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 contact@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 https://resources.useready.com/publications/a-practitioners-guide-to-alteryx-version-2018-1/ 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,

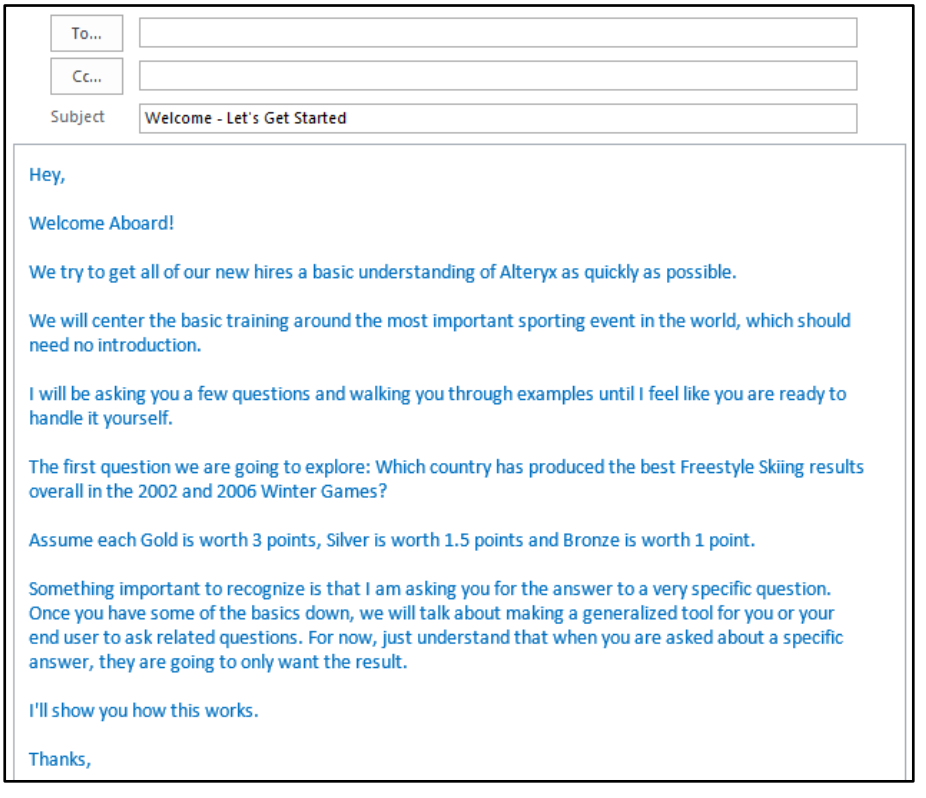
USEReady

Table of Contents

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!



2.1 Tools and Concepts

2.1.1 Input Data

Concept- Input Data

 Figure 2-1-Input Data	**Tool Palette: In/Out**
	Imports data from different data sources. For more details, use the link below. bit.ly/2JNVIIl

2.1.2 Browse

Concept- Viewing Data

 Figure-2-2-Browse	**Tool Palette: In/Out**
	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

2.1.3 Output Data

Concept – Outputting Data

 Figure 2-3-Output Data	**Tool Palette: In/Out**
	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

 Figure-2-4-Comment	**Tool Palette: Documentation**
	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

 **Figure-2-5-Filter**	**Tool Palette: Preparation**
	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

2.1.6 Formula

Concept – Creating Calculations

 **Figure-2-6-Formula**	**Tool Palette: Preparation**
	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

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

 **Figure-2-11-Sort**	**Tool Palette: Preparation**
	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

 **Figure-2-12-Summarize**	**Tool Palette: Transform**
	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

 Figure-2-13-Tool Container	**Tool Palette: Documentation**
	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

2.1.14 Transpose

Concept – Denormalizing Data

 Figure-2-14-Transpose	**Tool Palette: Transform**
	Helps to denormalize the data. For more details, use the link below. bit.ly/2H7Y86S

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

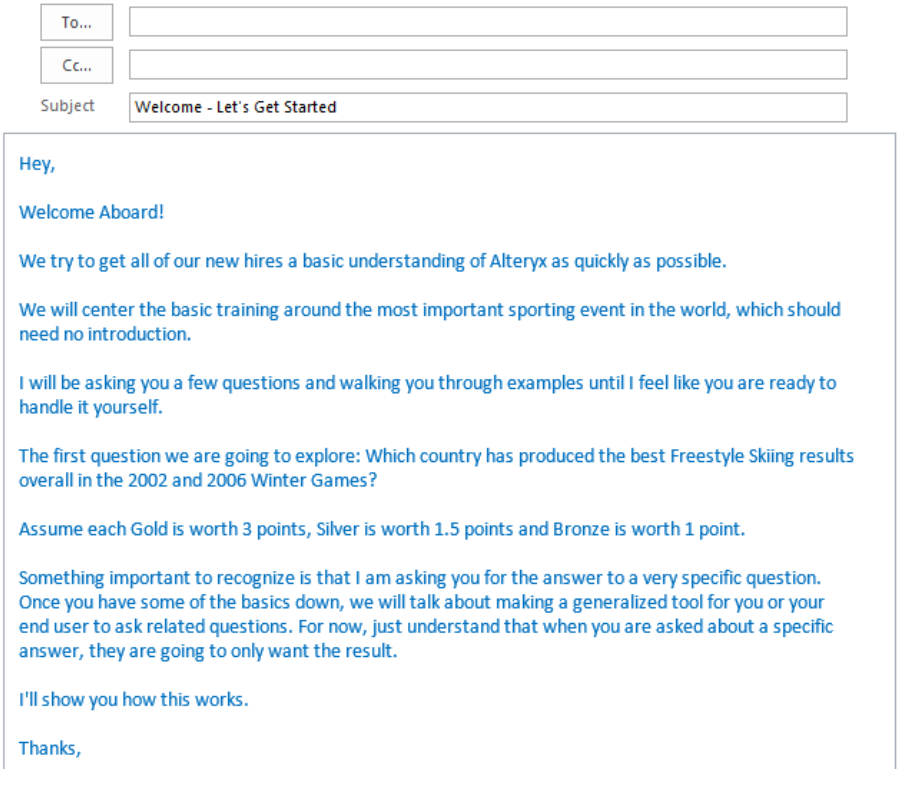
integrations with Tableau Server. Now output directly from an Alteryx workflow into Tableau Hyper is also supported.

- **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



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.

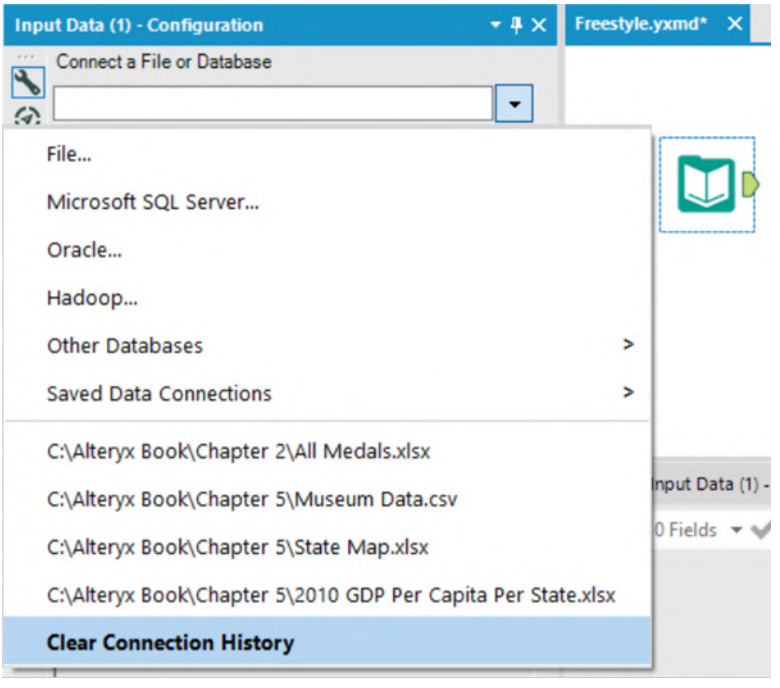


Figure-2-17-Freestyle Skiing Input Data Configuration

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.

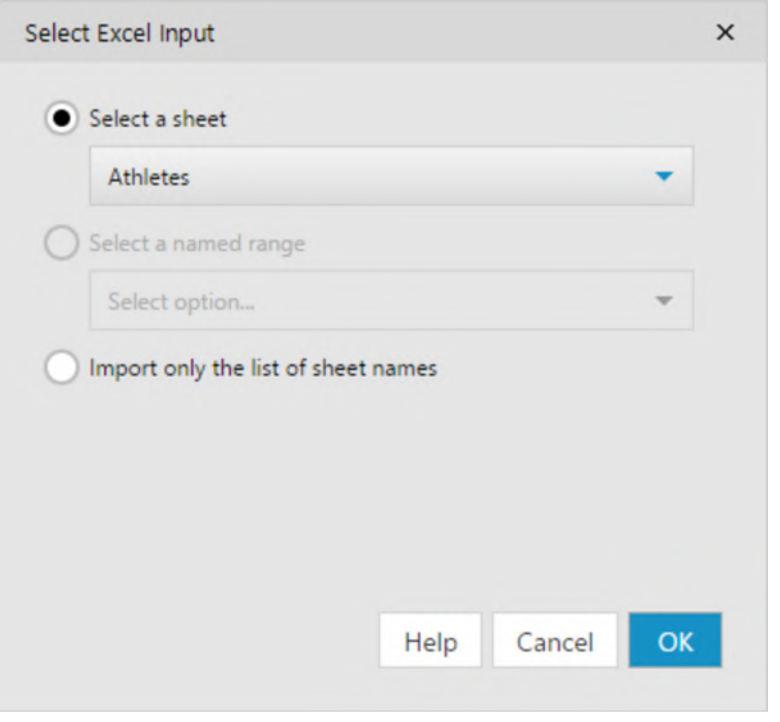


Figure-2-18-Freestyle Skiing – Data Input

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.

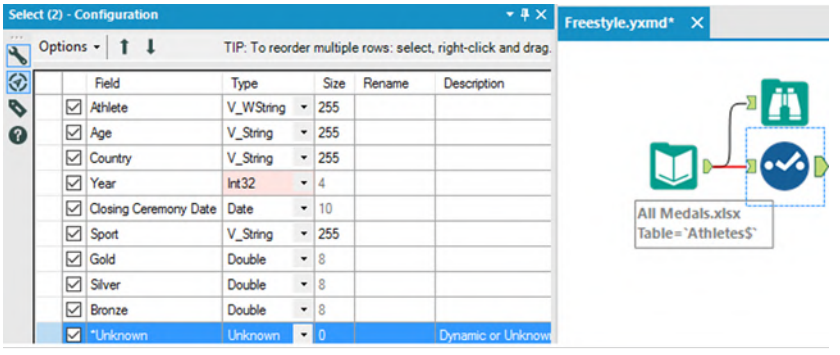


Figure-2-19-Freestyle Skiing
Select Configuration

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.

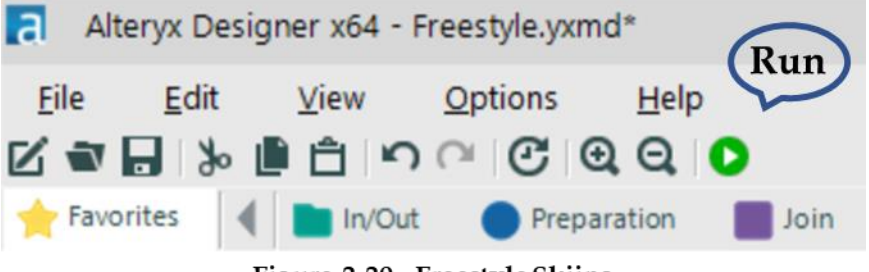


Figure-2-20 – Freestyle Skiing
Click Run to populate Browse

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.

