Data Refinery Lab

This lab will introduce the Data Refinery tool included with Watson Studio. Data Refinery is a self-service capability used to cleanse and shape tabular data. Cleansing the data consists of fixing or removing data that is incorrect, incomplete, improperly formatted, or duplicated. Shaping the data consists of customizing it by filtering, sorting, combining or removing columns, and performing other operations to transform the data into the appropriate format for analysis.

You create a *Data Refinery flow* as a set of ordered operations on data. Data Refinery includes a graphical interface to profile your data to validate it and over 20 customizable charts that give perspective and insights into your data. When you save the refined data set, you typically load it to a different location than where you read it from. In this way, your source data remains untouched by the refinement process.

As you interact with the Data Refinery tool, it will perform the ordered operations on a subset of the data. When you are satisfied with the flow of operations, you save the Data Refinery flow and then run a job to apply the series of operations on the entire dataset.

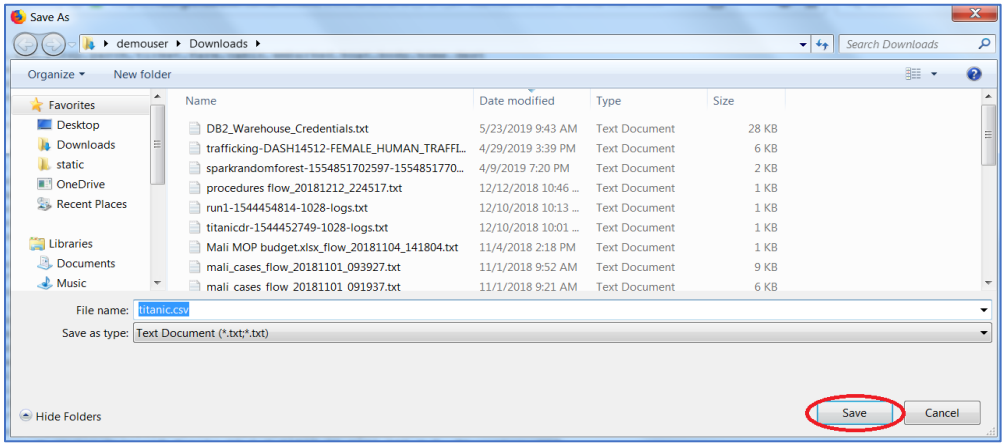
This lab will use the Titanic data set to demonstrate data profiling, data visualization, and data preparation capabilities of the Data Refinery tool. The lab consists of the following steps:

1. Use the Data Refinery Tool to:
    a. Profile the data to help determine missing values
    b. Visualize the data to gain a better understanding
    c. Prepare the data for modeling
    d. Run the sequence of data preparation operations on the entire data set.

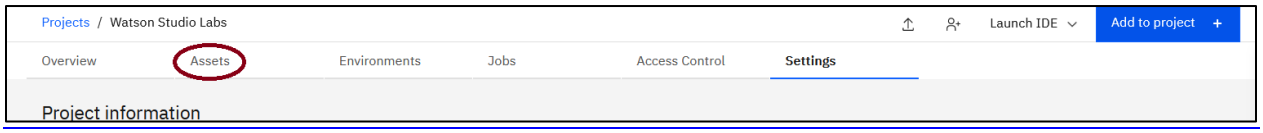
Step 1: Adding a Data Asset to the Watson Studio Labs project

1. Download the Titanic data file from the following location by clicking here.
2. Right-click on the screen and click on **Save Page As …** Note: If you are completing the labs on a Mac, you may not be able to save the file by right-clicking. Skip to **Appendix: Steps on Mac.**

3. Click on **Save** to save the titanic.csv file (Note, if the file shown is titanic.csv.txt, remove the .txt).

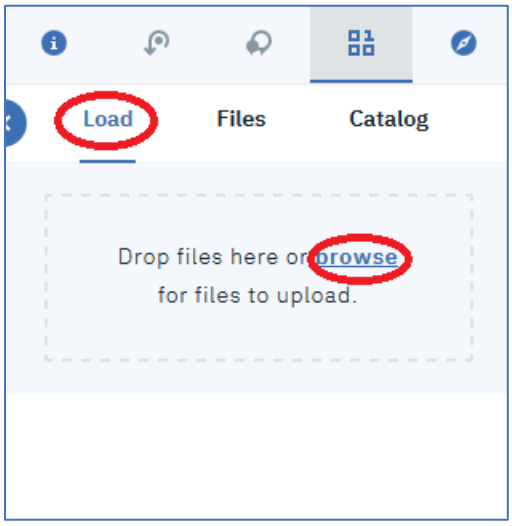


4. Go back to your Watson Studio Labs project. Click on the **Assets** tab.

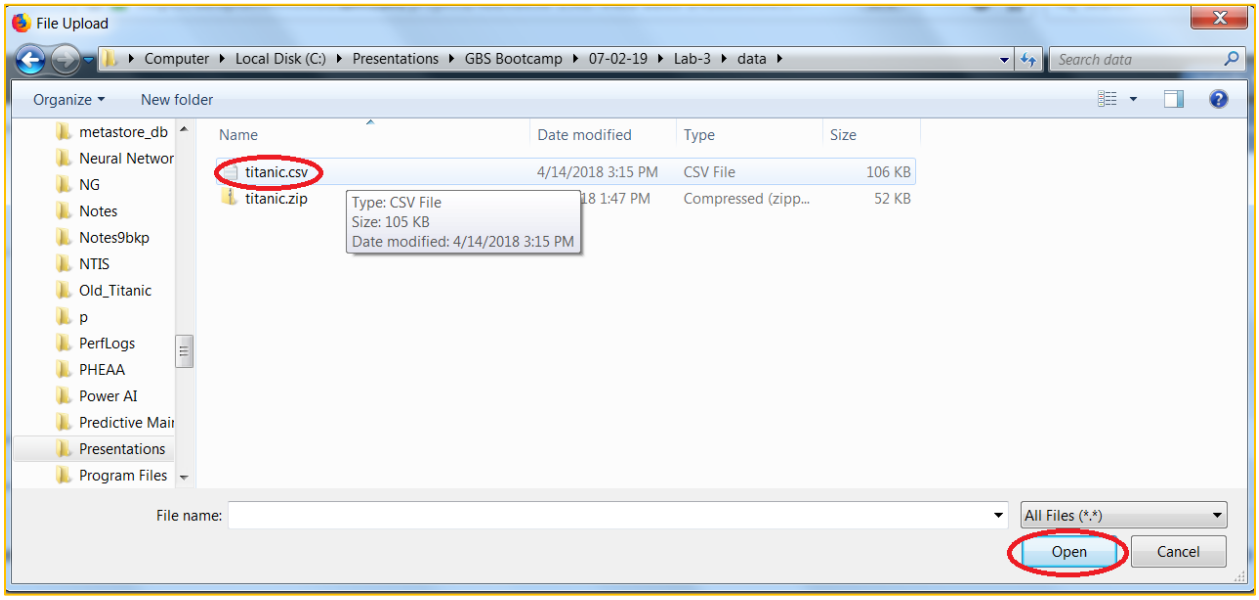


5. Click on the  icon.

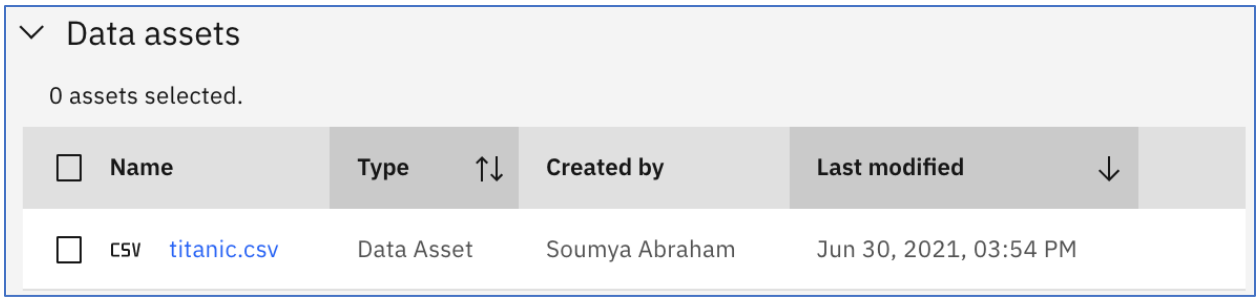
6. Click on the **Load** tab and then click on **browse**. If you don't see the **Load** tab, click on the  icon again.



7. Go to the folder where the titanic.csv file is stored. Select the titanic.csv file and then click **Open**.

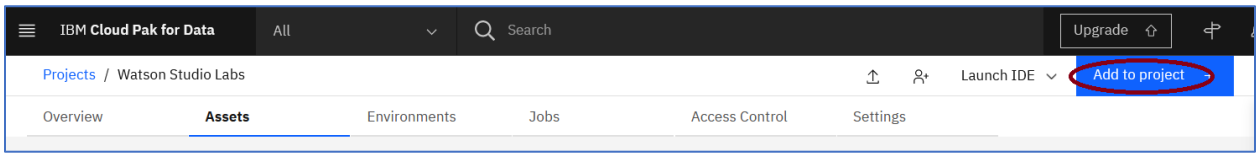


8. The file is now added as a Data Asset.

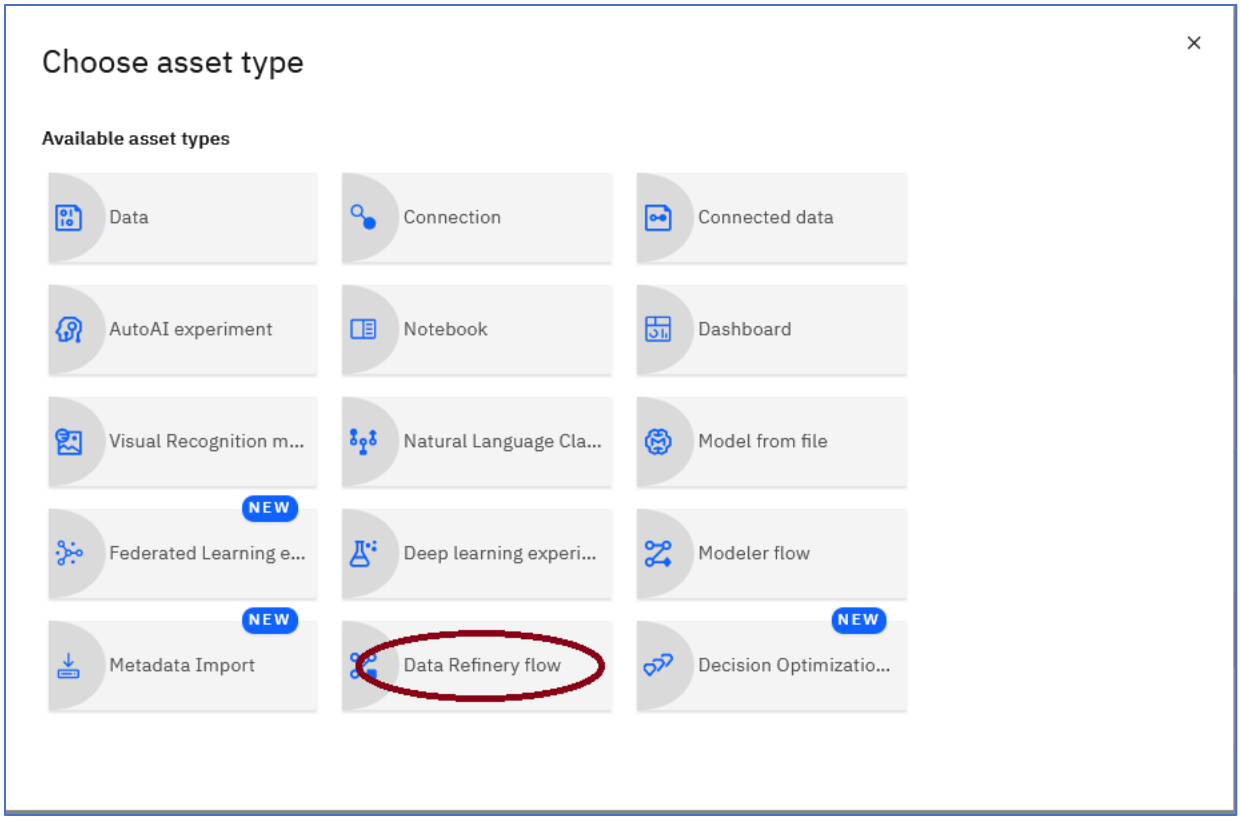


Step 2: Profile the data to help determine missing values.

