

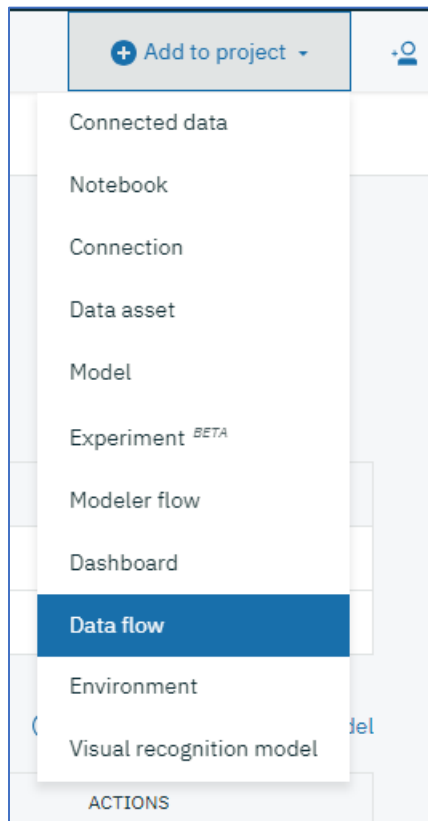
Data Refinery Lab

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:

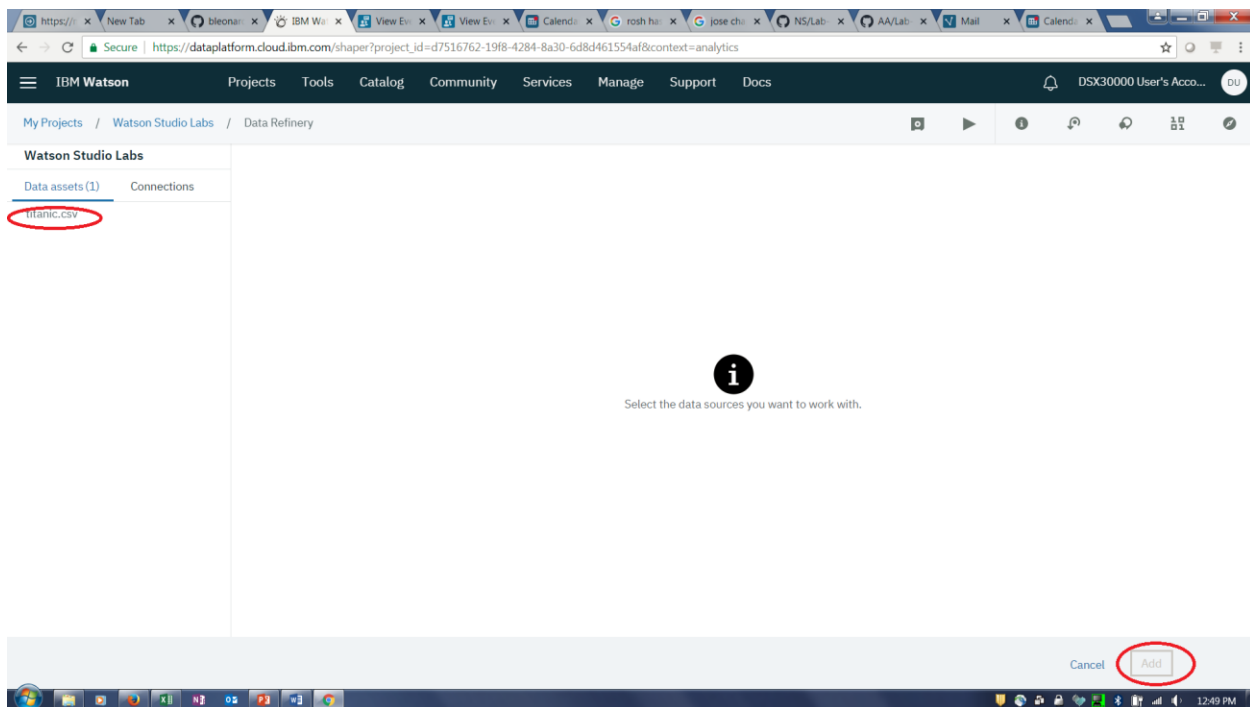
1. Use the Data Refinery Tool to:
 - a. Profile the data to help determine missing values
 - b. Visualize the data to gain a better understanding
 - c. Prepare the data for modeling
 - d. Run the sequence of data preparation operations on the entire data set.