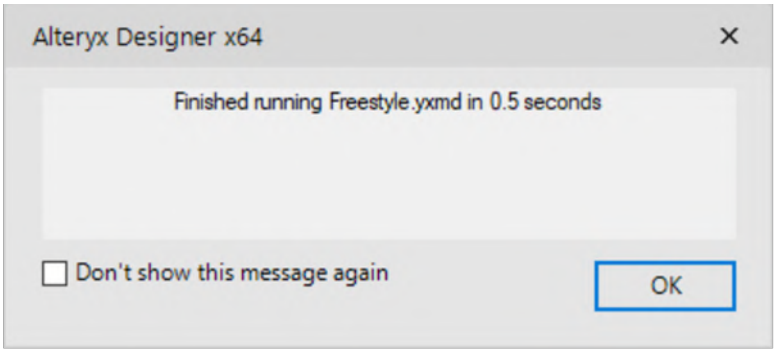
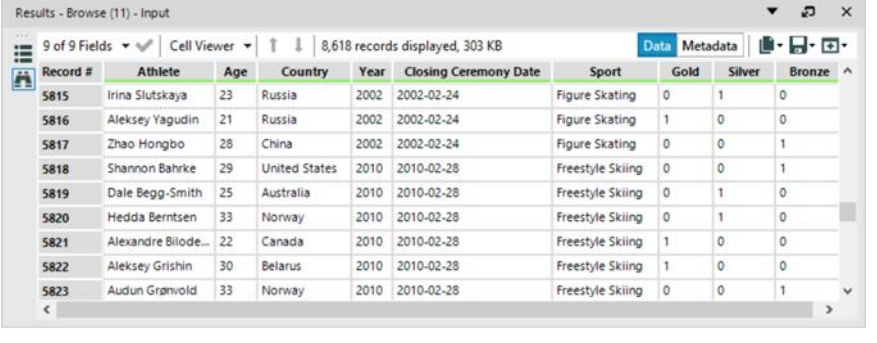


Figure-2-21-Freestyle Skiing – Message after Running Workflow

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.

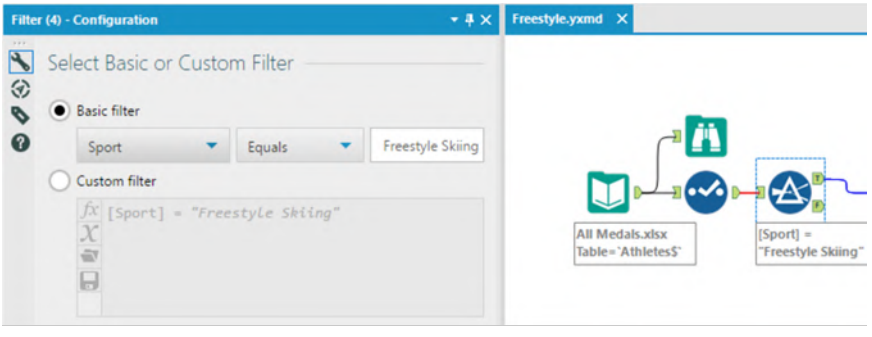


Figure-2-23-Freestyle Skiing – First Filter Configuration

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.

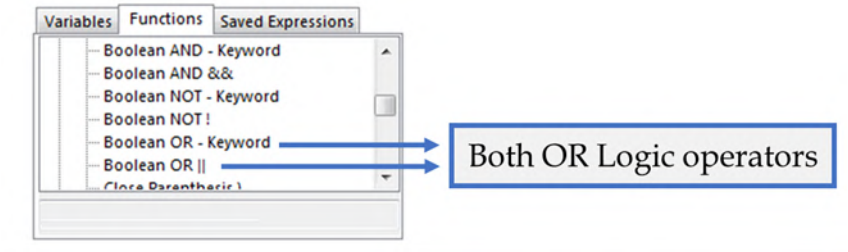


Figure-2-24-Freestyle Skiing
Filter Configuration - Functions

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.

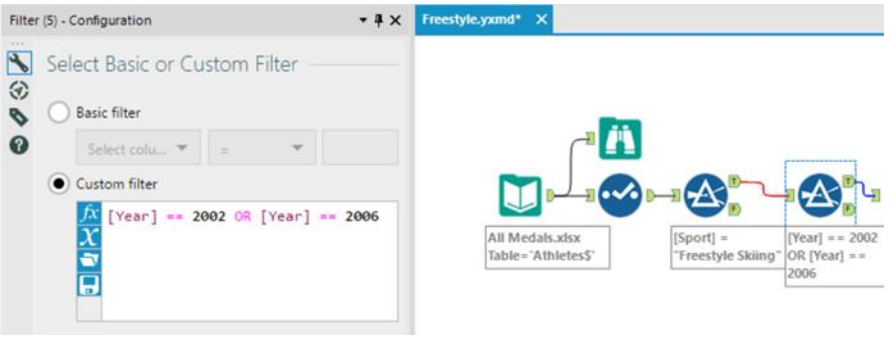


Figure-2-25-Freestyle Skiing Custom Filter Configuration

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.

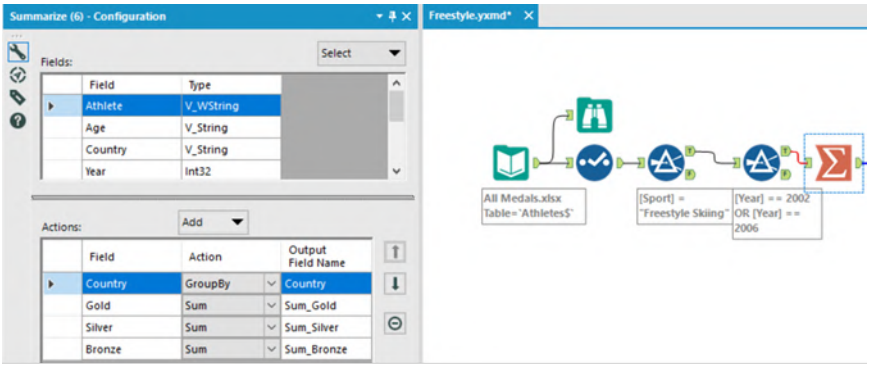
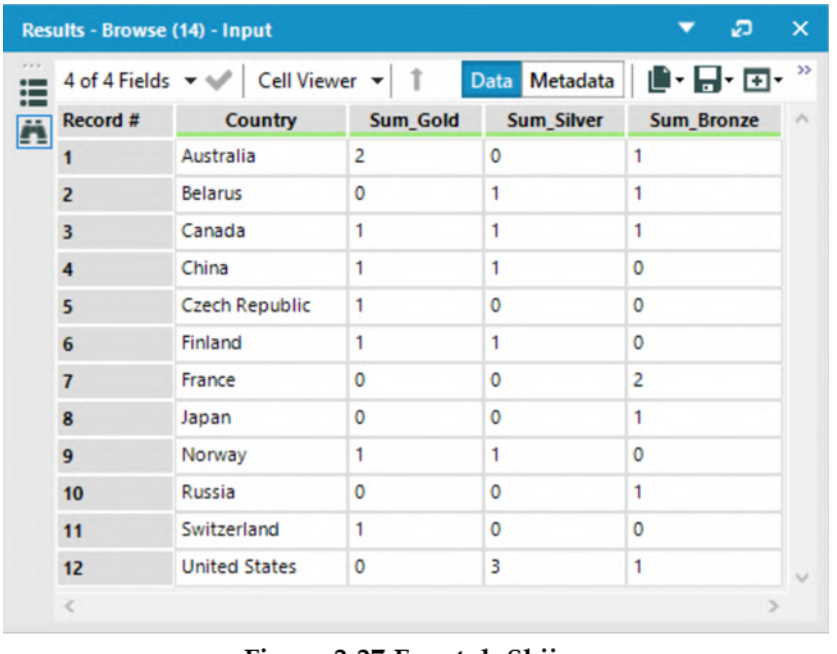


Figure-2-26-Freestyle Skiing Summarize Configuration

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]

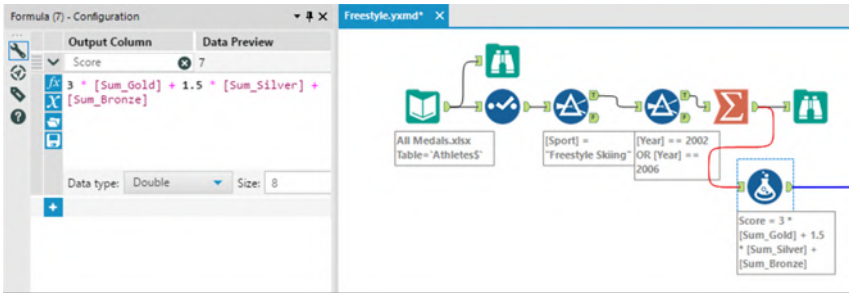


Figure-2-28-Freestyle Skiing - Formula Configuration

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

Results - Browse (13) - Input

5 of 5 Fields | Cell Viewer | Data | Metadata

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.

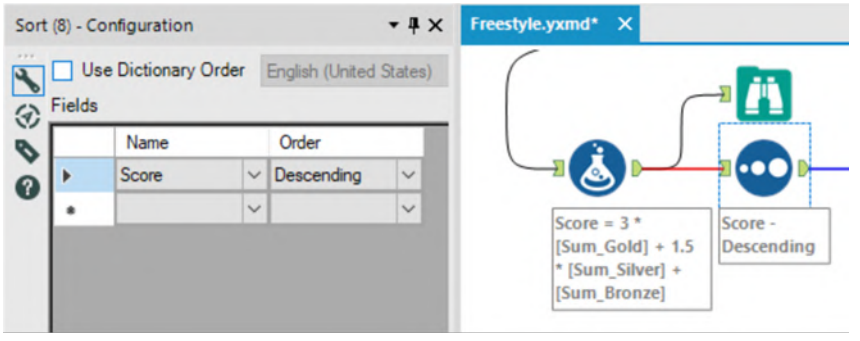


Figure-2-30-Freestyle Skiing
Sort Configuration

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.

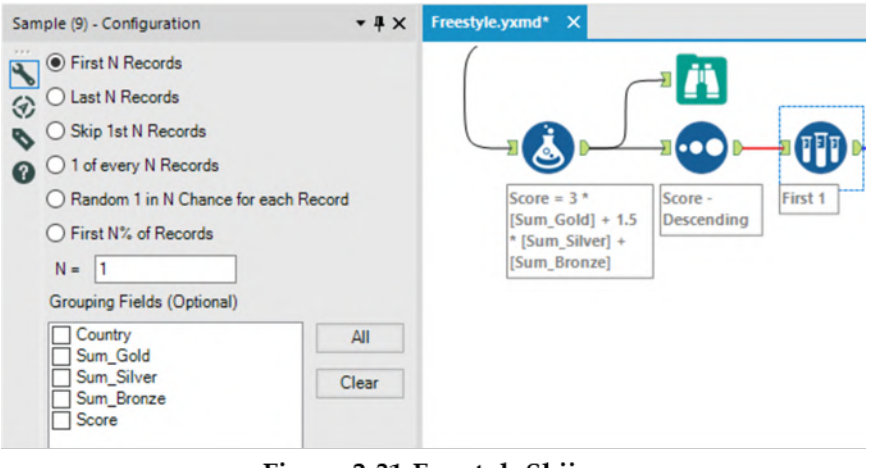


Figure-2-31-Freestyle Skiing
Sample Configuration

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

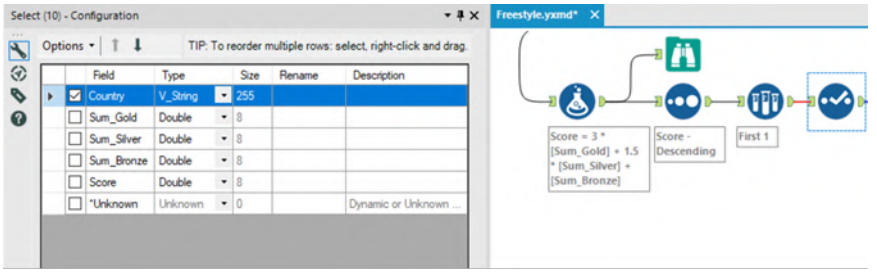


Figure-2-32-Freestyle Skiing
Select Configuration

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

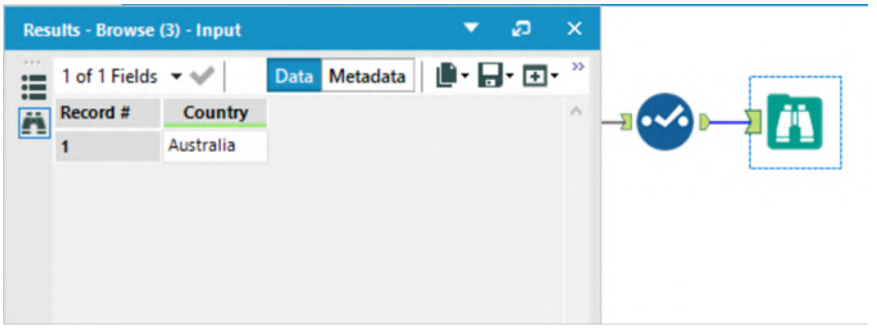


Figure-2-33-Freestyle Skiing
Browse after Select

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:

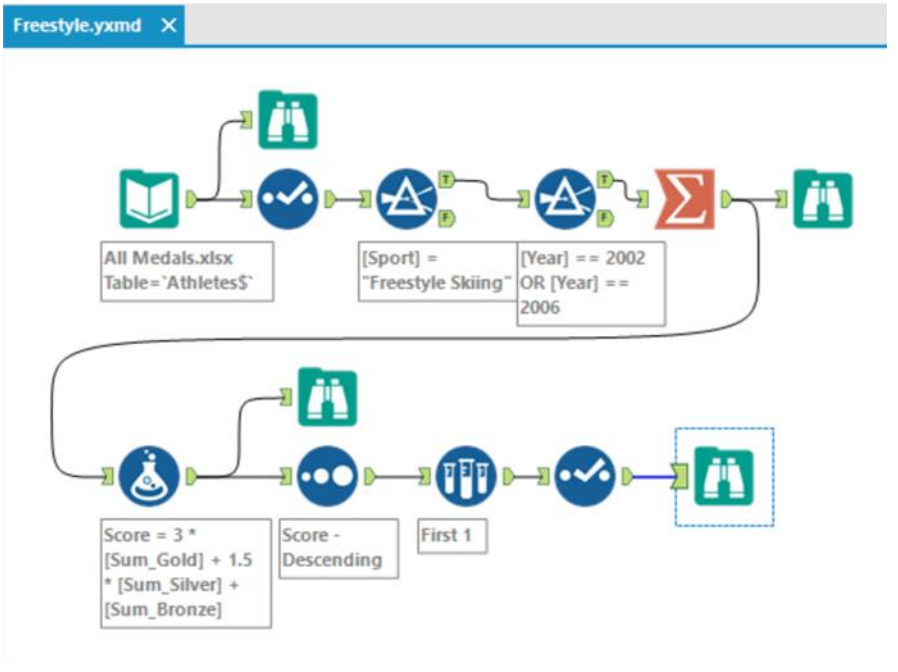
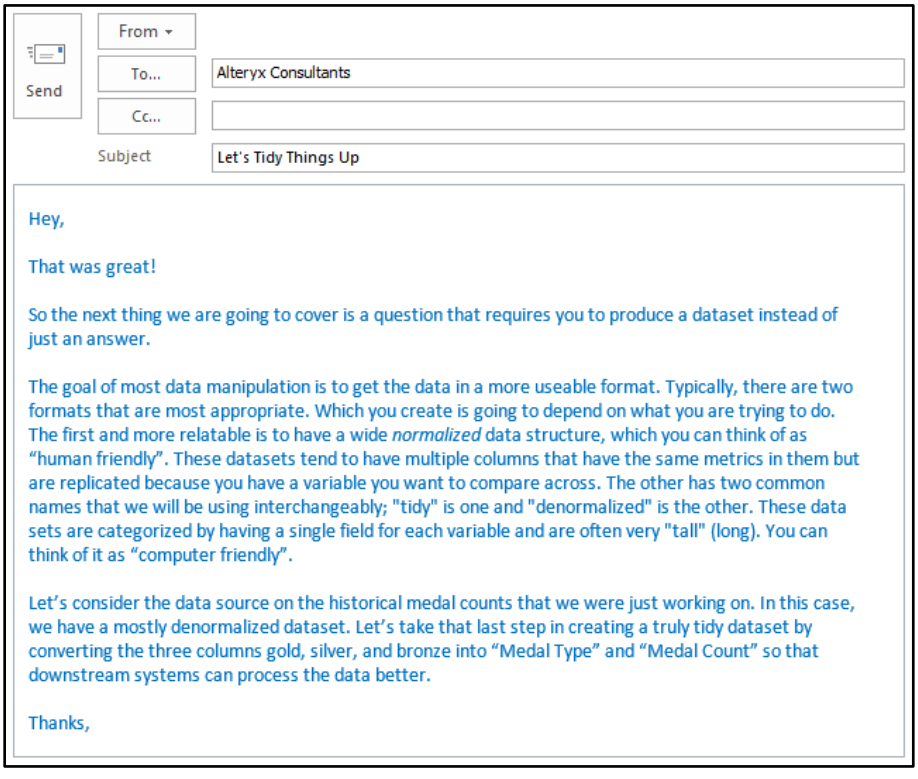


Figure-2-34-Freestyle Skiing Workflow after completion

2.3 Let's Tidy Things Up



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*

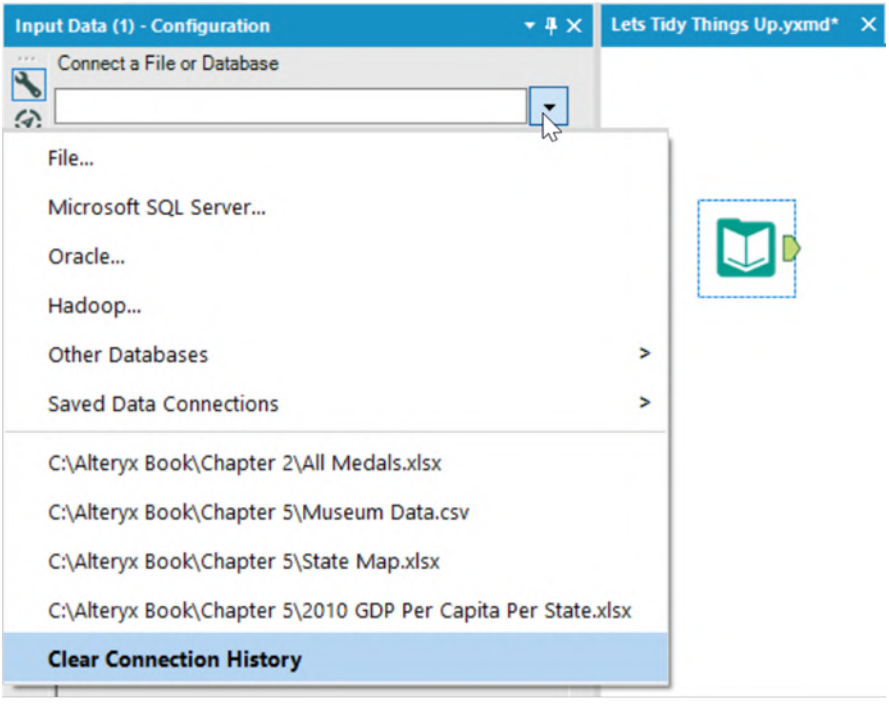


Figure-2-35-Let's Tidy Things Up – Data Input

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.

Results - Browse (3) - Input

9 of 9 Fields | Cell Viewer | 8,618 records displayed, 303 KB | Data | Metadata

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.

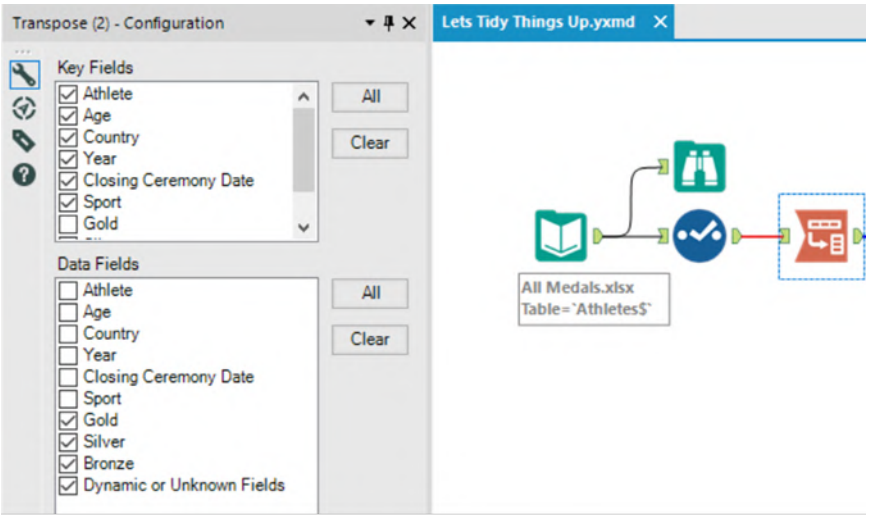
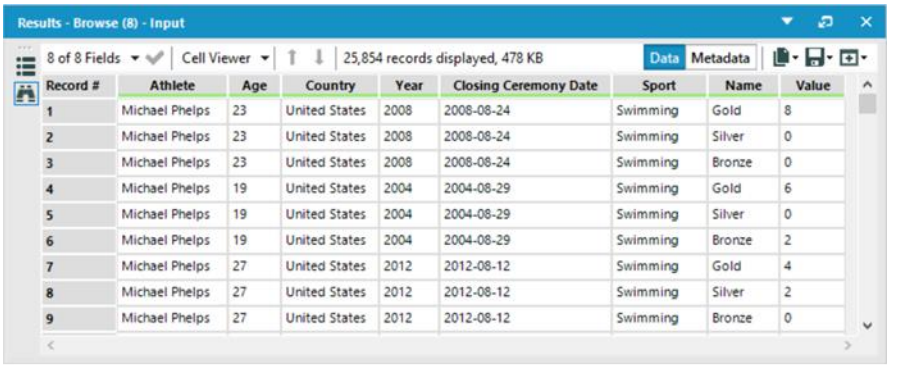


Figure-2-38-Let's Tidy Things Up – Transpose Configuration

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.

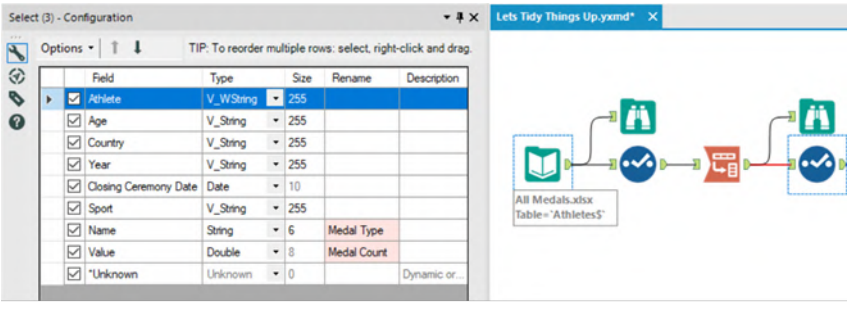


Figure-2-40-Let's Tidy Things Up – Select Configuration

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.

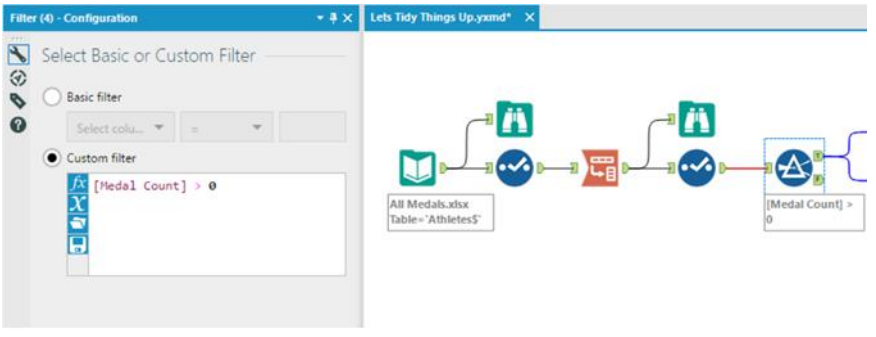


Figure-2-41-Let's Tidy Things Up – Filter Configuration

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*.

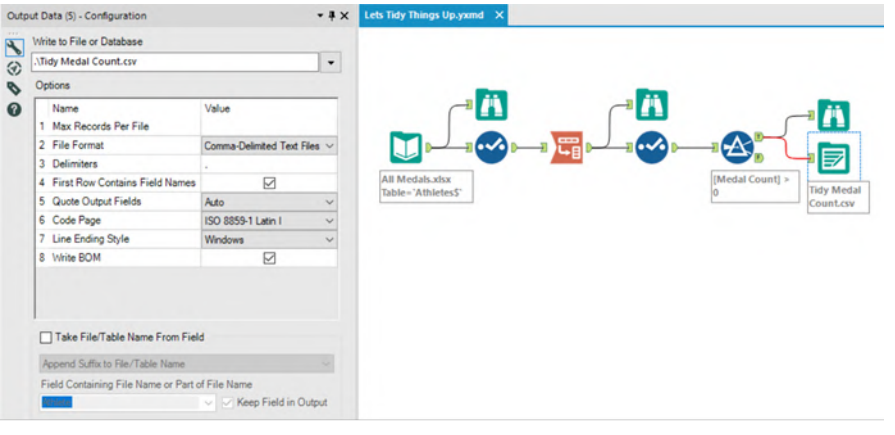


Figure-2-42-Let's Tidy Things Up – Output Data Configuration

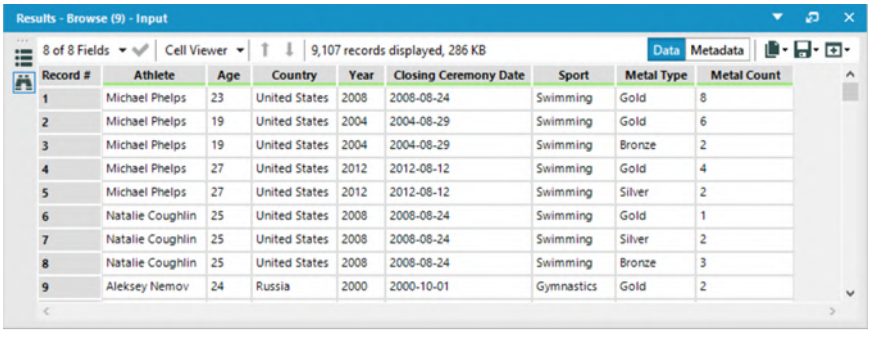
To do this, we are going to type.*Tidy Medal Data.csv* in the text box labeled *Write to File or Database*.

