (2015, April 8). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:31, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Dick%27s_Sporting_Goods_Park&oldid=655566313
- Toyota Stadium (Texas). (2014, December 16). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:31, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Toyota_Stadium_(Texas)&oldid=638418651
- BBVA Compass Stadium. (2015, April 6). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:31, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=BBVA_Compass_Stadium&oldid=655235636
- StubHub Center. (2015, March 12). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:31, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=StubHub_Center&oldid=651093792
- Providence Park. (2015, April 3). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:32, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Providence_Park&oldid=654711739
- Rio Tinto Stadium. (2015, March 31). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:32, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Rio_Tinto_Stadium&oldid=654304261
- Avaya Stadium. (2015, April 6). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:32, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Avaya_Stadium&oldid=655230355
- CenturyLink Field. (2015, March 20). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:32, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=CenturyLink_Field&oldid=652787982
- Sporting Park. (2015, March 13). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:32, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Sporting_Park&oldid=651202621
- BC Place. (2015, March 26). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:32, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=BC_Place&oldid=653675532
- Harvard Stadium. (2015, April 1). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:32, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Harvard_Stadium&oldid=654508027
- Village of Lisle-Benedictine University Sports Complex. (2014, September 6). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:32, April 9, 2015, fromhttp://en.wikipedia.org/w/index.php?title=Village_of_Lisle-Benedictine_University_Sports_Complex&oldid=624371085
- BBVA Compass Stadium. (2015, April 6). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:33, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=BBVA_Compass_Stadium&oldid=655235636
- Providence Park. (2015, April 3). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:33, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Providence_Park&oldid=654711739

- Memorial Stadium (Seattle). (2015, March 22). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:33, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Memorial_Stadium_(Seattle)&oldid=652979450
- Yurcak Field. (2014, October 11). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:33, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Yurcak_Field&oldid=629218625
- Maryland SoccerPlex. (2014, October 17). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:33, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Maryland_SoccerPlex&oldid=629989133
- Sahlen's Stadium. (2015, March 31). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:33, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Sahlen%27s_Stadium&oldid=654403301
- Atlanta Silverbacks Park. (2014, November 2). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:33, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Atlanta_Silverbacks_Park&oldid=632159843
- WakeMed Soccer Park. (2015, February 18). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:33, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=WakeMed_Soccer_Park&oldid=647633122
- Clarke Stadium. (2015, April 2). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:33, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Clarke_Stadium&oldid=654678732
- Lockhart Stadium. (2014, November 4). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:33, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Lockhart_Stadium&oldid=632454890
- Baseball Grounds of Jacksonville. (2015, March 2). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:33, April 9, 2015, fromhttp://en.wikipedia.org/w/index.php?title=Baseball_Grounds_of_Jacksonville&oldid=649550076
- IU Michael A. Carroll Track & Soccer Stadium. (2014, December 5). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:33, April 9, 2015, fromhttp://en.wikipedia.org/w/index.php?title=IU_Michael_A._Carroll_Track_%26_Soccer_Stadium&oldid=636779965
- National Sports Center. (2014, December 2). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:34, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=National_Sports_Center&oldid=636289847
- James M. Shuart Stadium. (2015, January 2). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:34, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=James_M._Shuart_Stadium&oldid=640688511
- Toyota Field. (2015, April 1). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:34, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Toyota_Field&oldid=654448149
- Al Lang Stadium. (2015, March 20). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:34, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Al_Lang_Stadium&oldid=652776072
- Blackbaud Stadium. (2015, March 23). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:34, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Blackbaud_Stadium&oldid=653227162
- American Legion Memorial Stadium. (2015, March 30). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:34, April 9, 2015,

fromhttp://en.wikipedia.org/w/index.php?title=American_Legion_Memorial_Stadium&oldid=654180309