Step 1: Profile the data to help determine missing values.

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



2. Select **titanic.csv** and then click on **Add**.

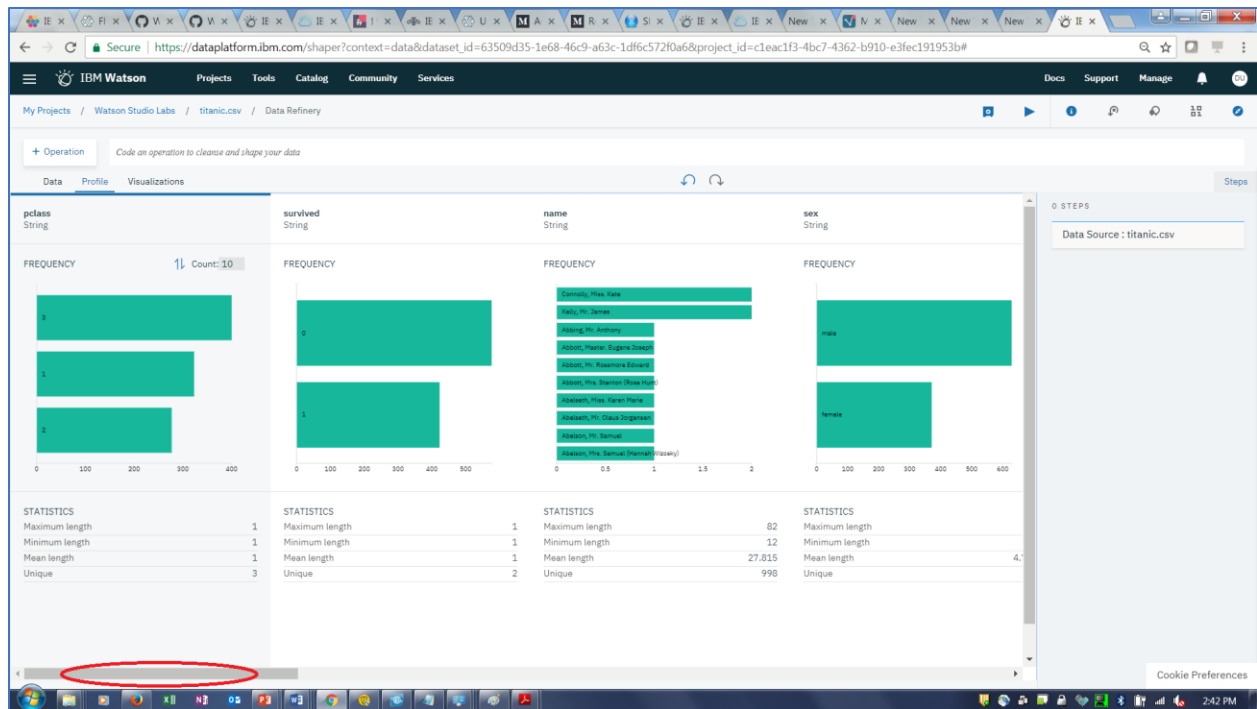


3. The Data Refinery panel will display the Titanic data set. Click on the **Profile** tab.

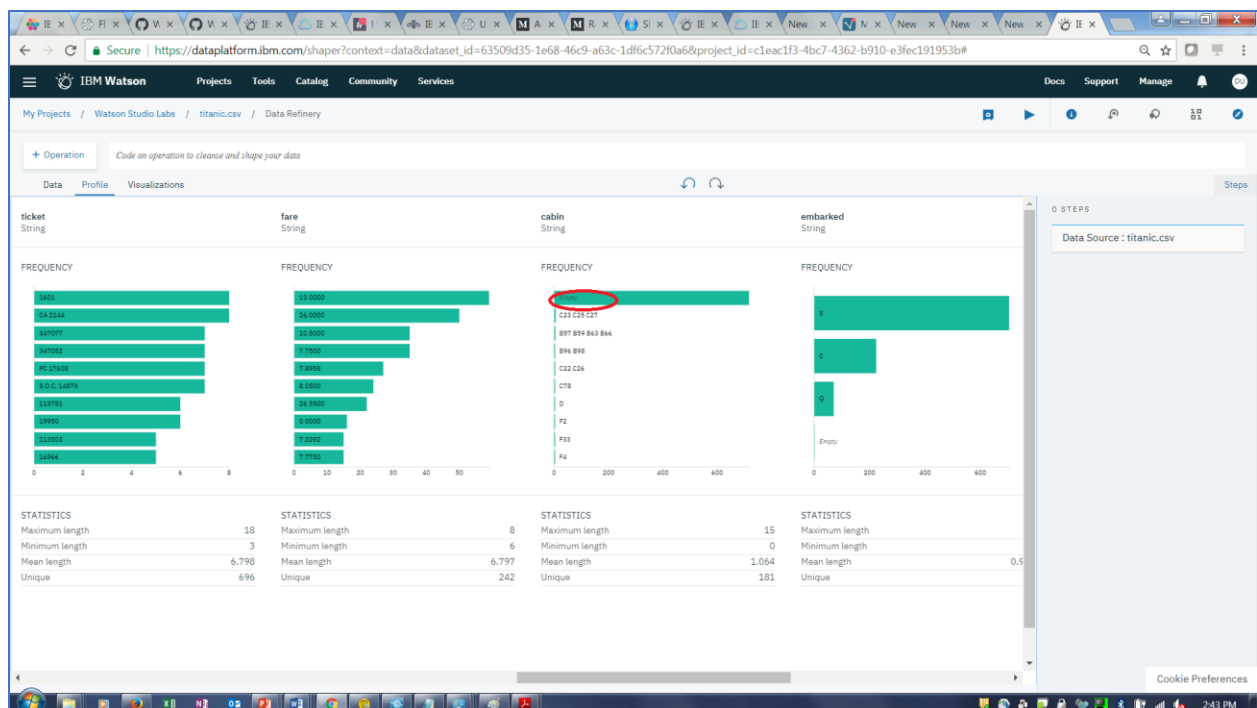
The screenshot shows the IBM Watson Data Platform interface with the 'Data Refinery' panel. The 'Profile' tab is selected, and the 'Data Source' is set to 'titanic.csv'. The table displays the Titanic dataset with columns: pclass, survived, name, sex, age, sibsp, parch, ticket, fare, and cal. The 'Profile' tab is highlighted, and the 'Data Source' is set to 'titanic.csv'. The table shows the first 20 rows of data, including passenger names, survival status, age, sex, family size, ticket number, fare, and cabin number.

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	cal
1	1	1	Allen, Miss. Elisabeth...	female	29	0	0	24160	211.3375	B1...
2	1	1	Allison, Master. Hud...	male	0.9167	1	2	113781	151.5500	C2...
3	1	0	Allison, Miss. Helen ...	female	2	1	2	113781	151.5500	C2...