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.



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.

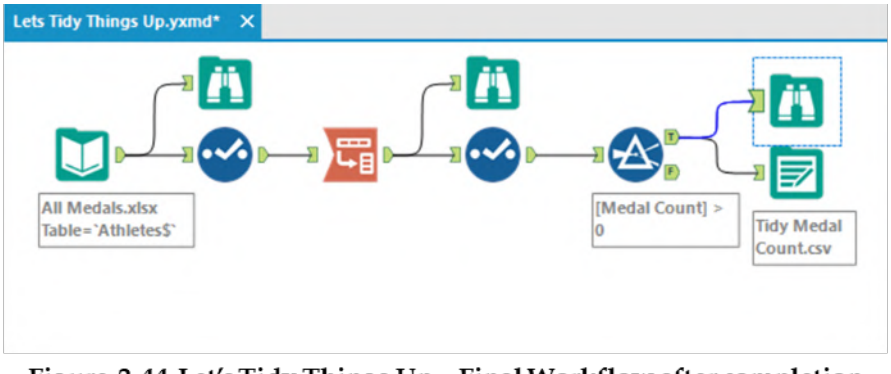
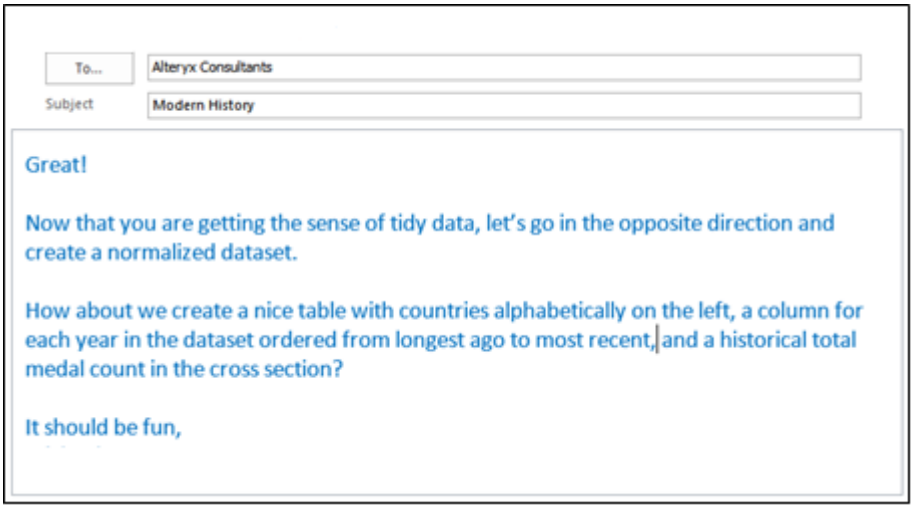


Figure-2-44-Let's Tidy Things Up – Final Workflow after completion

2.4 Modern History



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.

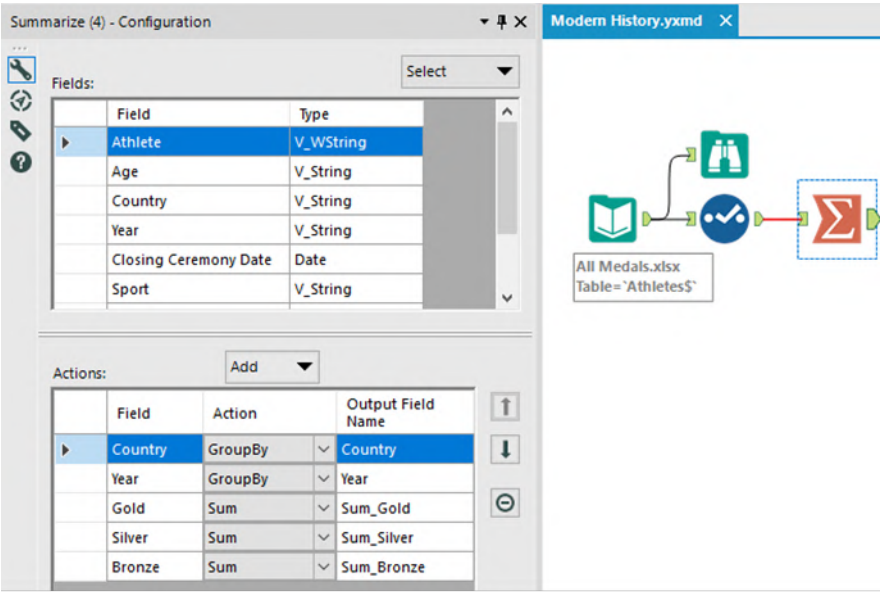


Figure-2-45-Modern History – Summarize Configuration

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.)

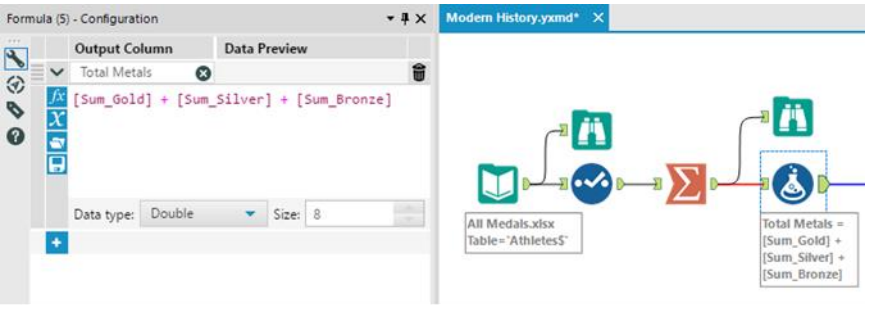


Figure-2-46-Modern History – Formula Configuration

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.

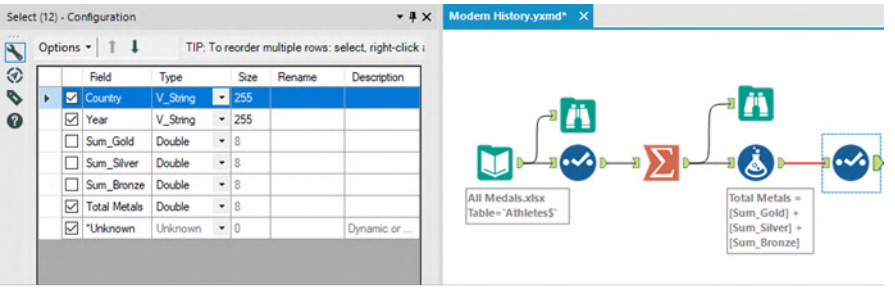


Figure-2-47-Modern History – Select Configuration

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.

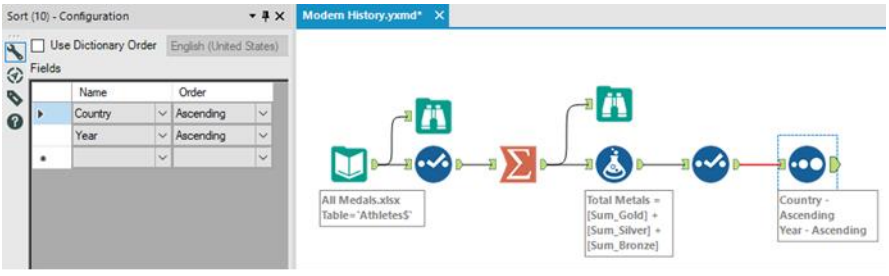


Figure-2-48-Modern History – Sort Configuration

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.

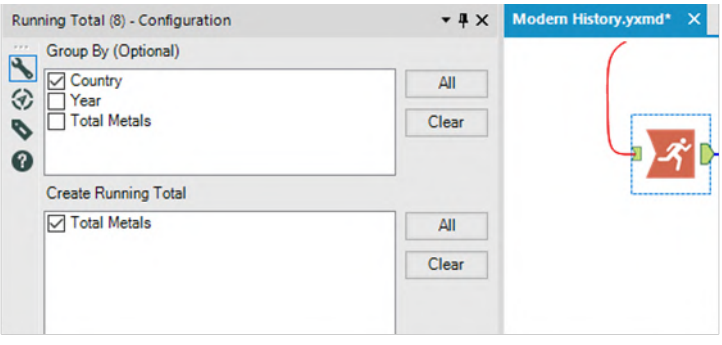
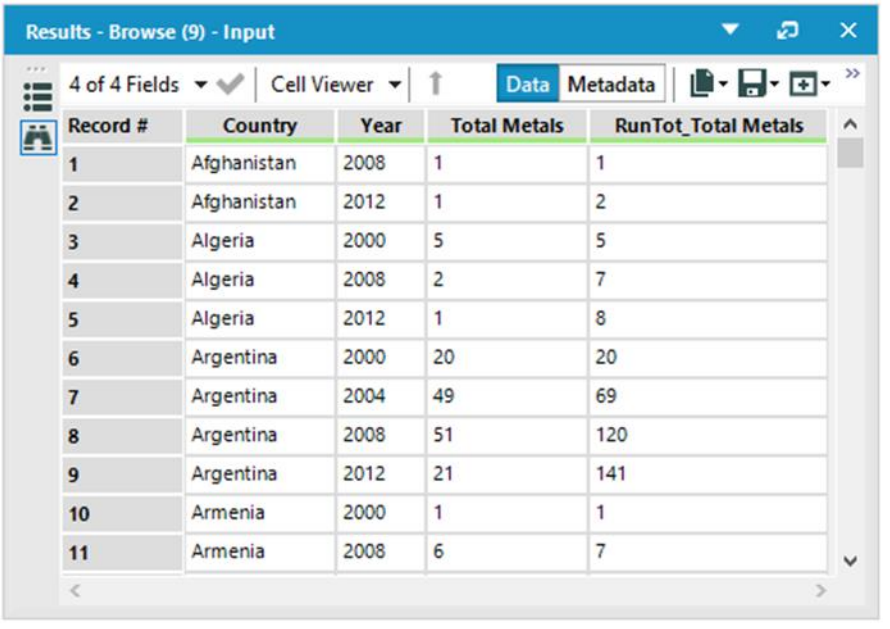


Figure-2-48-Modern History – Running Total Configuration

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.