1. Add a Data Flow by clicking on **Add to project.**

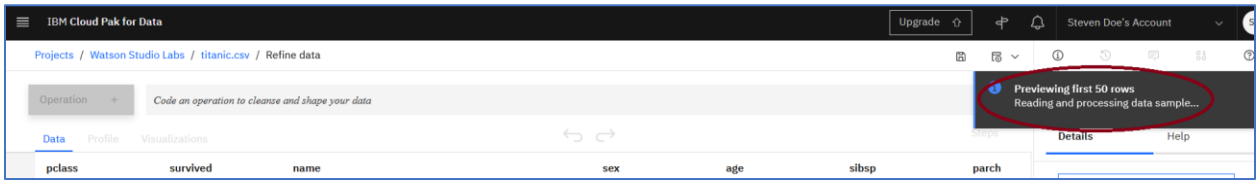


2. Click **Data Refinery flow**.

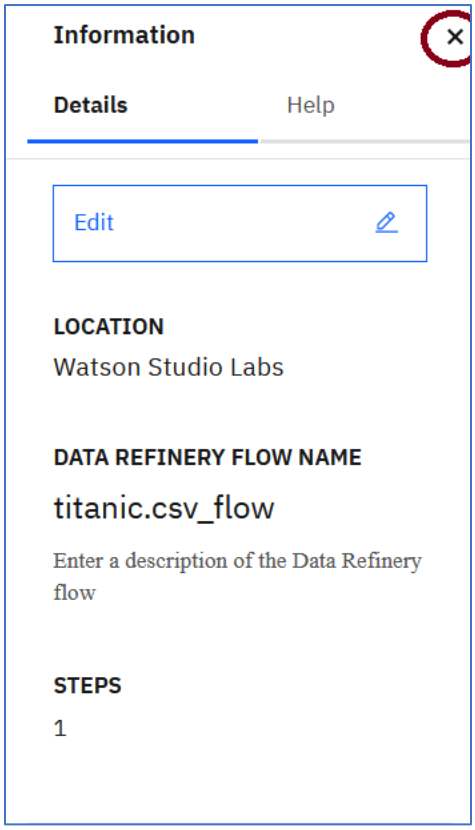


3. Select **Data assets**, and then **titanic.csv** and then click on **Add**.

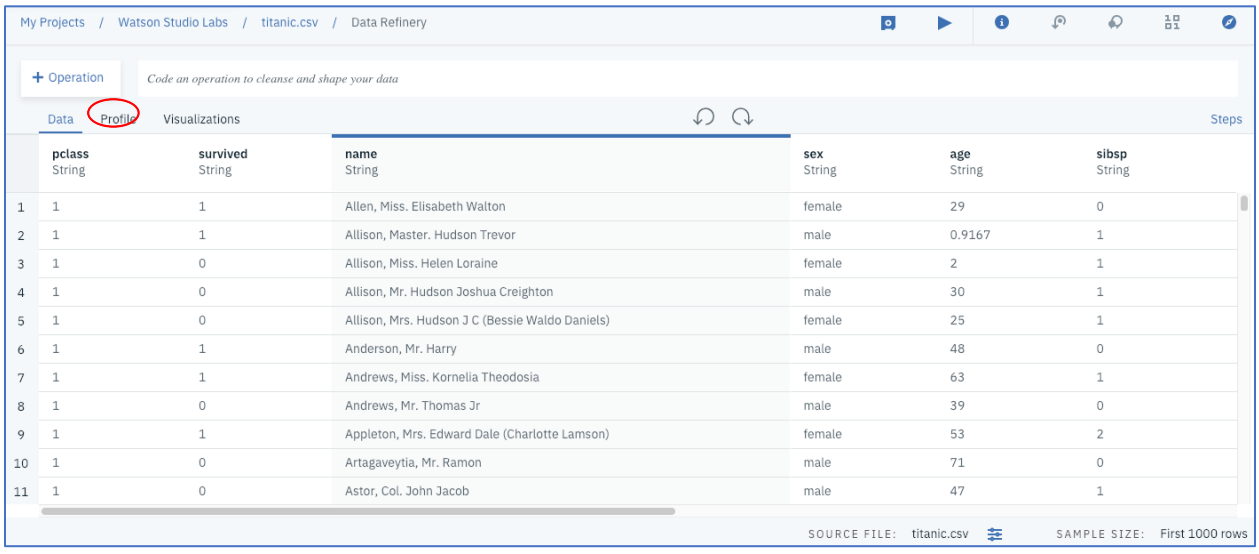
4. The Data Refinery panel will display the Titanic data set. Wait for the **Previewing first 50 rows** message to disappear.



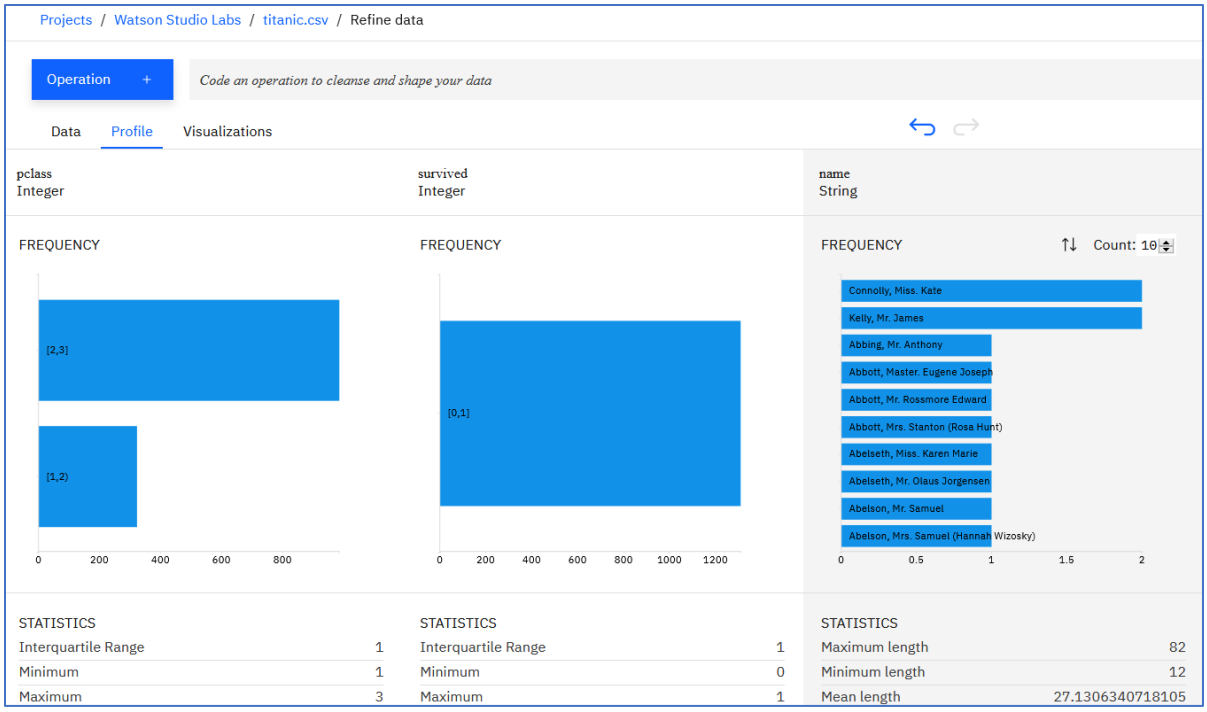
5. Close the **Information** panel by clicking **x.**



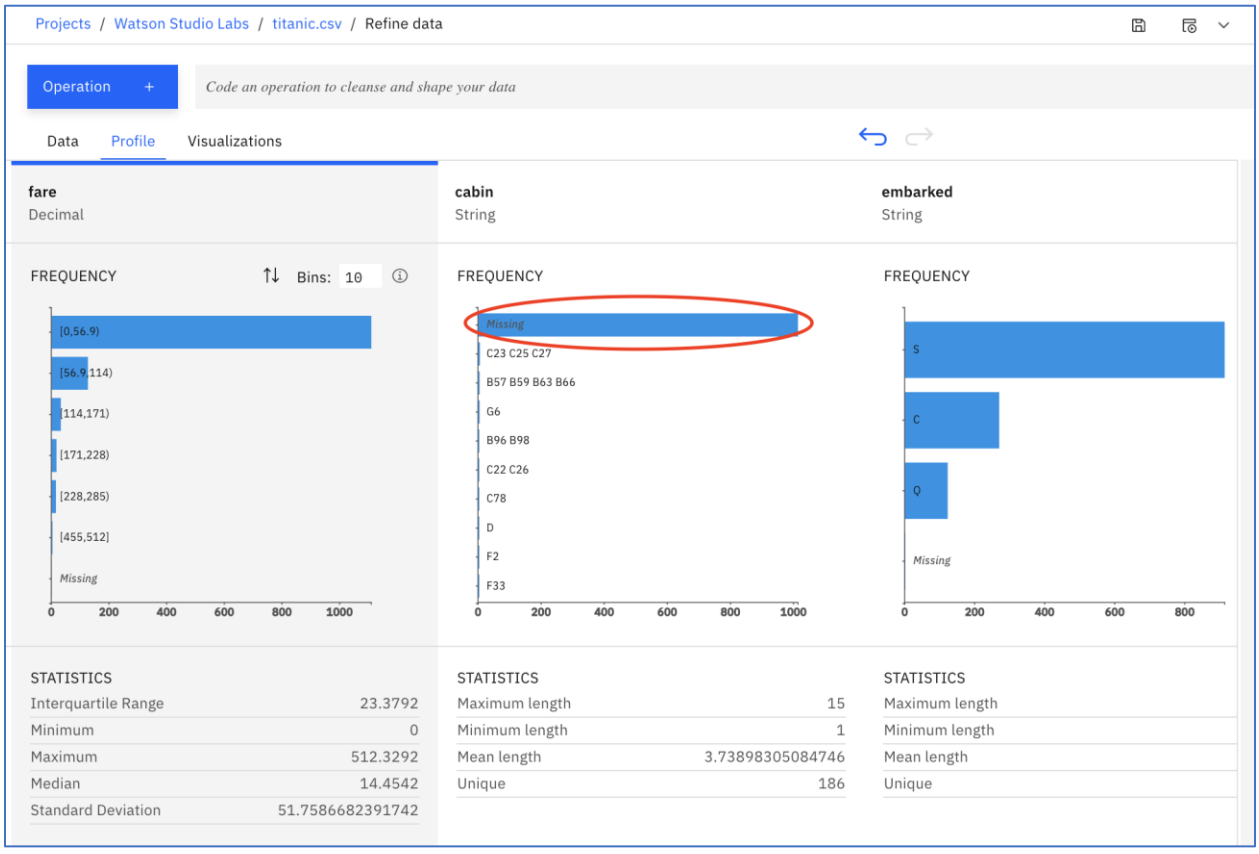
6. Click on the **Profile** tab.