4	1	0	Allison, Mr. Hudson ...	male	30	1	2	113781	151.5500	C2...
5	1	0	Allison, Mrs. Hudso...	female	25	1	2	113781	151.5500	C2...
6	1	1	Anderson, Mr. Harry	male	48	0	0	19952	26.5500	E1...
7	1	1	Andrews, Miss. Korn...	female	63	1	0	13502	77.9583	D1...
8	1	0	Andrews, Mr. Thom...	male	39	0	0	112050	0.0000	A3...
9	1	1	Appleton, Mrs. Edw...	female	53	2	0	11769	51.4792	C1...
10	1	0	Artagaveytia, Mr. Ra...	male	71	0	0	PC 17609	49.5042	B1...
11	1	0	Astor, Col. John Jacob	male	47	1	0	PC 17757	227.5250	C6...
12	1	1	Astor, Mrs. John Jac...	female	18	1	0	PC 17757	227.5250	C6...
13	1	1	Aubart, Mme. LeontL...	female	24	0	0	PC 17477	69.3000	B1...
14	1	1	Barber, Miss. Ellen "...	female	26	0	0	19877	78.8500	B1...
15	1	1	Barkworth, Mr. Alge...	male	80	0	0	27042	30.0000	A3...
16	1	0	Baumann, Mr. John D	male	0	0	0	PC 17318	25.9250	B1...
17	1	0	Baxter, Mr. Quigg Ed...	male	24	0	1	PC 17558	247.5208	B1...
18	1	1	Baxter, Mrs. James (...)	female	50	0	1	PC 17558	247.5208	B1...
19	1	1	Bazzani, Miss. Albina	female	32	0	0	11813	76.2917	D1...
20	1	0	Beattie, Mr. Thomson	male	36	0	0	13050	75.2417	C6...