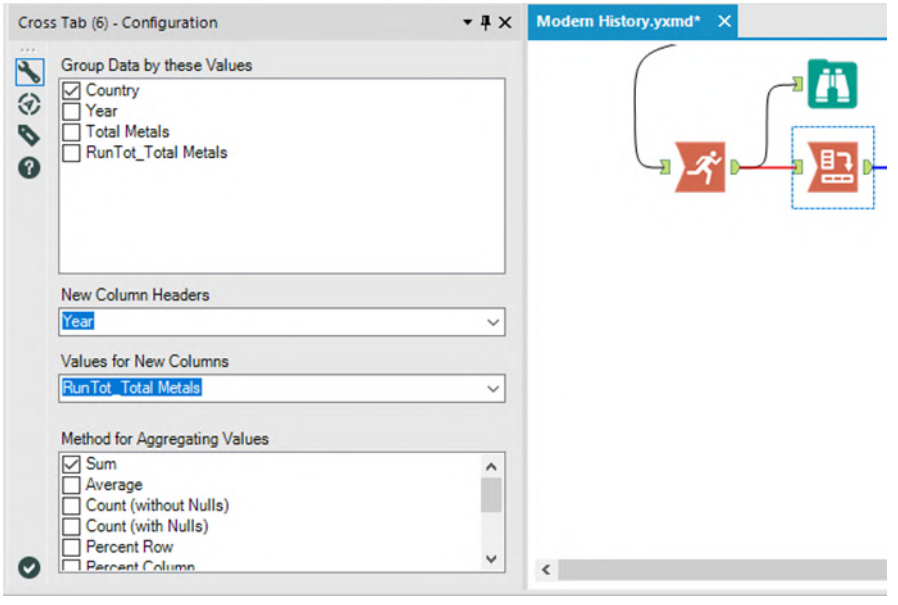
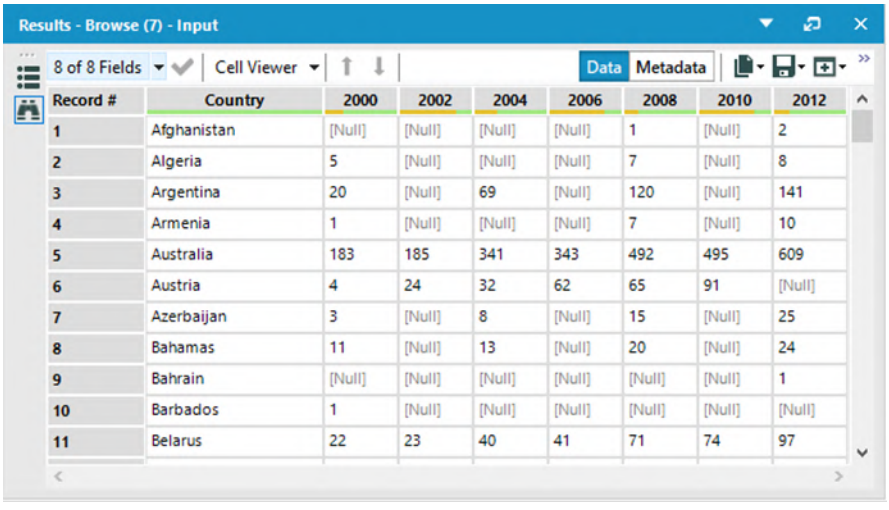


Figure-2-50-Modern History – Cross Tab

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:

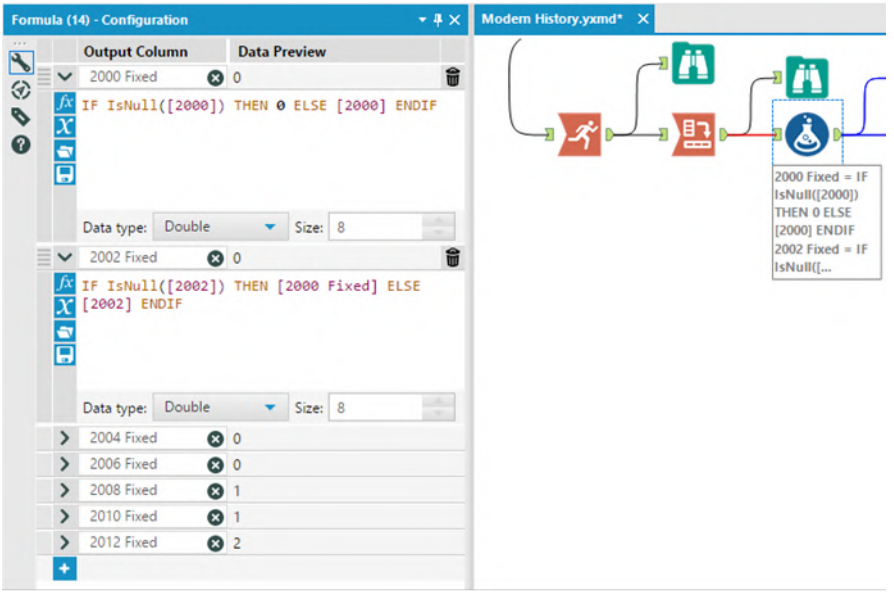
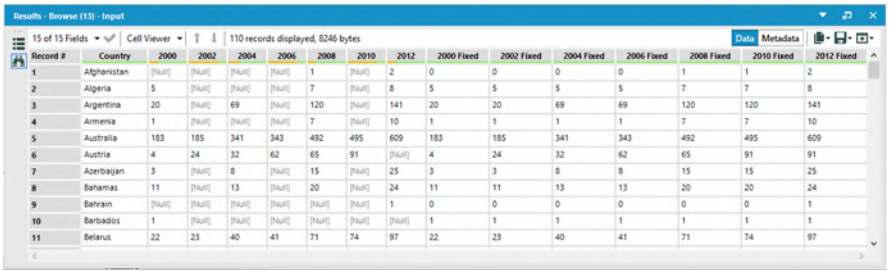


Figure-2-52-Modern History – Formula Configuration

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.

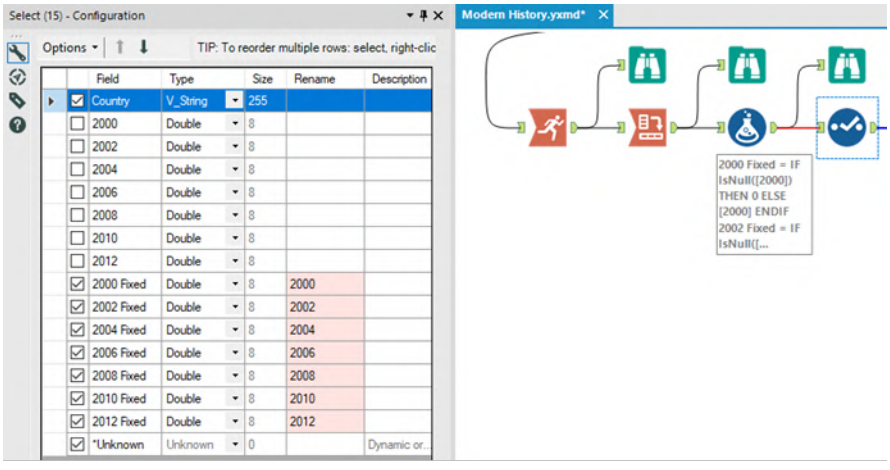
We can see that we have two sets of fields: those with the original sparse data and those with the new dense data.



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.

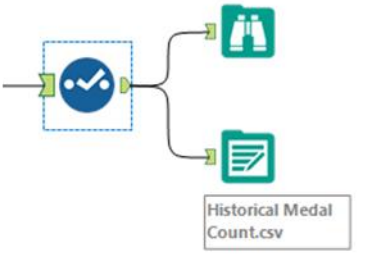


Figure-2-55-Modern History – Output

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

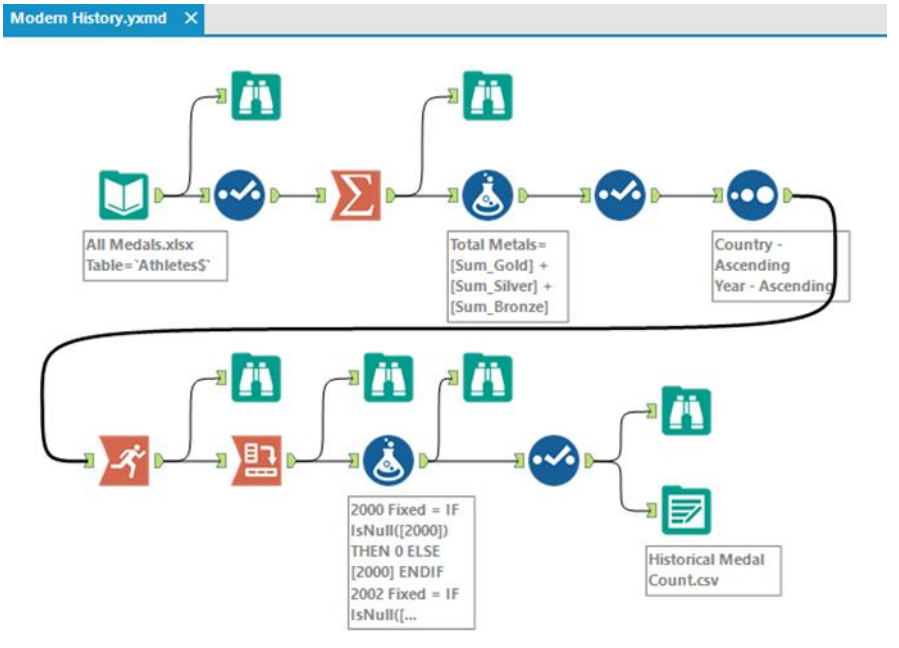
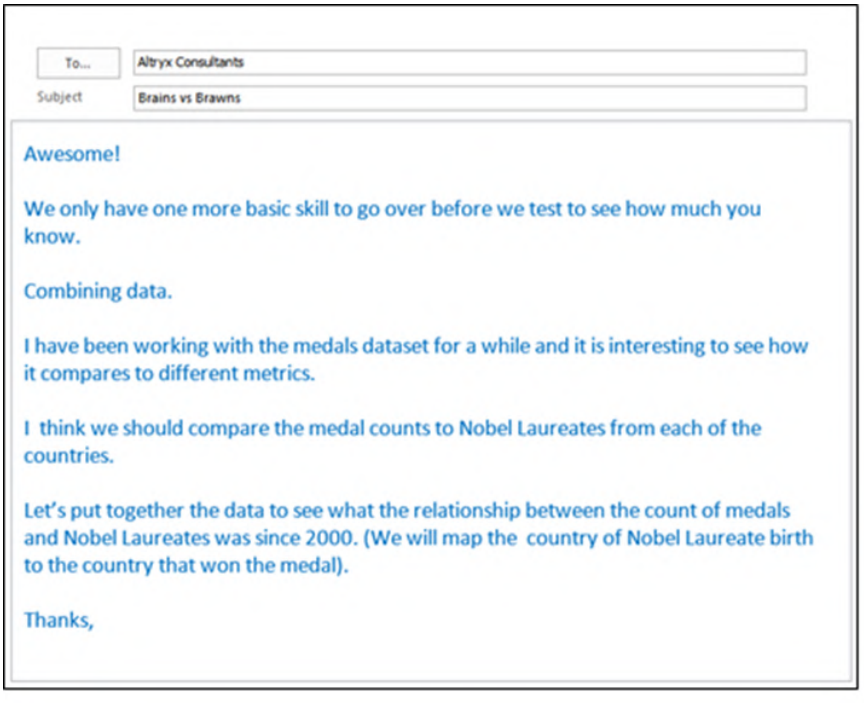


Figure-2-56-Modern History – Complete Workflow

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.