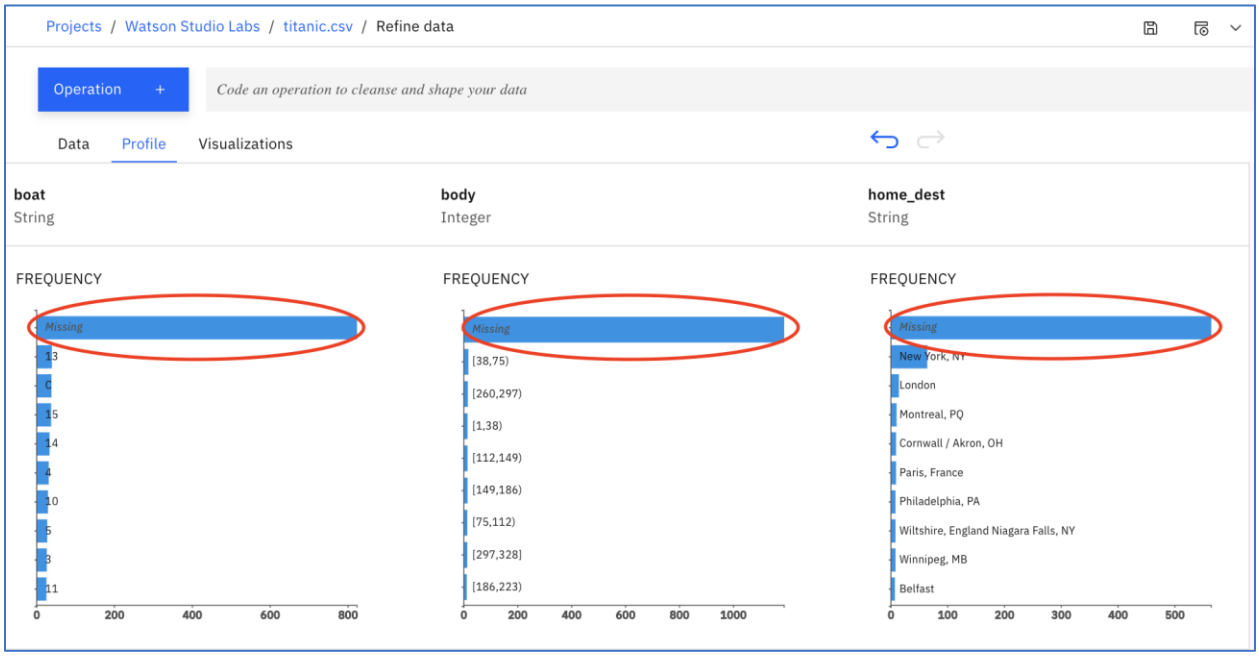
7. The Profile panel displays the counts of the top 10 count values for each categorical column, and a histogram for numerical data. You can also switch to sort from the bottom.



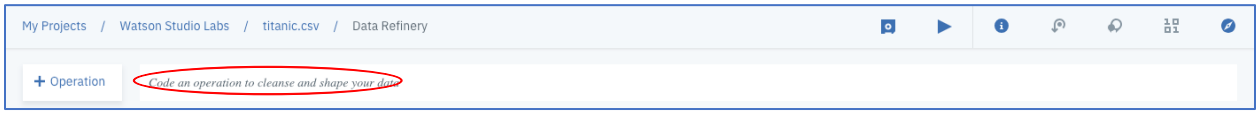
8. Scroll to the right to view the cabin column. Note that the cabin column has many missing values and should be removed as part of the data preparation step.



9. In a similar fashion, scroll to the right to examine the boat, body, and home_dest columns. These also have many missing values and should be removed as part of the data preparation step.

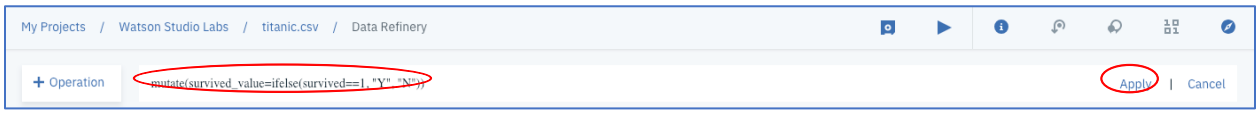


10. Age and Embarked also have missing values. Embarked has very few missing values. Age has over 200 missing values, but we will keep that column in the analysis. As part of data preparation, we will remove the rows that contain the missing age and embarked values.
11. Click on the **Data** tab. We will add columns that contain more readable values for the survived and pclass columns. The column survived_value will contain a "Y" or "N". The pclass_value column will contain "first", "second", or "third". We will use the mutate (R dpylr function) and ifelse functions to do the conversion. (Note, we could also use a Conditional Replace which would not require coding). Click on the **Code an operation to cleanse and shape your data.**

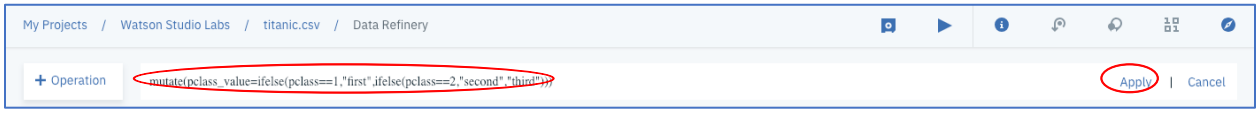


12. Copy and paste the following:
    mutate(survived_value=ifelse(survived==1, "Y", "N"))

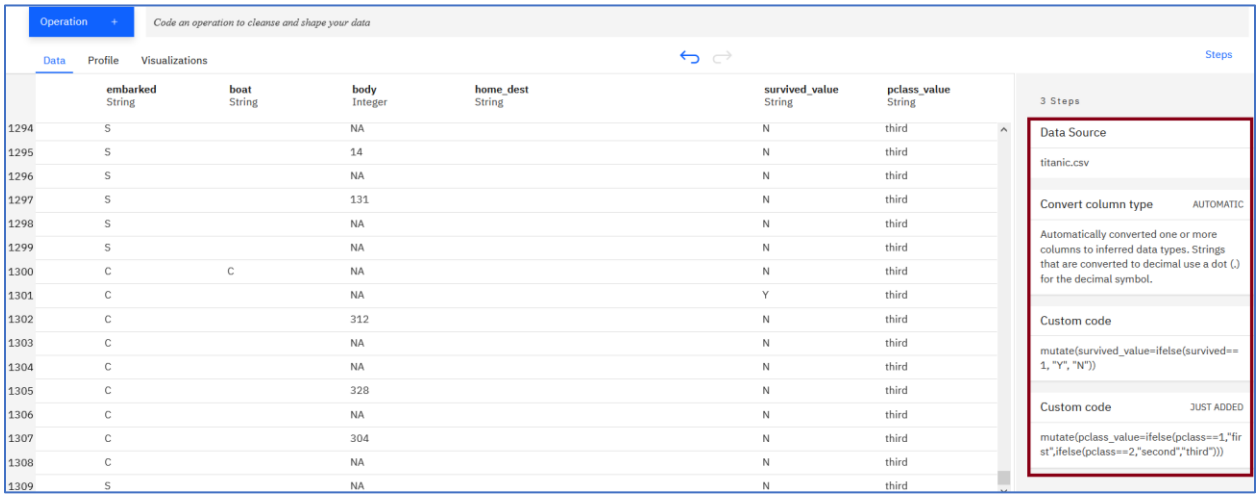
    and then click Apply. If you scroll to the right, you should see the new column "survived_value".



13. Copy and paste the following to create pclass_value, mutate(pclass_value=ifelse(pclass==1,"first",ifelse(pclass==2,"second","third")))



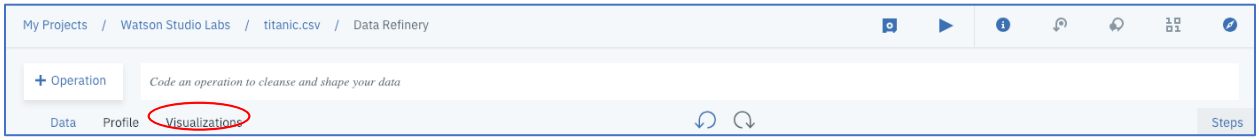
14. The survived_value and pclass_value columns are show below. Notice that the **Steps** panel will contain a running list of the transformations. The first transformation in the list is applied by default by the Data Refinery tool to infer data types.