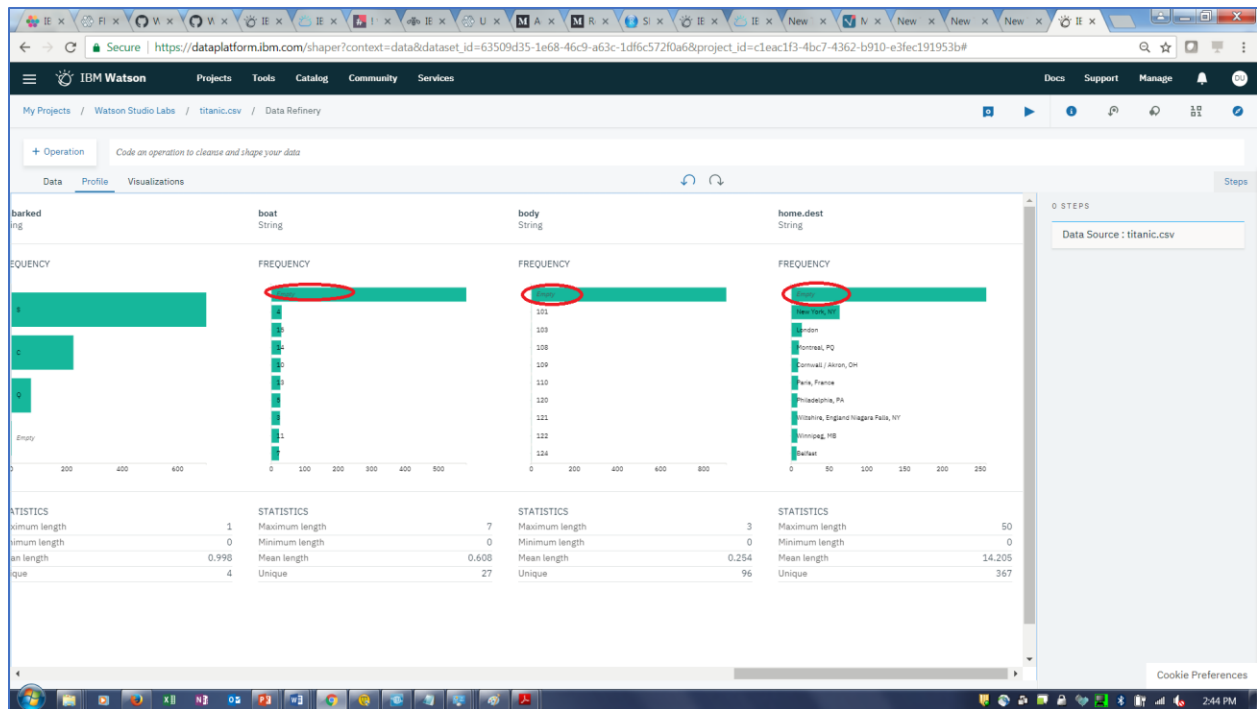
4. The Profile panel displays the counts of the top 10 count values for each column. Note that you can change 10 to another number if desired. You can also switch to the bottom 10 counts for a column. Scroll to the right to view the cabin column.



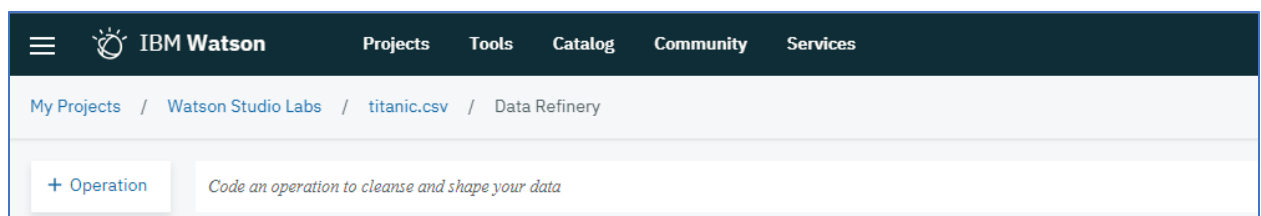
- Note that the cabin column has many missing values and should be removed as part of the data preparation step.



- 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.

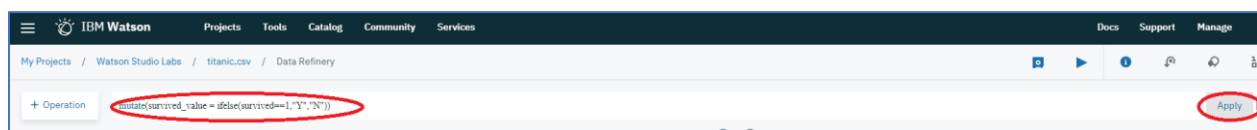


7. Age and Embarked also have missing values. Embarked has very few missing values. Age has over 100 missing values, but we will keep that column in the analysis. As part of data preparation, we will remove the rows that contain the missing age and embarked values.
8. Click on the **Data** tab. We will 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 dplyr function) and ifelse functions to do the conversion. Click on the **Code an operation to cleanse and shape your data**.