- Skyline Sports Complex. (2015, April 4). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:34, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Skyline_Sports_Complex&oldid=654929230
- Louisville Slugger Field. (2015, April 8). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:34, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Louisville_Slugger_Field&oldid=655428425
- Saputo Stadium. (2015, March 23). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:34, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Saputo_Stadium&oldid=653113798
- Highmark Stadium. (2015, April 4). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:34, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Highmark_Stadium&oldid=654950422
- City Stadium (Richmond). (2015, February 19). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:34, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=City_Stadium_(Richmond)&oldid=647801967
- Sahlen's Stadium. (2015, March 31). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:34, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Sahlen%27s_Stadium&oldid=654403301
- St. Louis Soccer Park. (2015, April 3). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:34, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=St._Louis_Soccer_Park&oldid=654795723
- Ontario Soccer Centre. (2015, March 2). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:35, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Ontario_Soccer_Centre&oldid=649486610
- Legion Stadium. (2014, December 16). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:35, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Legion_Stadium&oldid=638361469
- Scottsdale Stadium. (2015, March 19). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:38, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Scottsdale_Stadium&oldid=652123780
- House Park. (2015, March 30). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:38, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=House_Park&oldid=654152211
- Sand Creek Stadium. (2015, March 30). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:38, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Sand_Creek_Stadium&oldid=654238924
- StubHub Center. (2015, March 12). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:38, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=StubHub_Center&oldid=651093792
- Bishop McGuinness Catholic High School (Oklahoma). (2015, April 8). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:38, April 9, 2015, fromhttp://en.wikipedia.org/w/index.php?title=Bishop_McGuinness_Catholic_High_School_(Oklahoma)&oldid=655587468
- Anteater Stadium. (2014, December 16). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:38, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Anteater_Stadium&oldid=638362970
- Merlo Field. (2015, March 30). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:38, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Merlo_Field&oldid=654210299

- Rio Tinto Stadium. (2015, March 31). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:38, April 9, 2015,
  from http://en.wikipedia.org/w/index.php?title=Rio_Tinto_Stadium&oldid=654304261
- Bonney Field. (2015, April 6). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:38, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Bonney_Field&oldid=655241105
- Starfire Sports. (2015, March 22). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:38, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=Starfire_Sports&oldid=653021031
- ONEOK Field. (2015, January 30). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:38, April 9, 2015, from http://en.wikipedia.org/w/index.php?title=ONEOK_Field&oldid=644849140
- Thunderbird Stadium. (2015, April 7). In *Wikipedia, The Free Encyclopedia*. Retrieved 16:39, April 9, 2015,
  from http://en.wikipedia.org/w/index.php?title=Thunderbird_Stadium&oldid=655408047
- https://community.alteryx.com/t5/Alteryx-Knowledge-Base/Emailing-Different-Reports-to-Different-Recipients/ta-p/38517

Expensive Beauty Products

- Created by Author

Applications Wanted

- All Recorded Traffic Tickets:
  - Data provided by the Baltimore Department of Finance on Open Baltimore (https://data.baltimorecity.gov/) Direct Link: https://data.baltimorecity.gov/Transportation/All-Time/ks7u-tsjz (accessed on

Where's the Joe?

- Mermaid Coffee Company: Data modified to exclude "Starbucks" from everywhere in the data.
  - Corporation
  - Alteryx Gallery

Meta-Morphosis

- Created by Author

Let's Do It (In-DB)

- Data was obtained from Adventure Works by using a sample from the below dataset: https://msftdbprodsamples.codeplex.com/downloads/get/165405

Green on the Go

- Data for Supercharger Stations was downloaded from the below link using Alteryx tools: https://www.tesla.com/findus/list/superchargers/United+States
- https://gallery.alteryx.com/#!app/Twitter-Search/574db168a248970bdc36d527

Meet the programmer in You

- http://www.wunderground.com/history/airport/KMDW/2017/01/01/CustomHistory.html
- https://www.wunderground.com/weather/api/d/pricing.html.

Predictive Analytics with Alteryx

- Workflows were used from the below description, section 11 and 15 from the Alteryx help options in the tool. https://help.alteryx.com/9.5/SamplesByDescription.htm
- https://rpubs.com/jbowmer/autoassessment1
- https://rstudio-pubs-static.s3.amazonaws.com/20858_82b28110c8ec44129848a691c4f02246.html

Content:

The Alteryx Help documentation was used extensively in the research for this manual. Alteryx Help can be accessed through the Help Menu, Properties Window, or with the Hot Key F1 in Alteryx or can be accessed directly at http://help.alteryx.com/

Authored and Published by

USEREADY
HELPING YOU SUCCEED WITH DATA


Anirudh Kalva


Anish Bhola


Ankush Laxmeshwar


Damanpreet Kaur


Honey Shahi


Rashmi Shettigar


Rahul Shetty


Rima Upadhye


Sarayu Rakshith B R


Shamalee Thakur


Srinivasan Ramanathan

ABOUT USEREADY

USEReady is a leading provider of business analytics, data management and BI consulting services. We relentlessly strive towards value-driven innovation and digital transformation, and that coupled with our deep domain and technology expertise, rich talent pool, strong focus on operational excellence, and absolute customer fanaticism, enables us to help our customers differentiate in the market. We have proven track record of success in versatile domains such as banking, capital markets, asset management, insurance, healthcare, pharma, media, retail, etc. Consistently winning Tableau's Service Partner of the Year award is a testimony to our performance. Read more – www.useready.com

GET IN TOUCH : www.useready.com | contact@useready.com