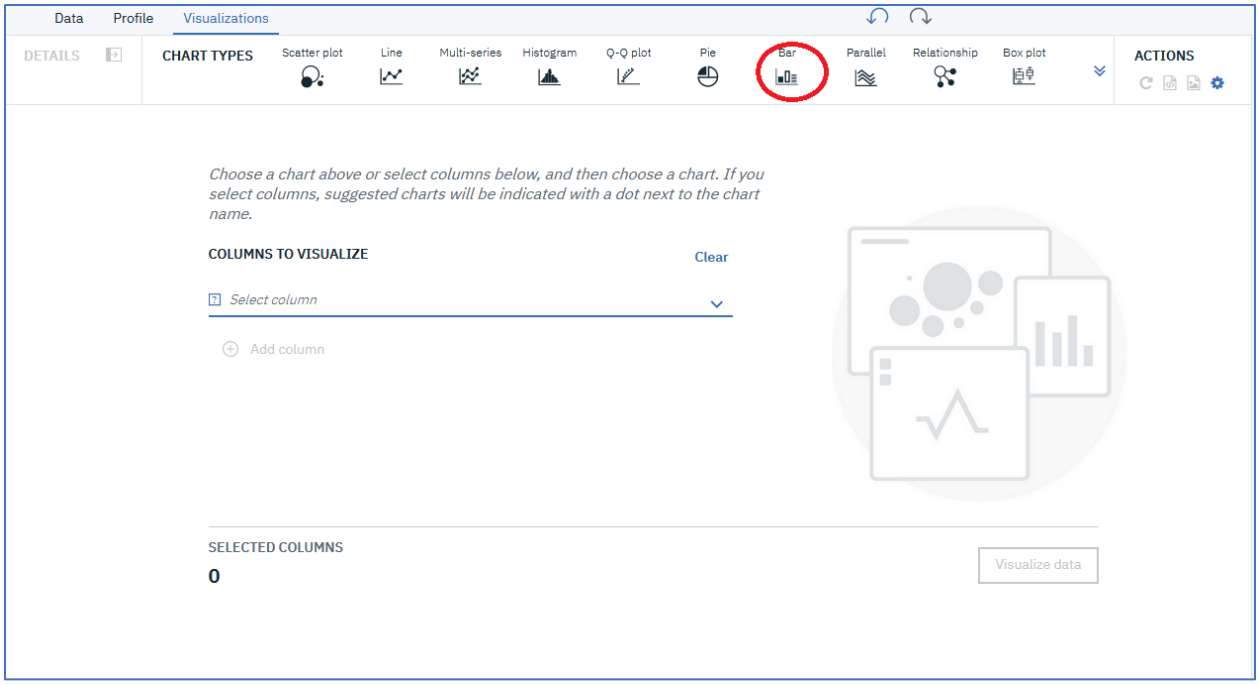
	embarked String	boat String	body Integer	home_dest String	survived_value String	pclass_value String
1294	S		NA		N	third
1295	S		14		N	third
1296	S		NA		N	third
1297	S		131		N	third
1298	S		NA		N	third
1299	S		NA		N	third
1300	C	C	NA		N	third
1301	C		NA		Y	third
1302	C		312		N	third
1303	C		NA		N	third
1304	C		NA		N	third
1305	C		328		N	third
1306	C		NA		N	third
1307	C		304		N	third
1308	C		NA		N	third
1309	S		NA		N	third

Step 3: Visualize the data to get a better understanding

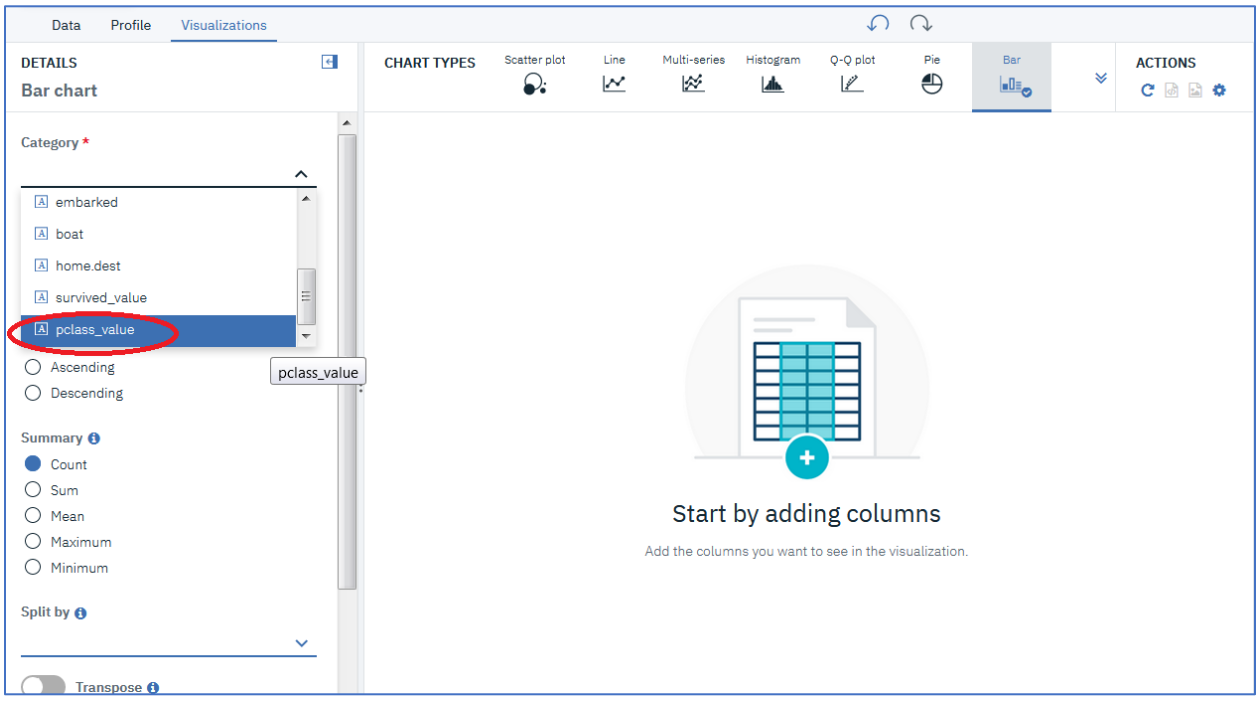
1. Click on the **Visualizations** tab.



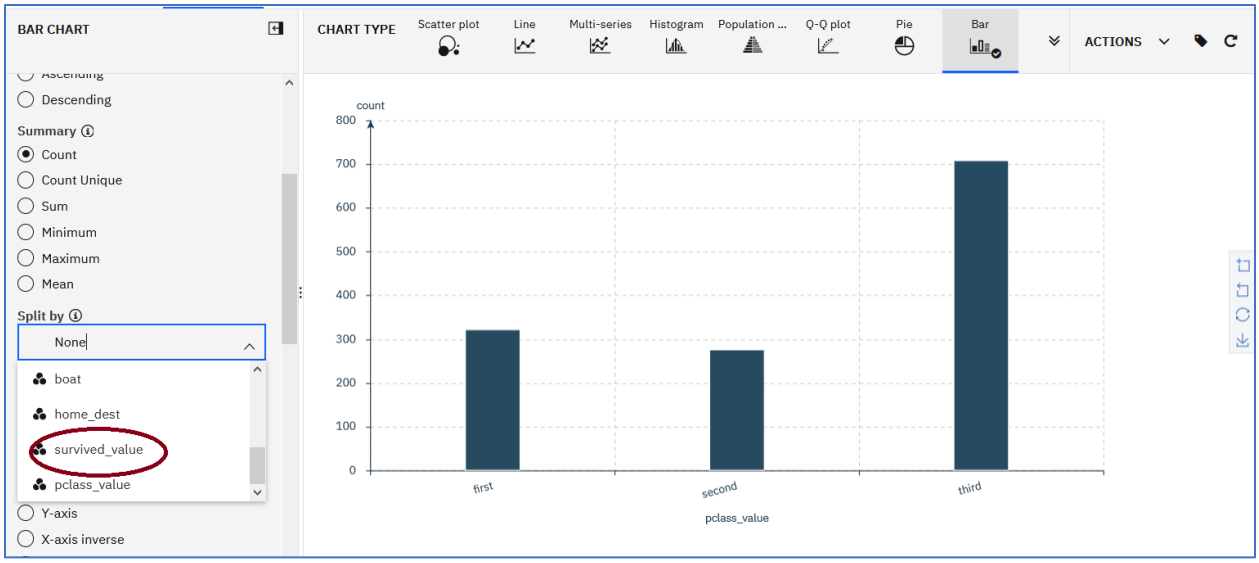
2. Let's take a look at the breakdown of passengers by passenger class. We will use our new pclass_value field. Select the **Bar** Chart Type.



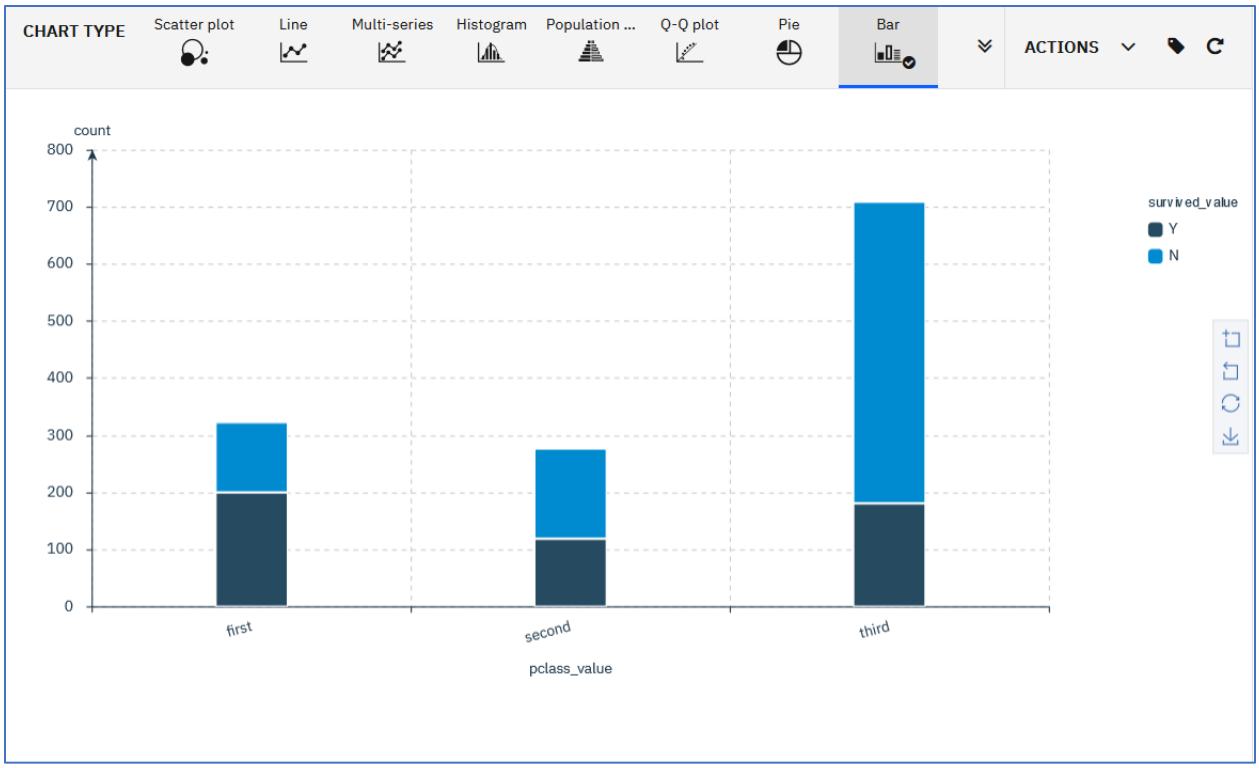
3. In the **Category** required field, select **pclass_value**.