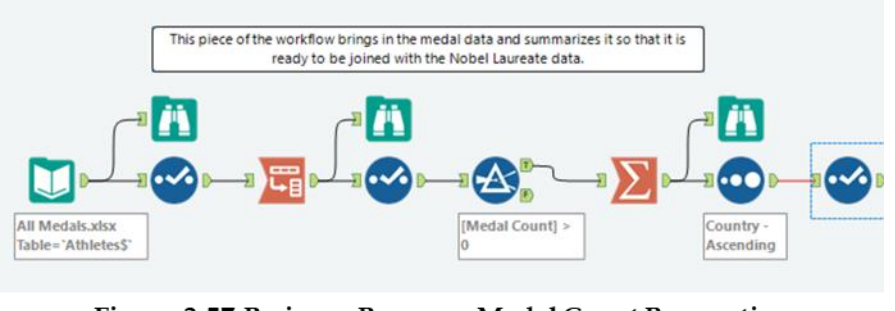


Figure-2-57-Brains vs Brawns – Medal Count Preparation

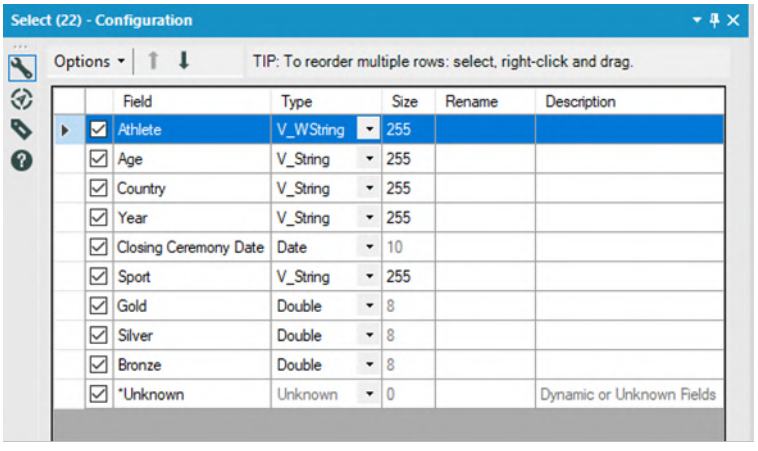


Figure-2-58-Brains vs Brawns – Initial Steps - Select

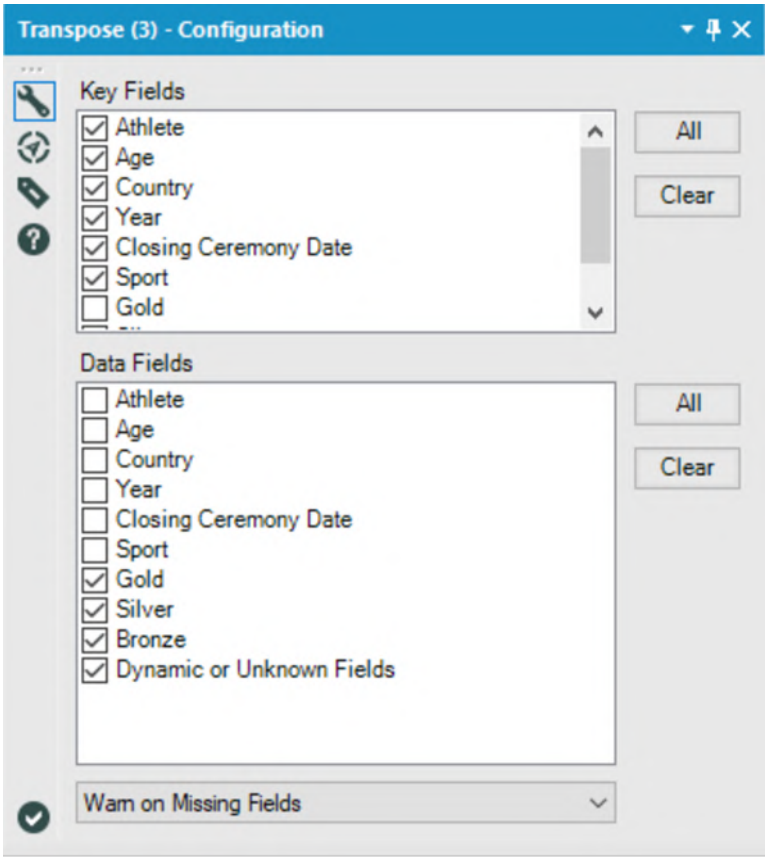


Figure-2-59-Brains vs Brawns – Initial Steps - Transpose

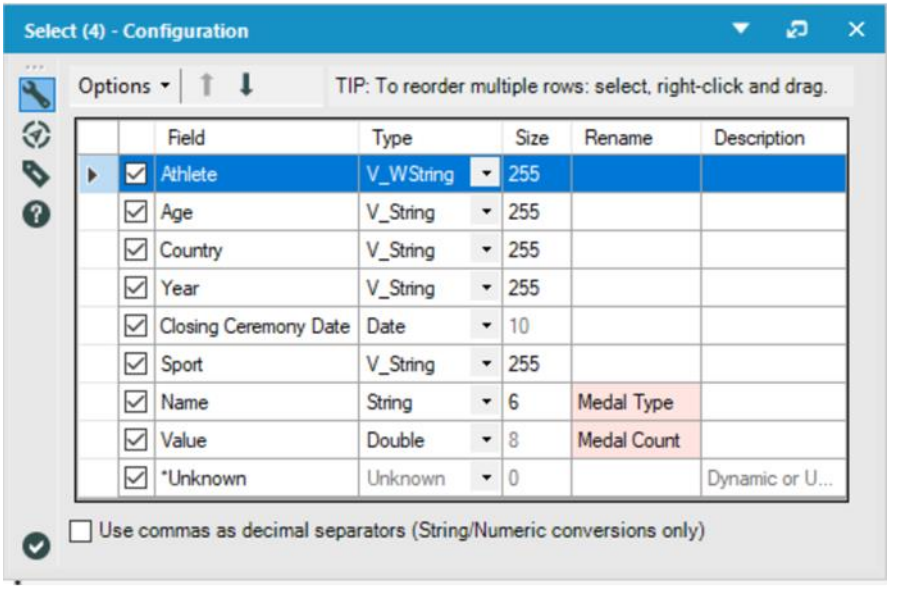


Figure-2-60-Brains vs Brawns – Initial Steps - Select

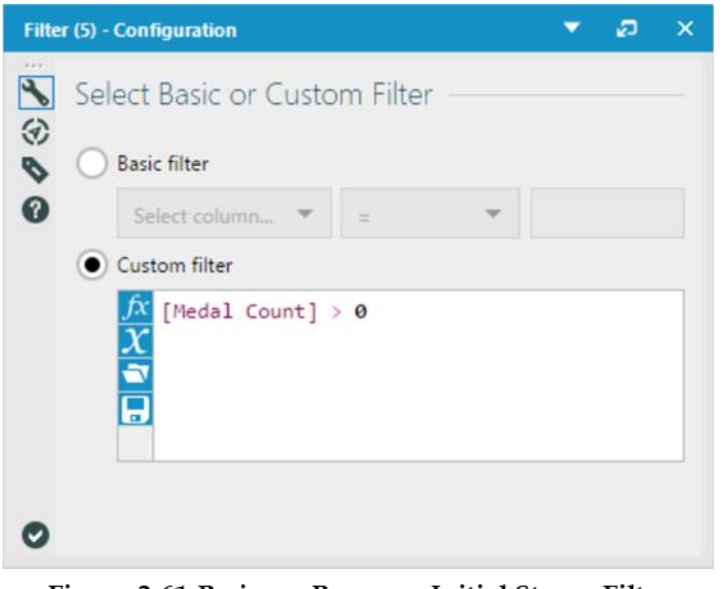


Figure-2-61-Brains vs Brawns – Initial Steps - Filter

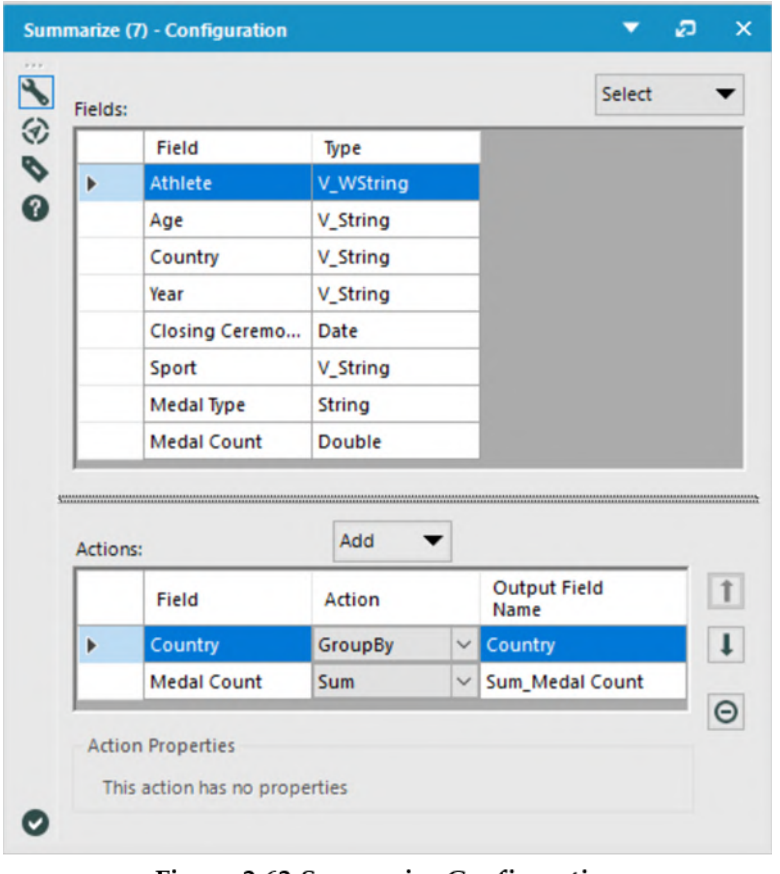


Figure-2-62-Summarize Configuration

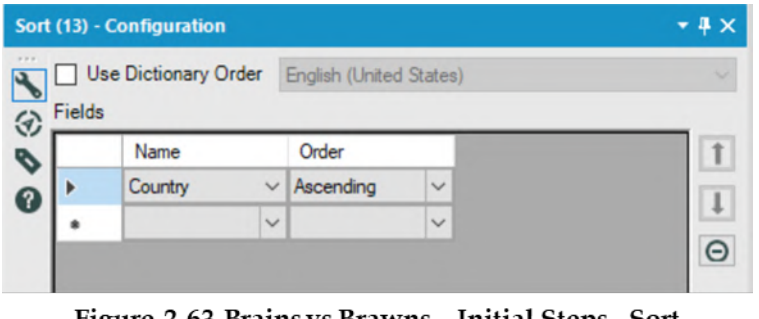


Figure-2-63-Brains vs Brawns – Initial Steps - Sort

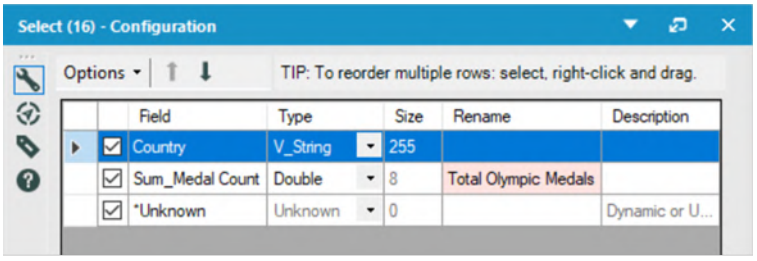


Figure-2-64-Brains vs Brawns – Initial Steps - Select

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.)

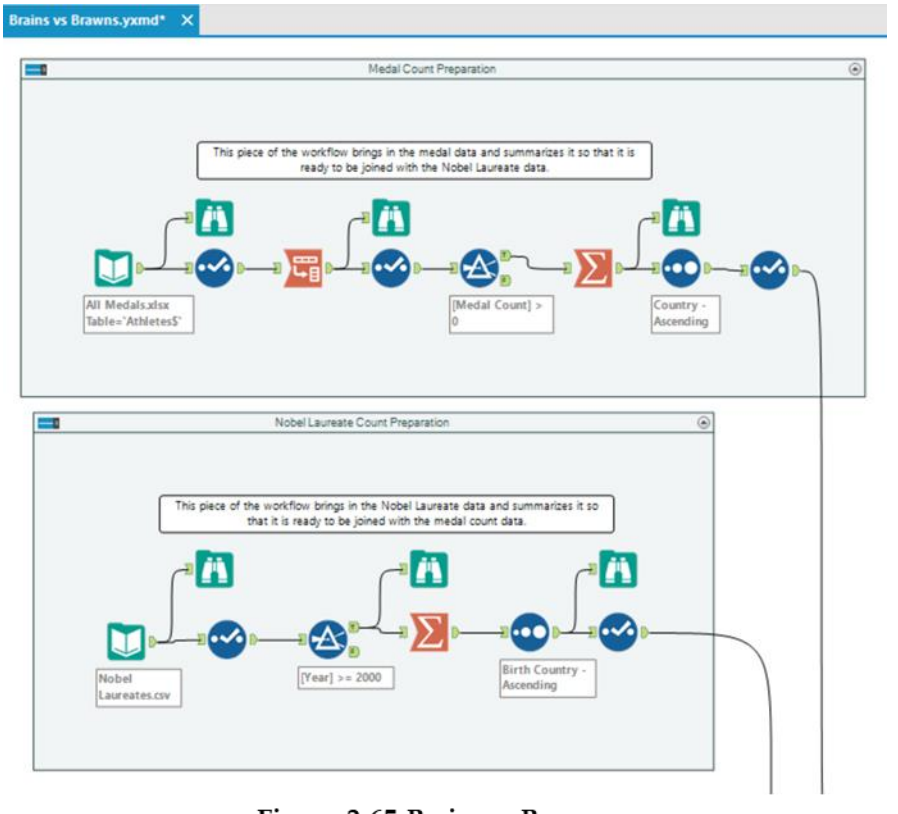
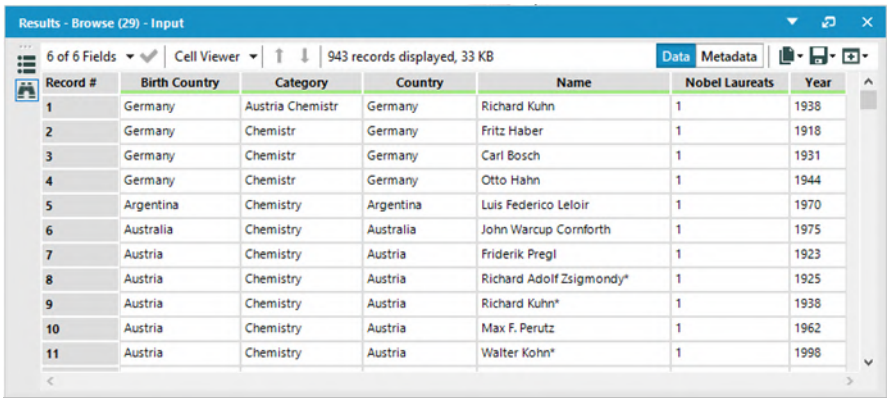


Figure-2-65-Brains vs Brawns –
Running unrelated analyses simultaneously

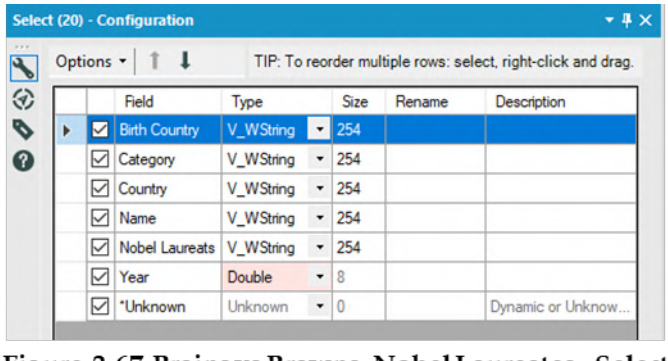
Notice that we now have two completely separate workflows. This is a useful feature because we can run unrelated analyses at the same time, which aids in testing and in conditional application development.