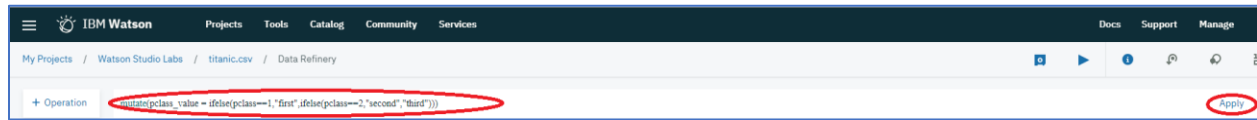


9. Type the following:
`mutate(survived_value=ifelse(survived==1, "Y", "N"))`

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



10. Type the following to create pclass_value,
`mutate(pclass_value=ifelse(pclass==1,"first",ifelse(pclass==2,"second","third")))`



11. The result is shown below. Notice that the right panel will contain a running list of the transformations.

My Projects / Watson Studio Labs / titanic.csv / Data Refinery

+ Operation `Code an operation to cleanse and shape your data`

Data Profile Visualizations

	ticket	fare	cabin	embarked	boat	body	home.dest	survived_value	pclass_value
	String	String	String	String	String	String	String	String	String
1	24160	211.3375	B5	S	2		St Louis, MO	Y	first
2	113781	151.5500	C22 C26	S	11		Montreal, PQ / Ches...	Y	first
3	113781	151.5500	C22 C26	S			Montreal, PQ / Ches...	N	first
4	113781	151.5500	C22 C26	S		135	Montreal, PQ / Ches...	N	first
5	113781	151.5500	C22 C26	S			Montreal, PQ / Ches...	N	first
6	19952	26.5500	E12	S	3		New York, NY	Y	first
7	13502	77.9583	D7	S	10		Hudson, NY	Y	first
8	112050	0.0000	A36	S			Belfast, NI	N	first
9	11769	51.4792	C101	S	D		Bayside, Queens, NY	Y	first
10	PC 17609	49.5042		C		22	Montevideo, Uruguay	N	first
11	PC 17757	227.5250	C62 C64	C		124	New York, NY	N	first
12	PC 17757	227.5250	C62 C64	C	4		New York, NY	Y	first

2 STEPS

Data Source : titanic.csv

Custom code

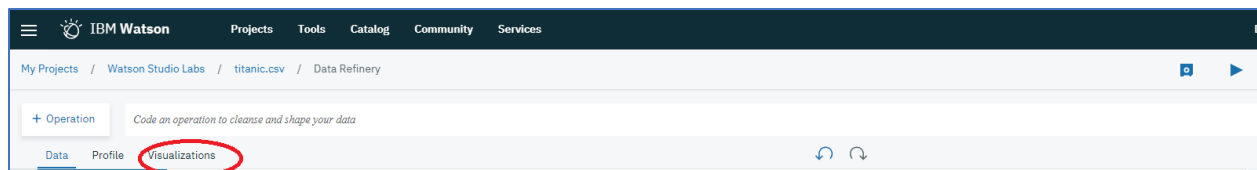
```
mutate(survived_value = ifelse(survived==1,"Y","N"))
```

Custom code JUST ADDED

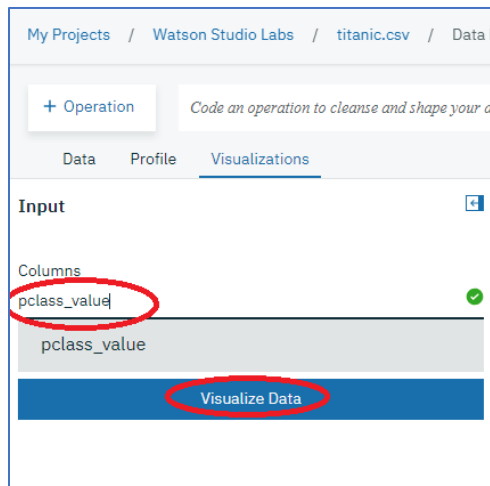
```
mutate(pclass_value = ifelse(pclass==1,"first",ifelse(pclass==2,"second","third"))
```