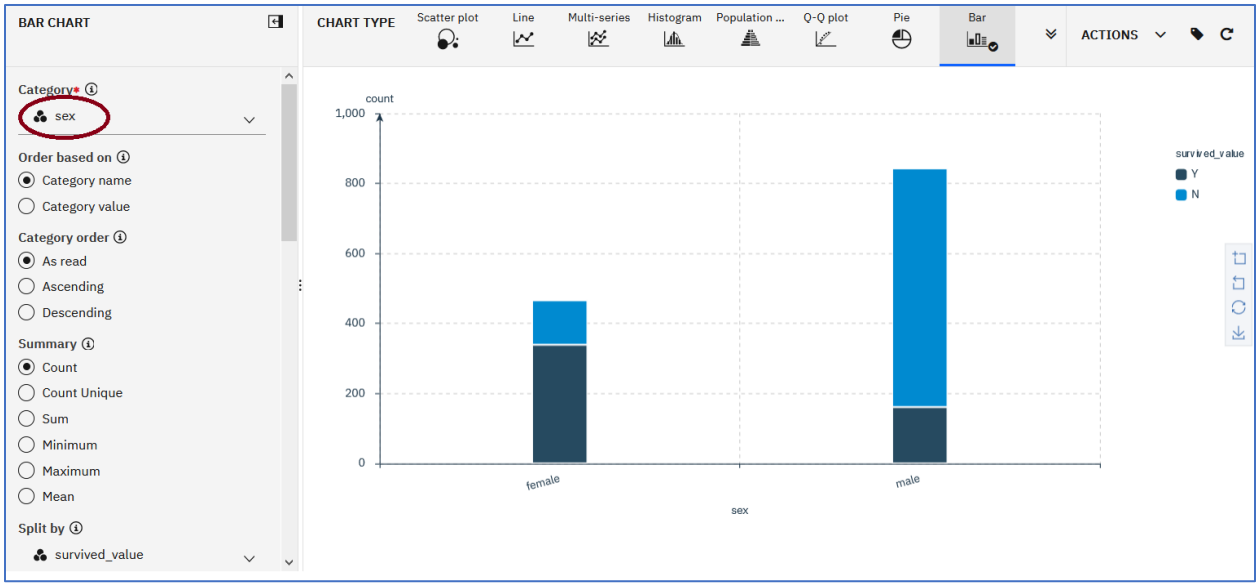
4. In the **Split by** field, select **survived_value**.



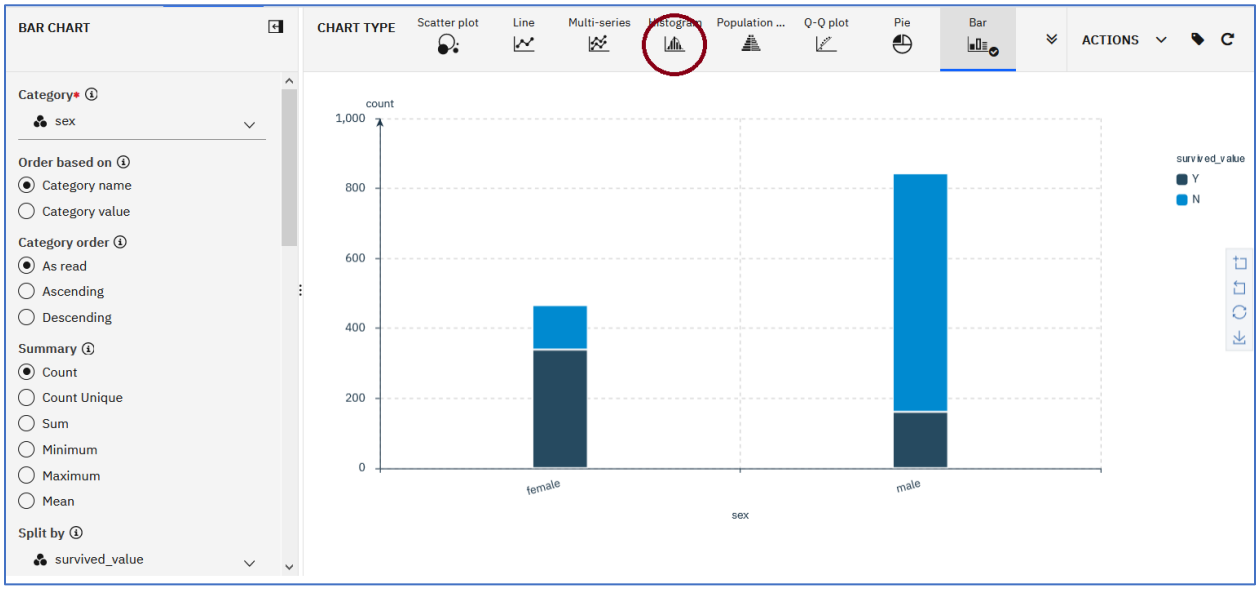
5. Select **Stacked** if not already selected. The percentage of survivor is the greatest in first-class, followed by second-class, and then third-class passengers.



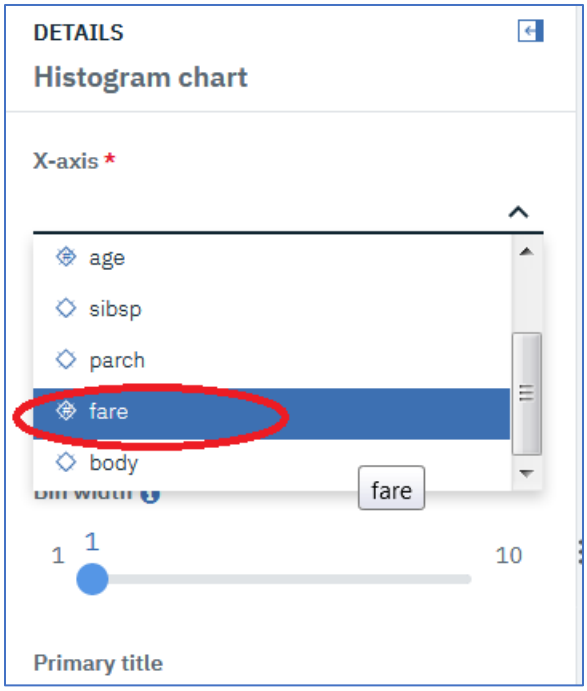
6. Change the **Category** to **sex**. We can see that survivorship for females is significantly greater than for males.



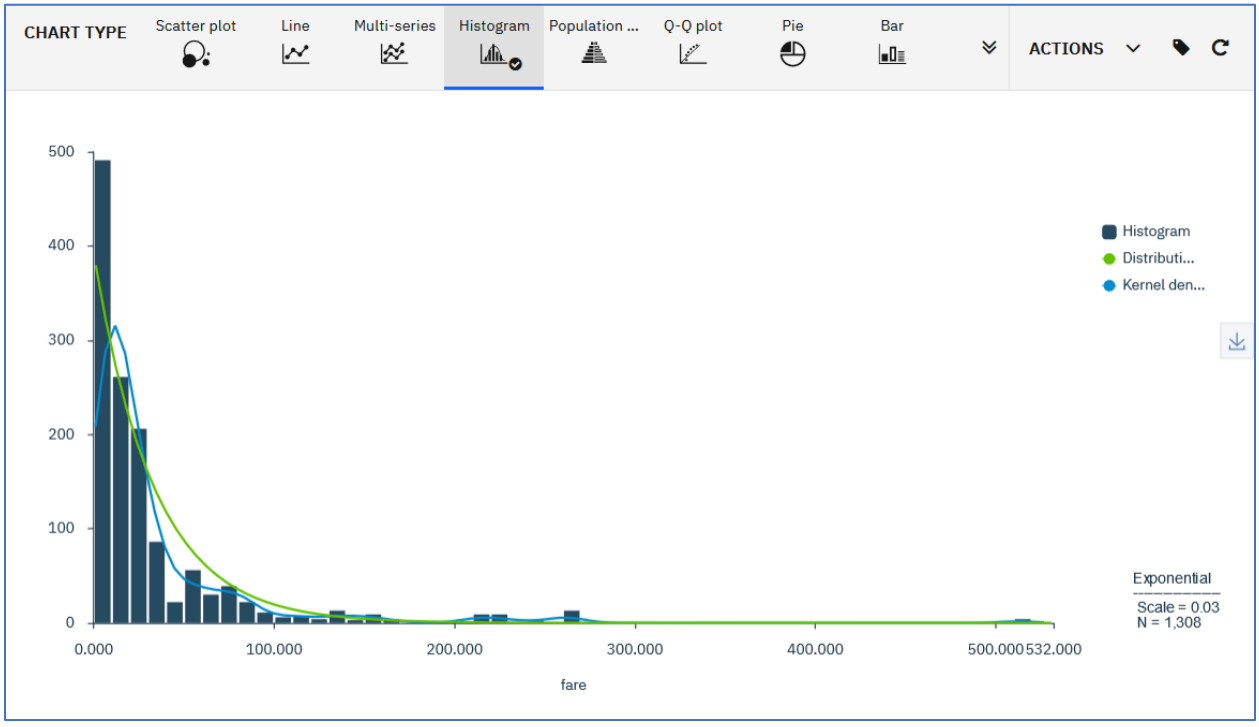
7. Click on the **Histogram** Chart Type.



8. Select **fare** for the X-axis. Select **None** for the Split by.



9. The result is shown below. Note that it is highly skewed which affects the performance of some machine learning algorithms. One way to deal with this is to apply a logarithmic transformation. We will do that as part of data preparation.



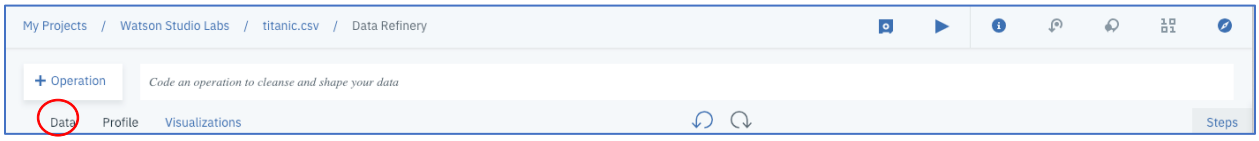
Step 4: Prepare the data for modeling

Based on the data analysis, we need to do the following to prepare the data for modeling.

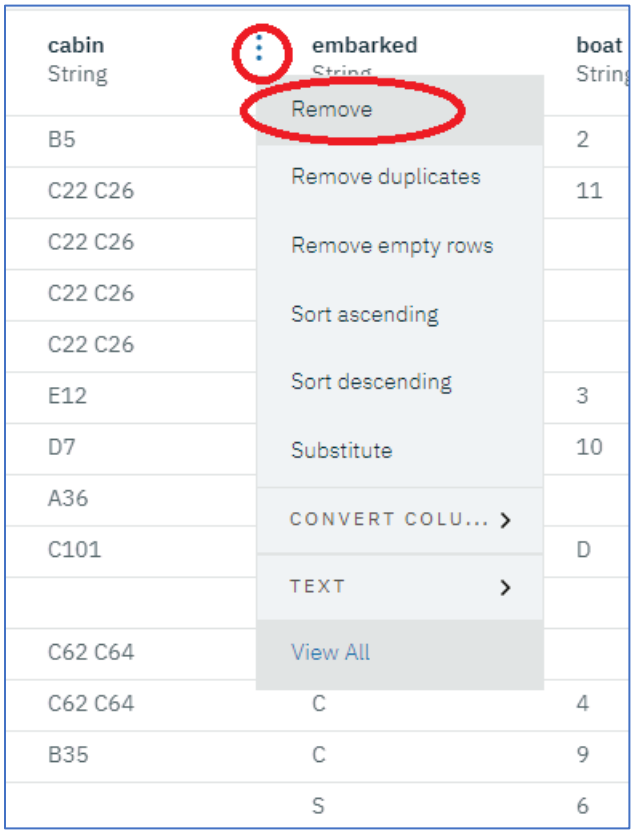
1. Remove columns cabin, boat, body, home.dest
2. Remove rows with missing values of age and embarked.
3. Create a new column(log_fare) that is the logarithm of the fare column

We will also bin the age, and log_fare fields.