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**.

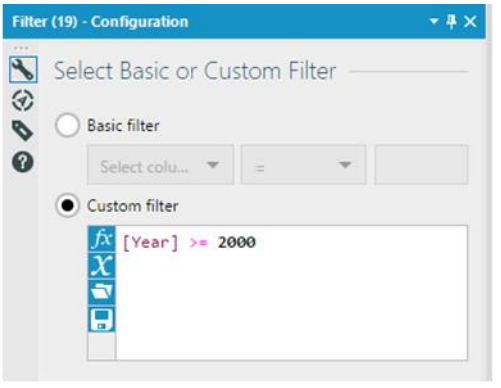


Figure-2-68-Brains vs Brawns, Nobel Laureates - Filter

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.

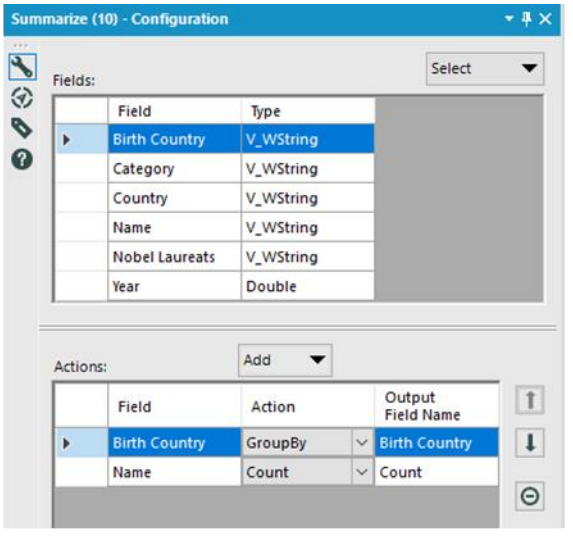


Figure-2-69-Brains vs Brawns, Nobel Laureates - Summarize

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

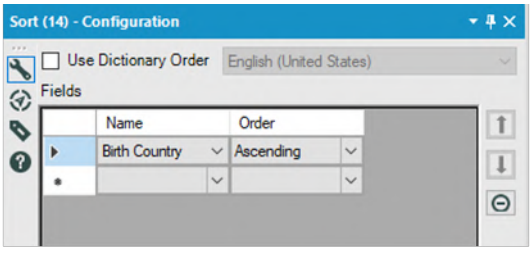
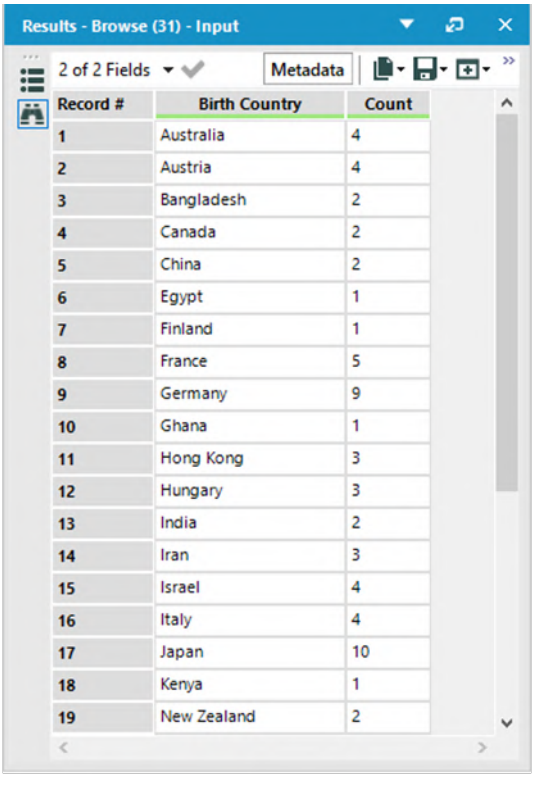


Figure-2-70-Brains vs Brawns, Nobel Laureates -Sort

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.

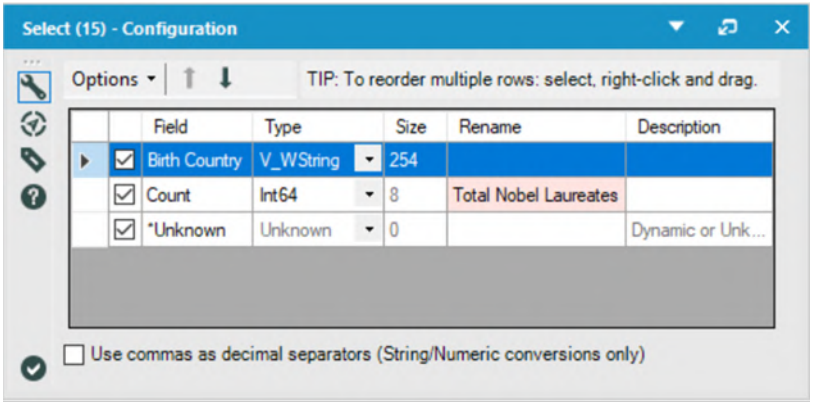


Figure-2-72-Brains vs Brawns, Nobel Laureates – Rename Count using Select

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.

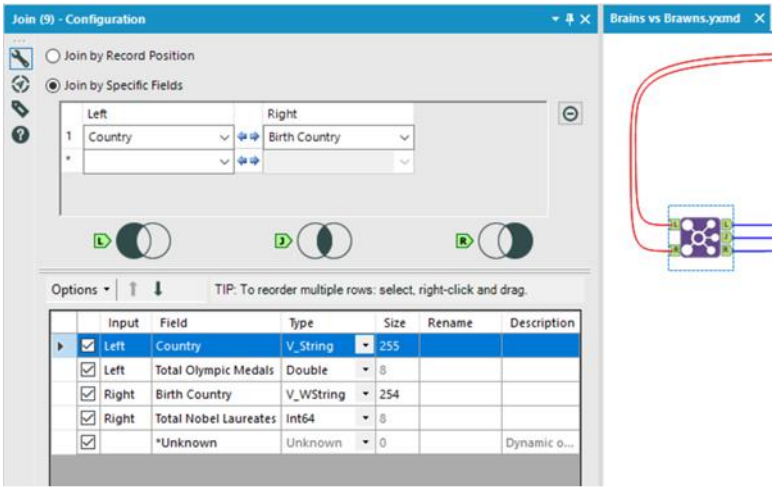


Figure-2-73-Brains vs Brawns, Nobel Laureates – Join Configuration

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*.

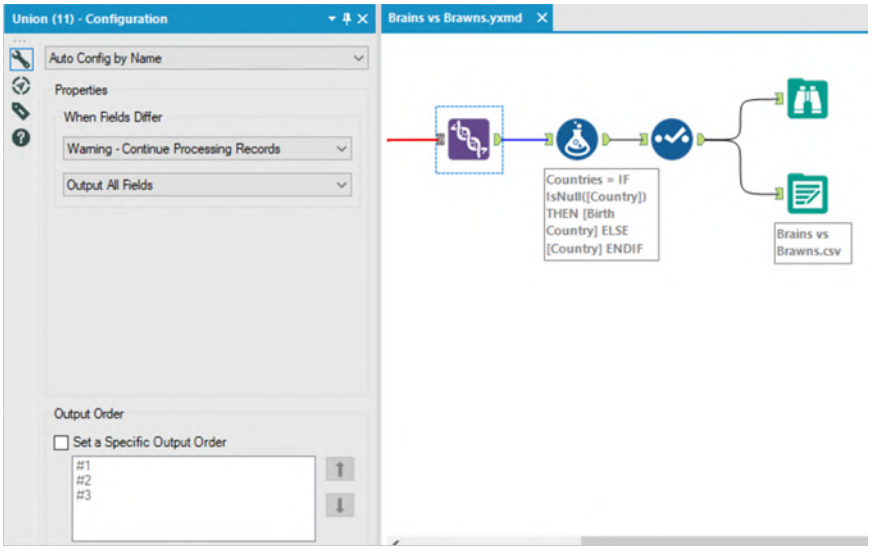


Figure-2-74-Brains vs Brawns, Nobel Laureates – Union Configuration

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.

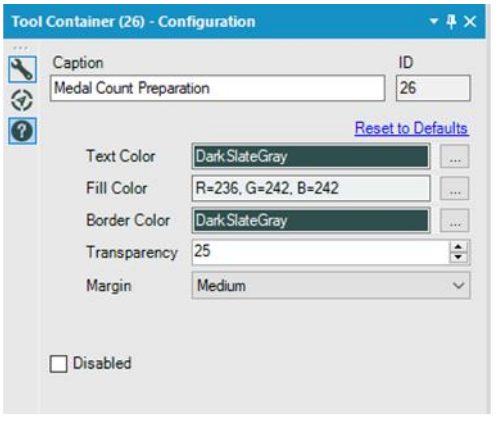


Figure-2-75-Brains vs Brawns, Medal Count Preparation – Tool Container Configuration

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

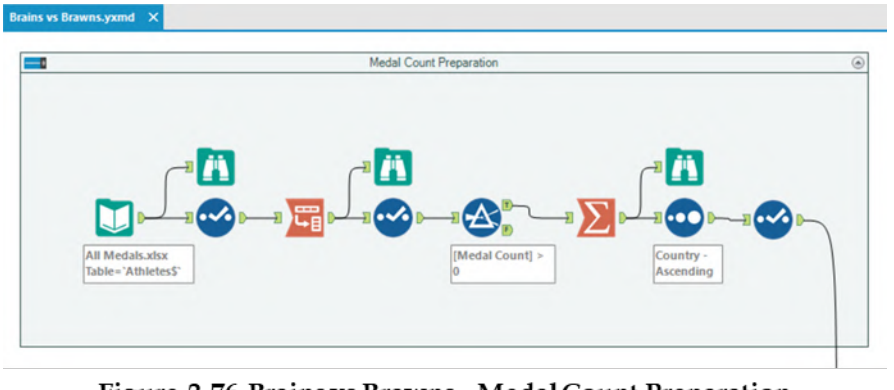


Figure-2-76-Brains vs Brawns - Medal Count Preparation stream with Tool Container

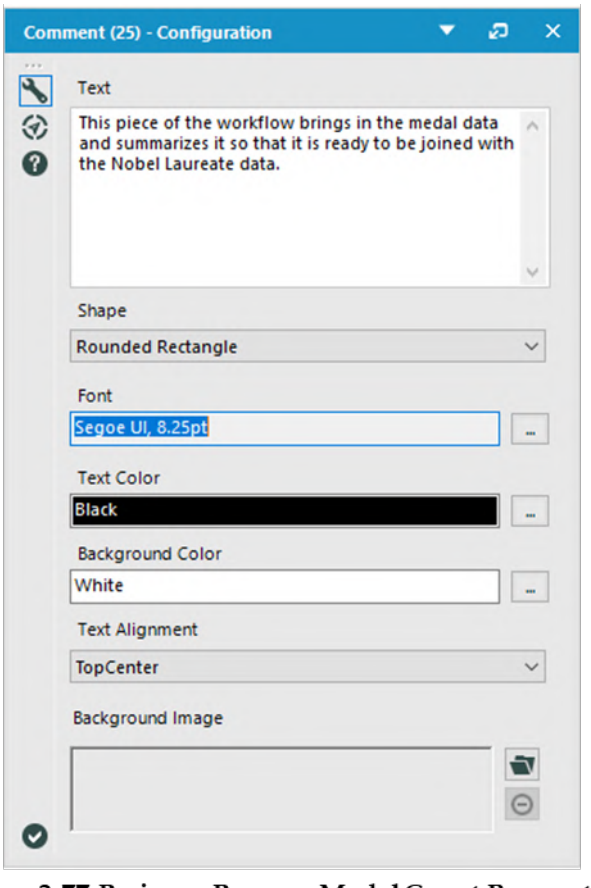


Figure-2-77-Brains vs Brawns, Medal Count Preparation – Comment Configuration

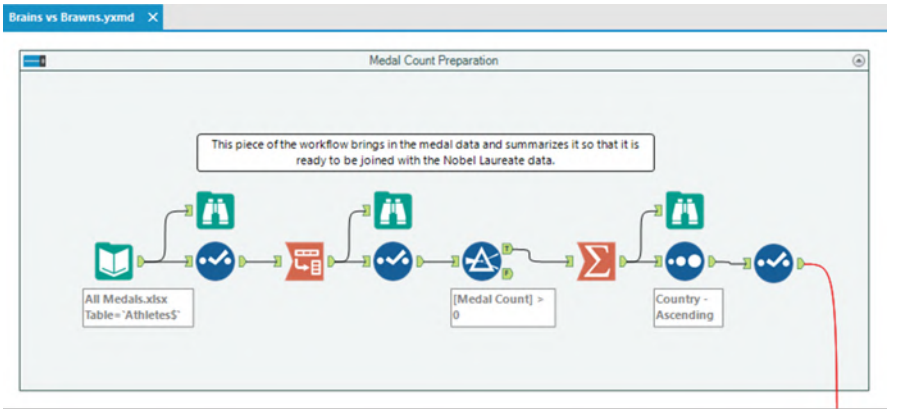


Figure-2-78-Brains vs Brawns - Medal Count Preparation Comment

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.