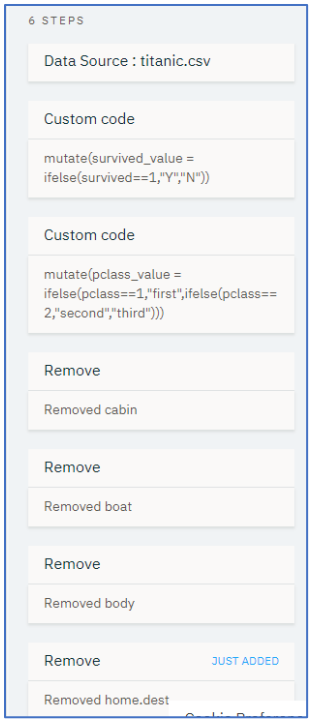
1. Return to the Data panel by clicking on the **Data** tab



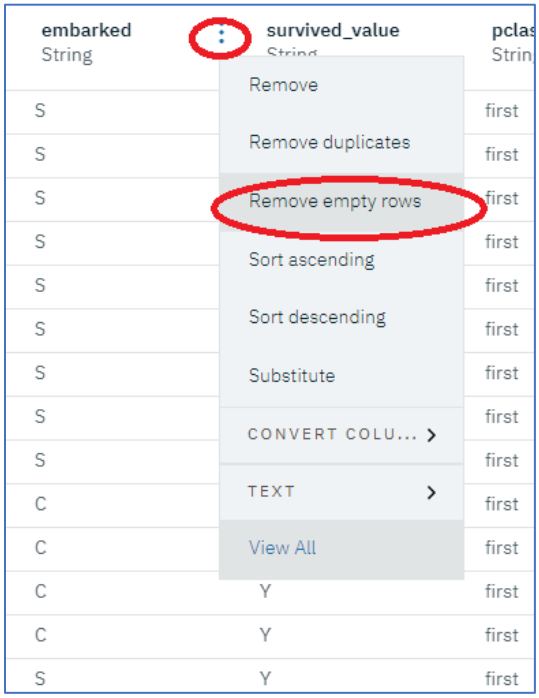
2. Remove the **cabin** column by selecting on the vertical ellipse adjacent to the cabin column and then clicking on **Remove**.



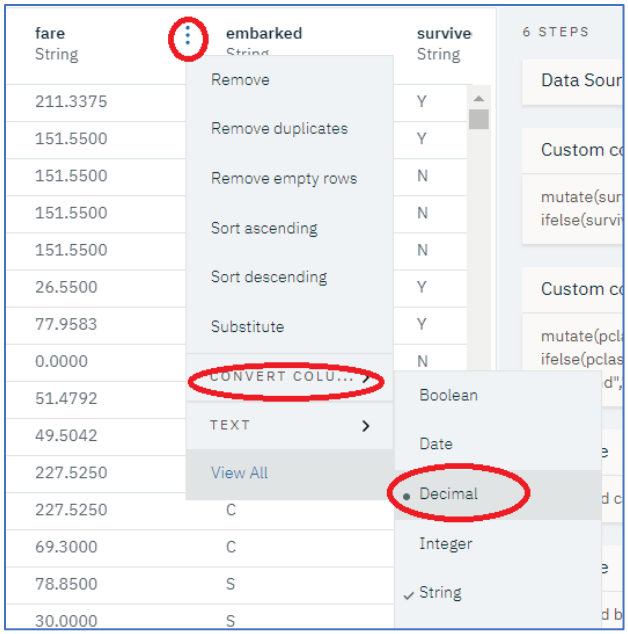
3. Remove the **boat**, **body**, and **home.dest** columns in a similar manner by selecting on the vertical ellipse adjacent to the column and clicking on **Remove**. Notice the STEPS panel on the right-hand side that provides a running list of the data operations.



4. For the **age** and **embarked** columns, click on the vertical ellipse adjacent to the columns, and click on **Remove empty rows**.



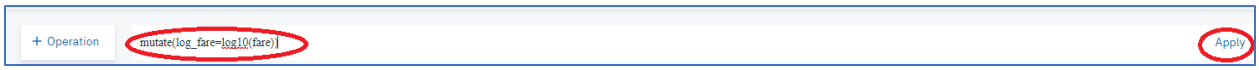
5. If the fare column is String, convert the **fare** column from a String to a Decimal by clicking on the vertical ellipse adjacent to the column, click on **Convert Column**, and then click on **Decimal**.



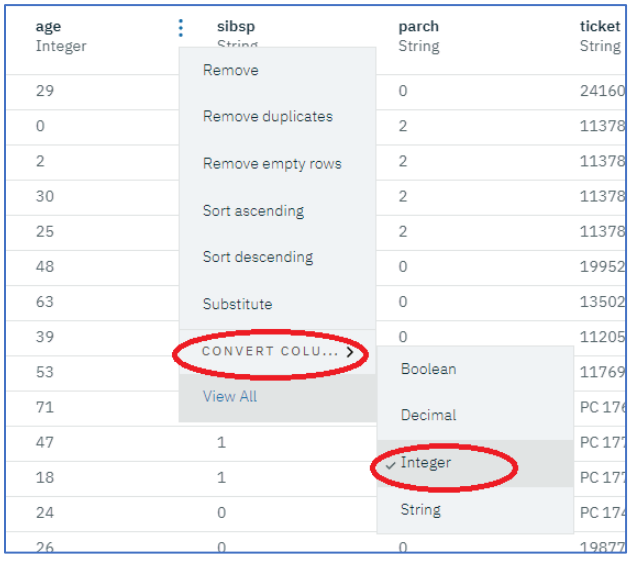
6. Create a new column that is the log to the base 10 of the fare by clicking into the **Code an operation to cleanse and shape your data,** and entering

   mutate(log_fare=log10(fare))

   then click **Apply**.



7. Convert the **age** from Decimal to Integer by clicking on the vertical ellipse adjacent to the age column, clicking on **Convert Column**, and clicking on **Integer**.

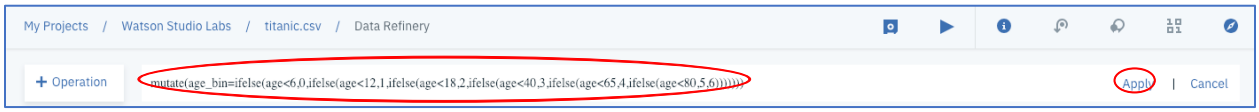


8. Bin the **age** column into the following bins by clicking into the **Code an operation to cleanse and shape your data,** and copying and pasting the following

mutate(age_bin=ifelse(age<6,0,ifelse(age<12,1,ifelse(age<18,2,ifelse(age<40,3,ifelse(age<65,4,ifelse(age<80,5,6)))))))

and then click **Apply**. Note, if this fails, it's because a line break has been inserted.

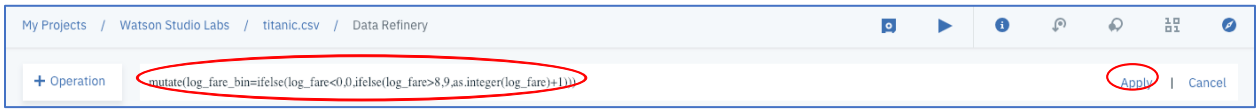
Bin	Age Range
0	0-5
1	6-11
2	12-17
3	18-39
4	40-64
5	65-79
6	Over 79



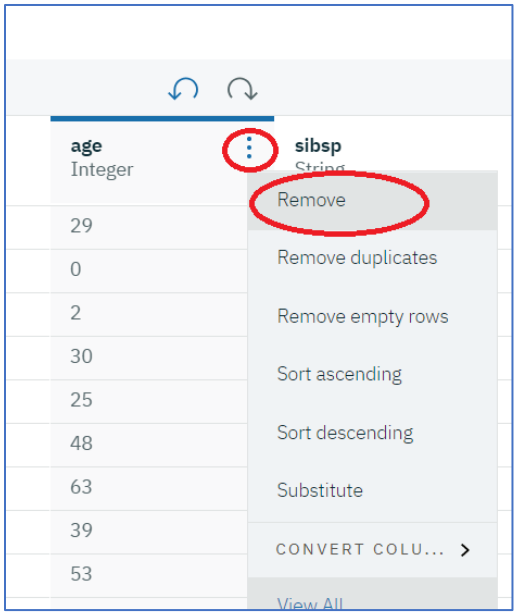
9. Bin the **log_fare** column, by clicking into the **Code an operation to cleanse and shape your data,** and copying and pasting the following

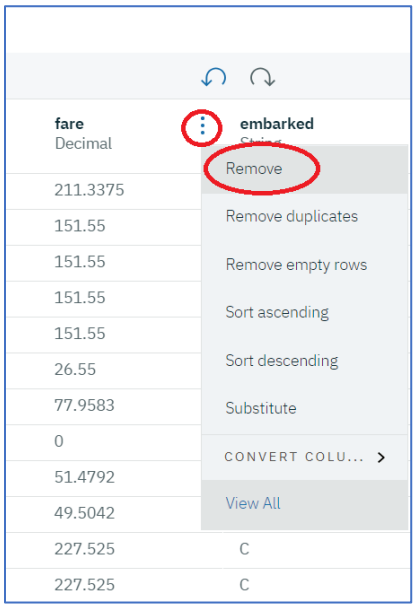
   mutate(log_fare_bin=ifelse(log_fare<0,0,ifelse(log_fare>8,9,as.integer(log_fare)+1)))

   and then clicking **Apply**.

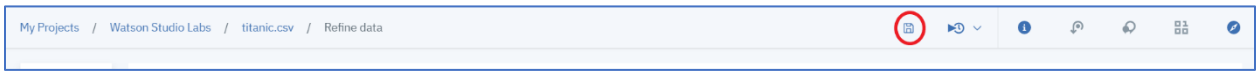


10. Now we will drop the **age**, **fare**, and **log_fare** columns as they are no longer needed for modeling purposes. Select the vertical ellipse adjacent to the column and click on **Remove** as shown below.


age
Integer
sibsp
Remove
Remove duplicates
Remove empty rows
Sort ascending
Sort descending
Substitute
CONVERT COLU...
View All
29
0
2
30
25
48
63
39
53

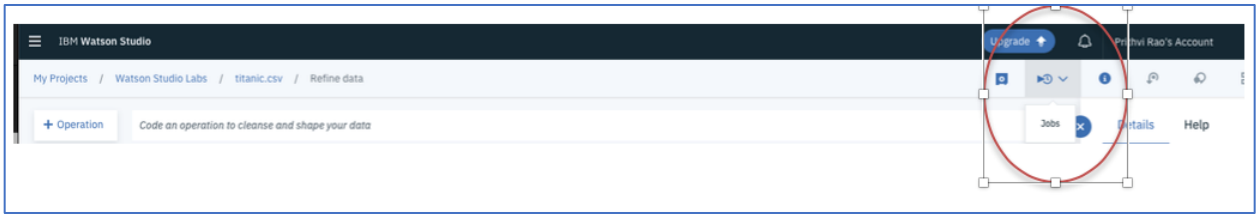

fare
Decimal
embarked
Remove
Remove duplicates
Remove empty rows
Sort ascending
Sort descending
Substitute
CONVERT COLU...
View All
211.3375
151.55
151.55
151.55
151.55
26.55
77.9583
0
51.4792
49.5042
227.525
C
227.525
C

11. Save the Data Flow by clicking on the Save Data Flow icon .

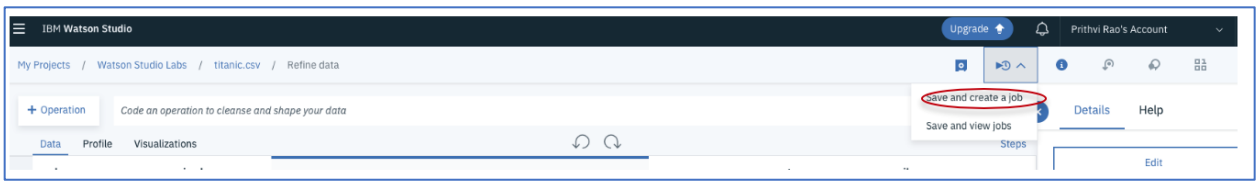


Step 5: Run the sequence of Data Flow operations on the entire data set.

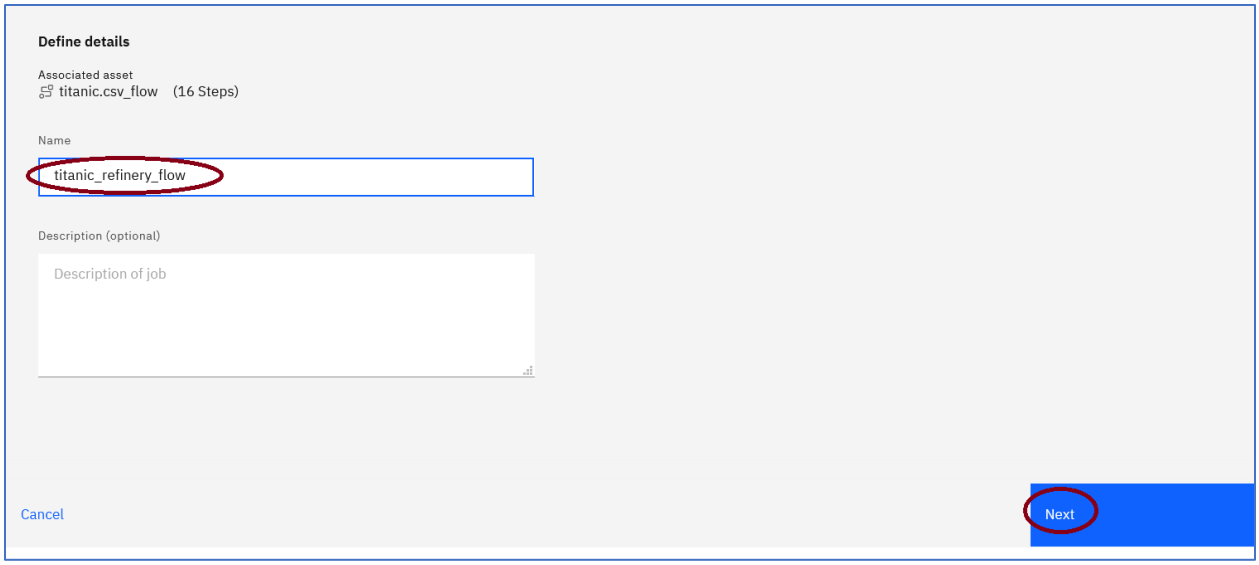
1. When users are interacting with the Data Refinery tool, the operations are applied to a subset of the data set to facilitate faster response times. To run the data operations on the entire data set, the user selects the **Jobs** icon .



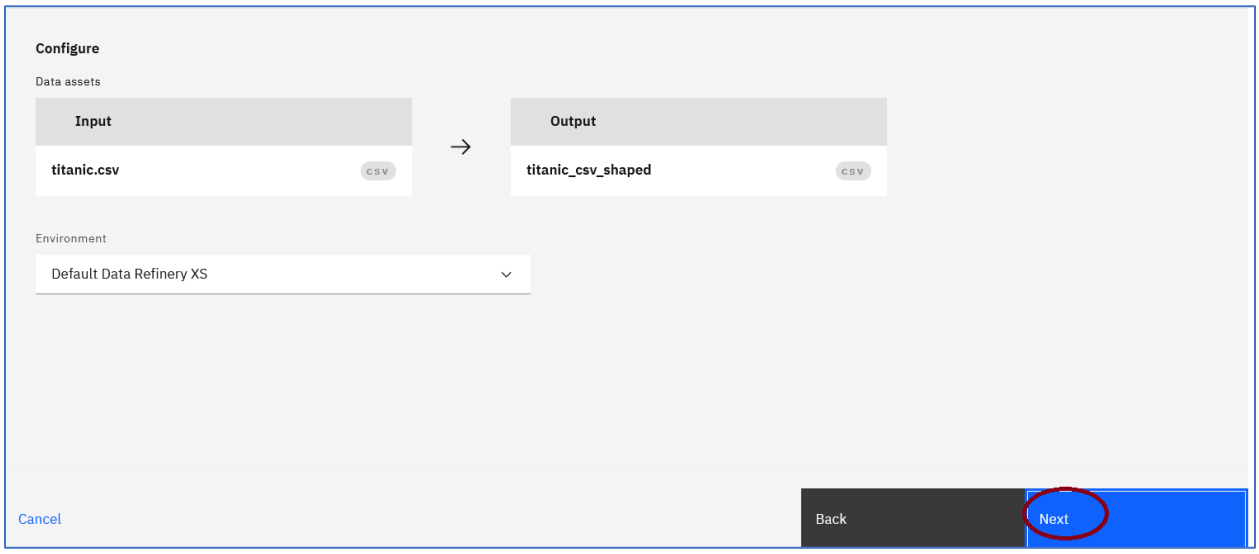
2. Selecting the **Jobs** icon, results in a drop down, select **Save and create a job**



3. This action results in the following page display. Fill in the **Name**, for example **titantic_refinery_flow**, and click on the **Next** button.

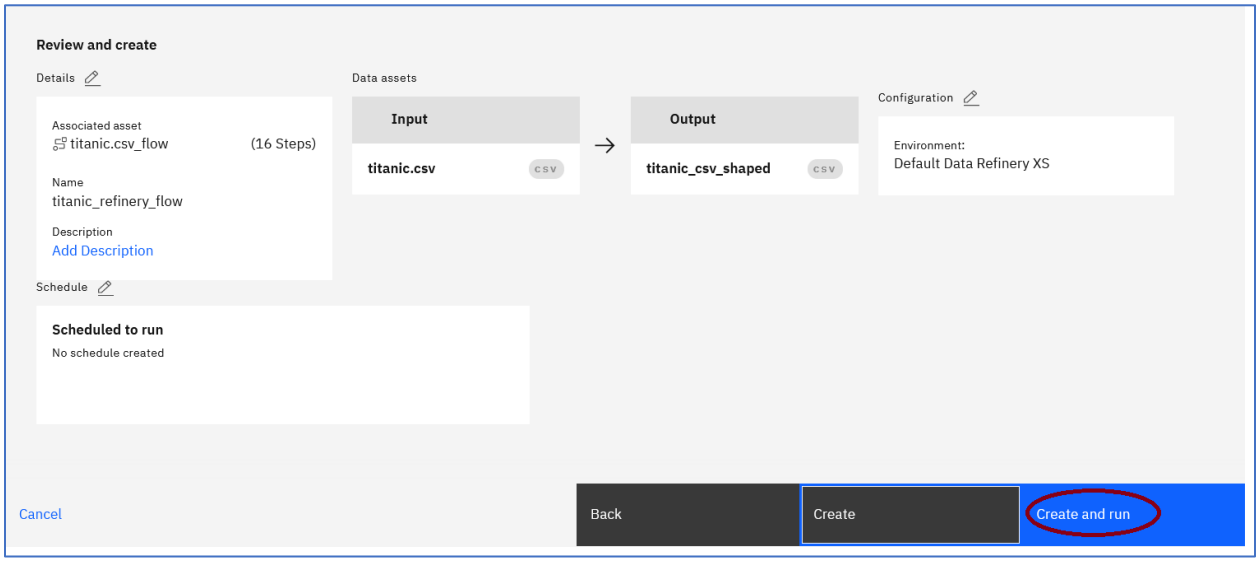


4. Keep the default Runtime, by clicking the **Next** button on the **Configure** panel.

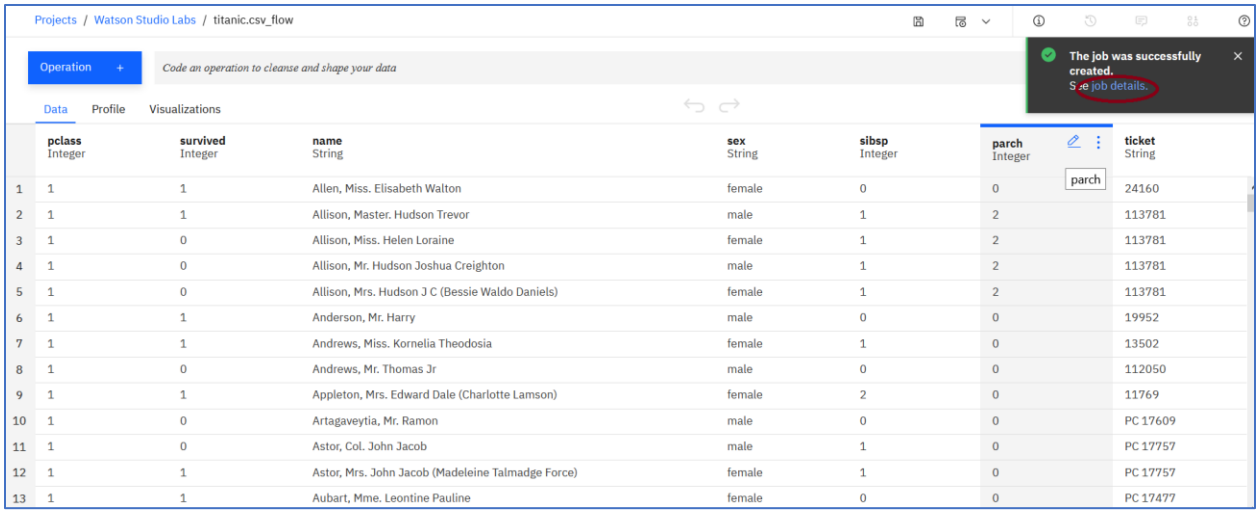


5. A schedule can be set up if the transformation process needs to run on a scheduled basis. We will run the job immediately. So click the **Next** button on the **Schedule** panel.