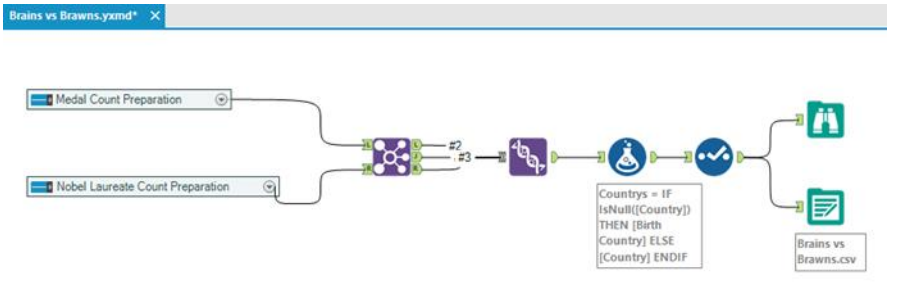


Figure-2-79-Brains vs Brawns – Simplified Workflow

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.

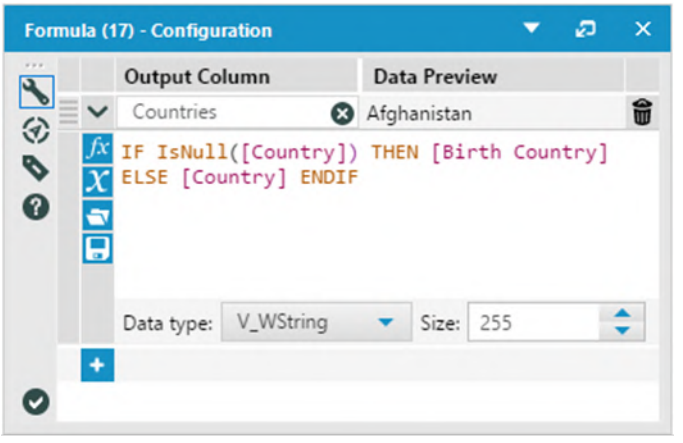


Figure-2-80-Brains vs Brawns, Final Output – Formula Configuration

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*.

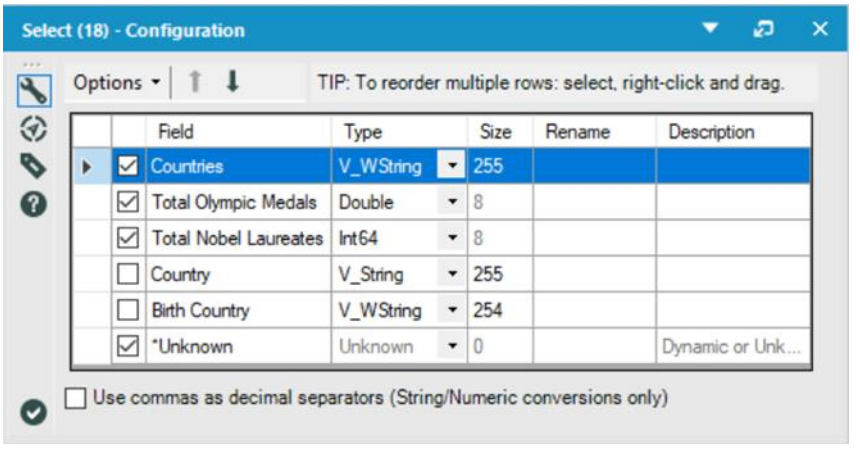
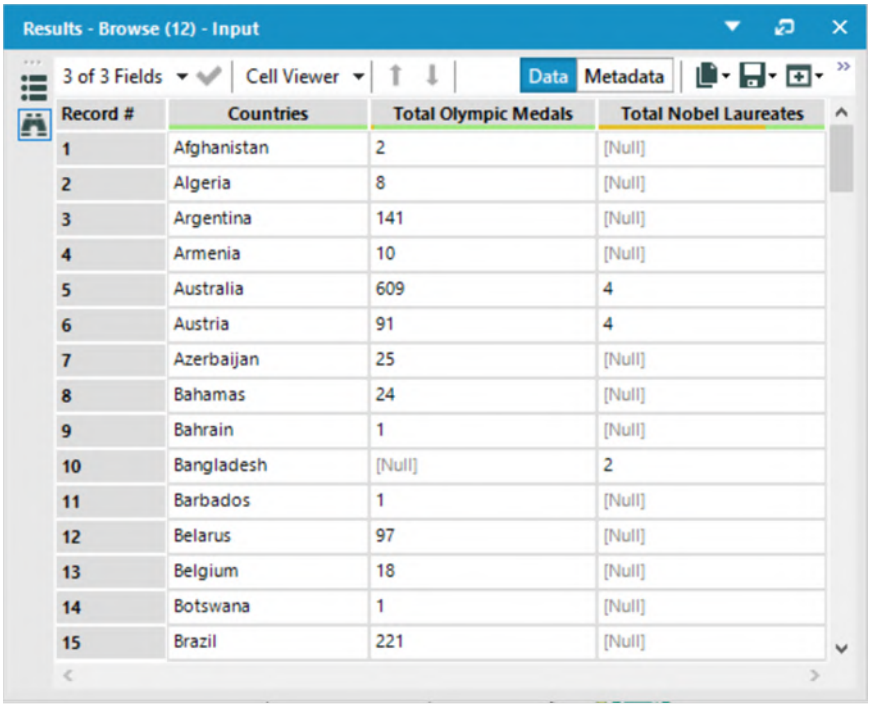


Figure-2-81-Brains vs Brawns, Final Output – Select Configuration

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.

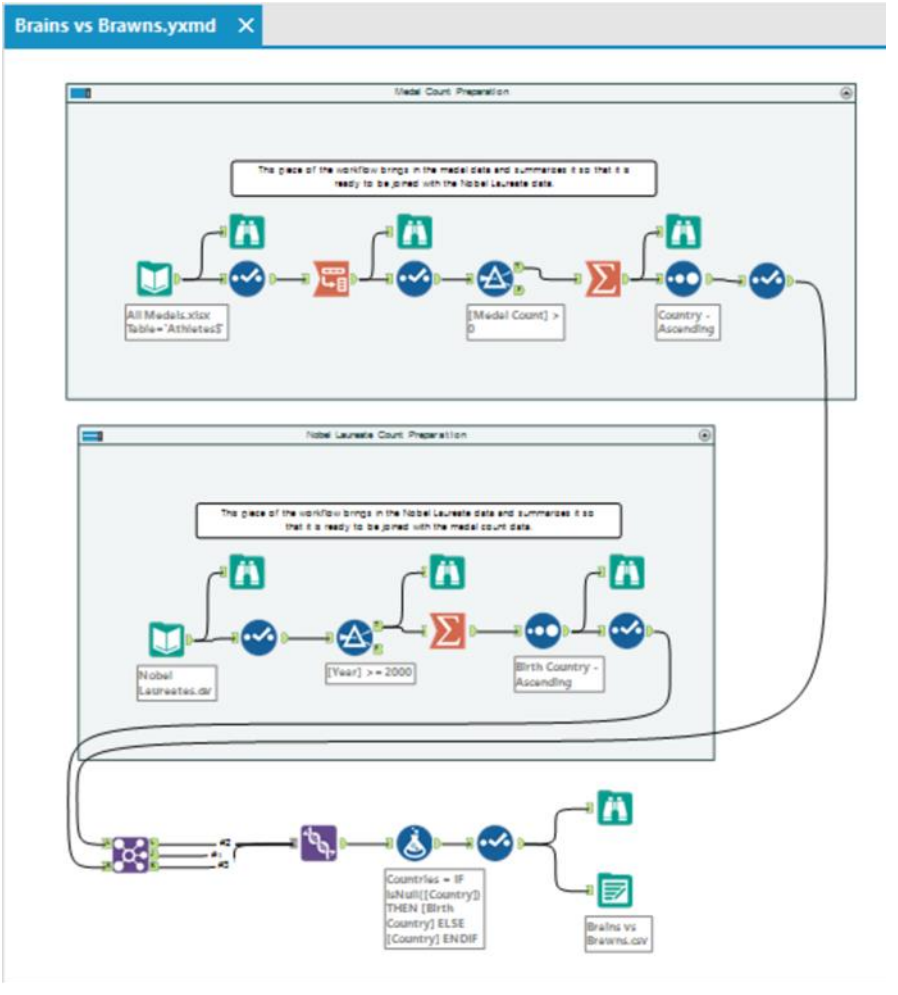
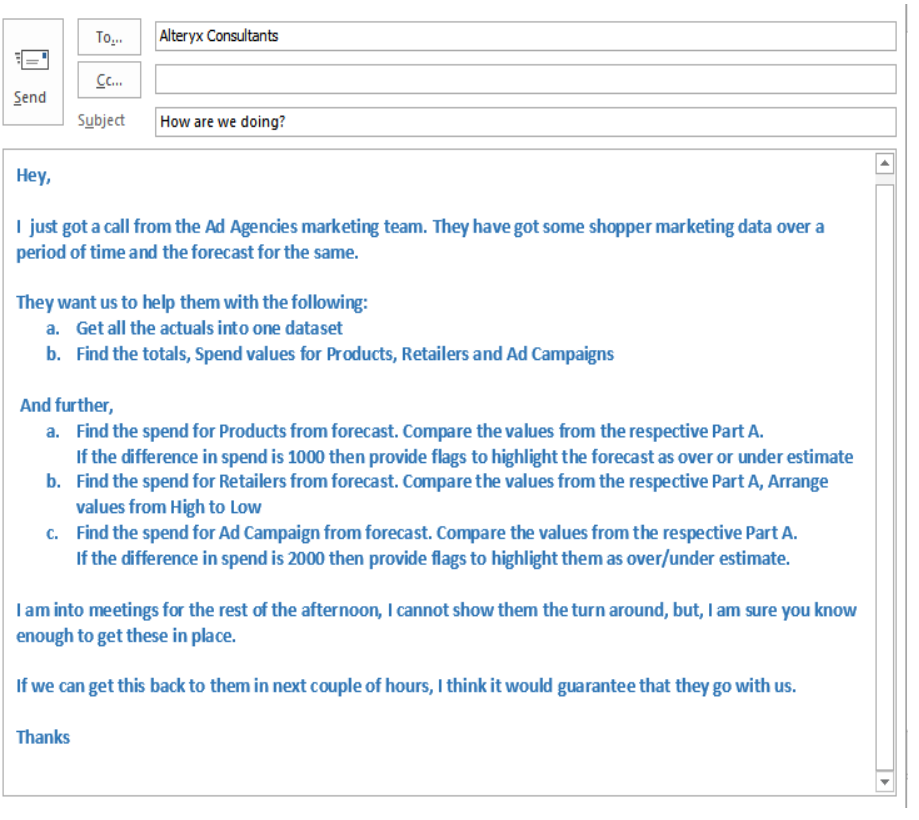


Figure-2-83-Brains vs Brawns – Complete Workflow

2.6 How are we doing?



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