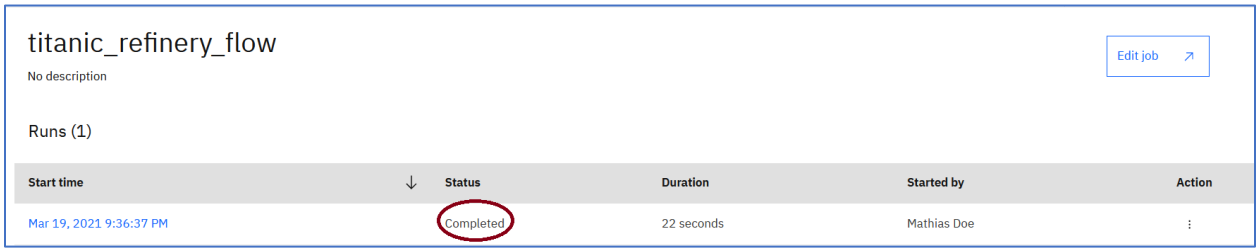
6. Review the job parameters, and then click **Create and run**.



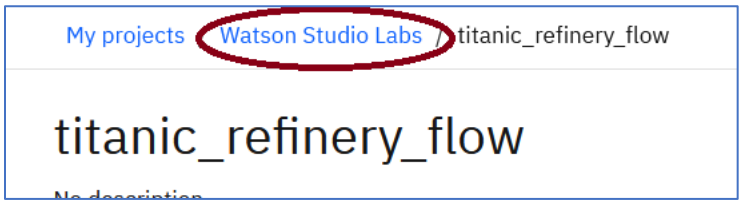
7. The display returns to the Data Refinery view and a status message is displayed that the job is submitted. Click on the **job details** link.



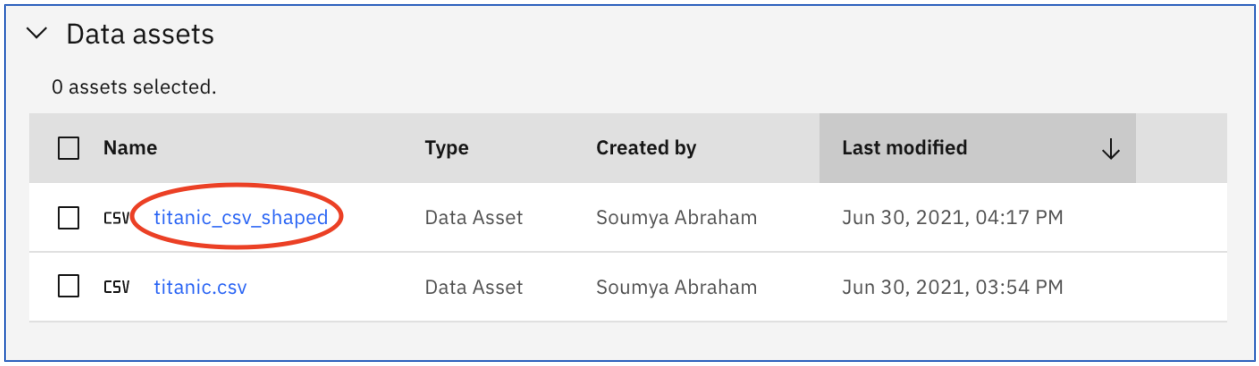
8. The status of the job will go from **Running** to **Completed**.



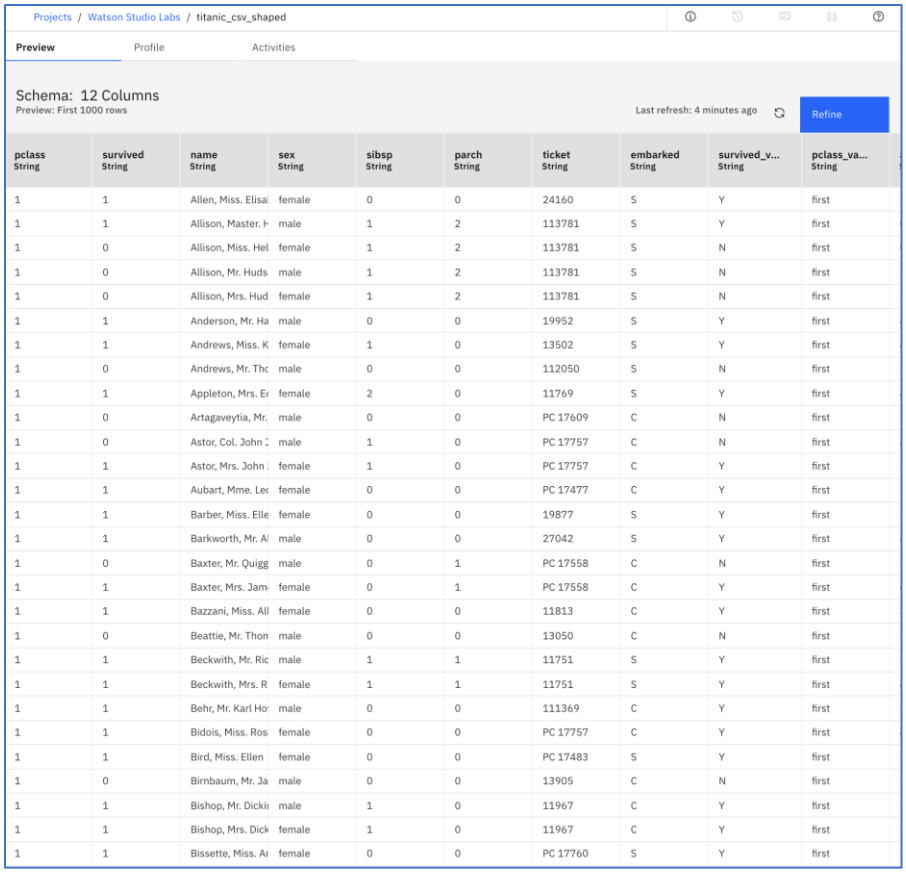
9. The output of the Data Refinery process is listed in the Data Assets. Click on **Watson Studio Labs**



10. Click on **titanic_csv_shaped** to view the asset contents.



11. The asset contents are displayed below. Review to confirm that the data transformations specified have been applied to all the data.

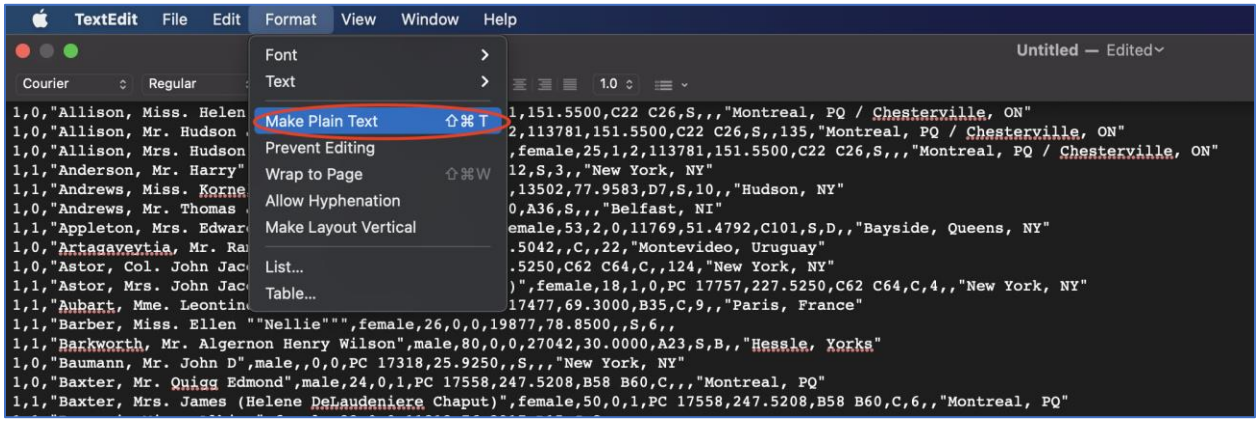


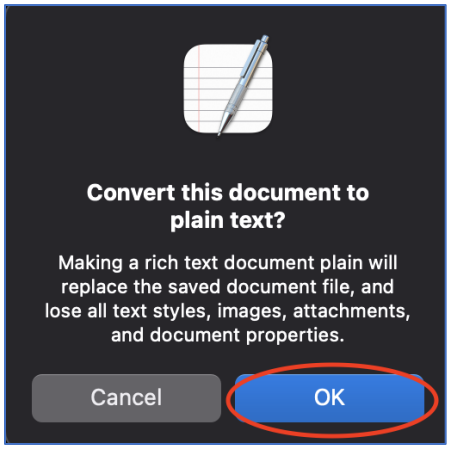
You have completed the Lab !!!

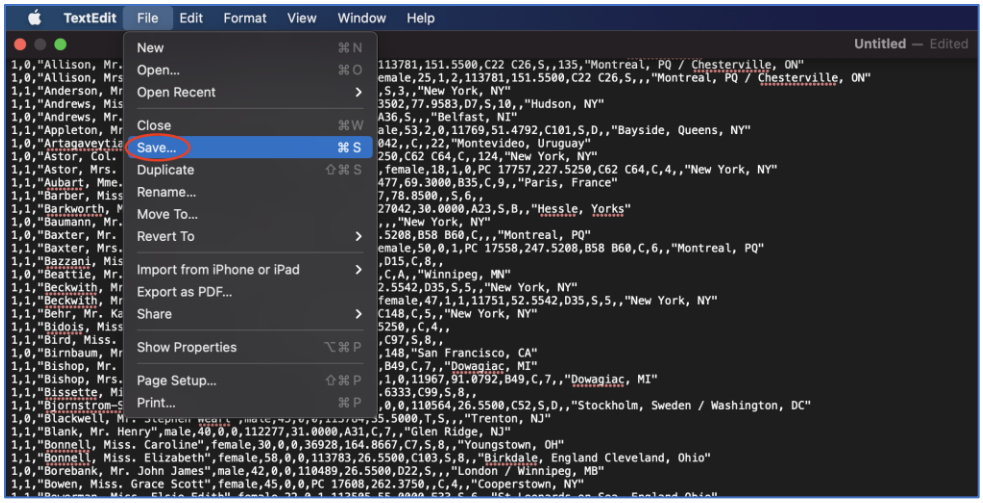
- ✓ Profiled the data to help determine missing values
- ✓ Visualized the data to gain a better understanding
- ✓ Prepared the data for modeling
- ✓ Ran the sequence of data preparation operations on the entire data set.
- ✓ Verified the output data asset.

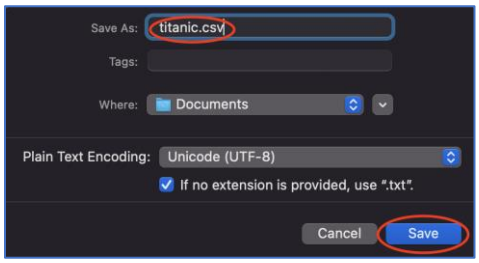
Appendix: Steps on Mac

If you are unable to use the method above to save the file, you'll need to copy the contents of the file into an text editor, such as TextEdit. You can copy all of the content easily by pressing **Command** + **A** and then going over to TextEdit and pressing **Command** + **V**. Once you have copied the content into the TextEdit, you will then need to go to the toolbar, click **Format** and then select **Make Plain Text**. Afterwards, return to TextEdit's toolbar and save the file as **titanic.csv**









Return to Step 4.