Step 3: Visualize the data to get a better understanding

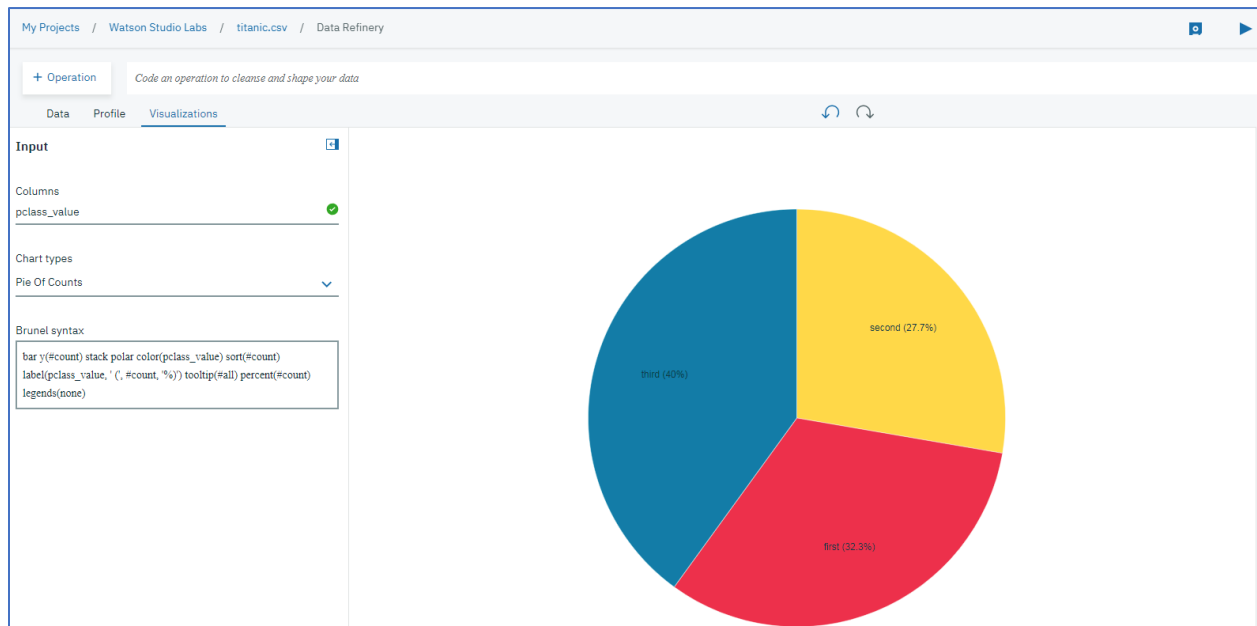
1. Click on the **Visualizations** tab.



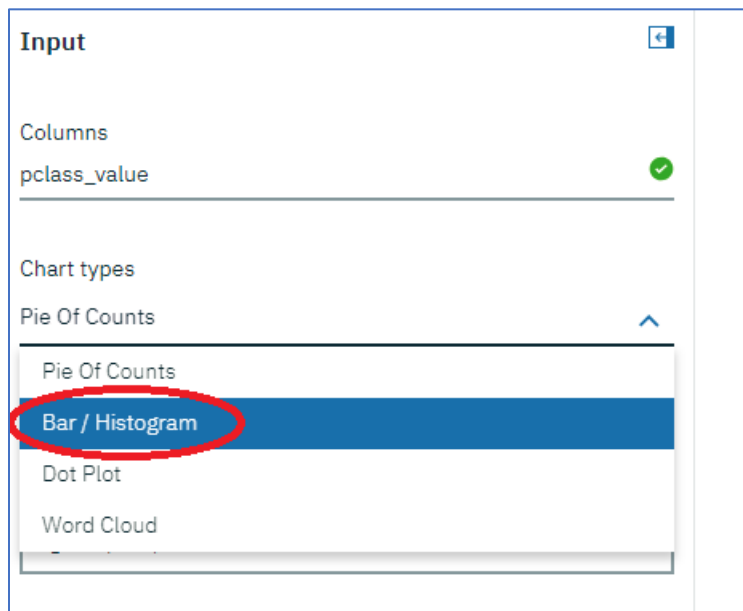
2. Let's take a look at the breakdown of passengers by passenger class. We will use our new pclass_value field. Enter or select pclass_value and then click **Visualize Data**



3. The result is shown below.

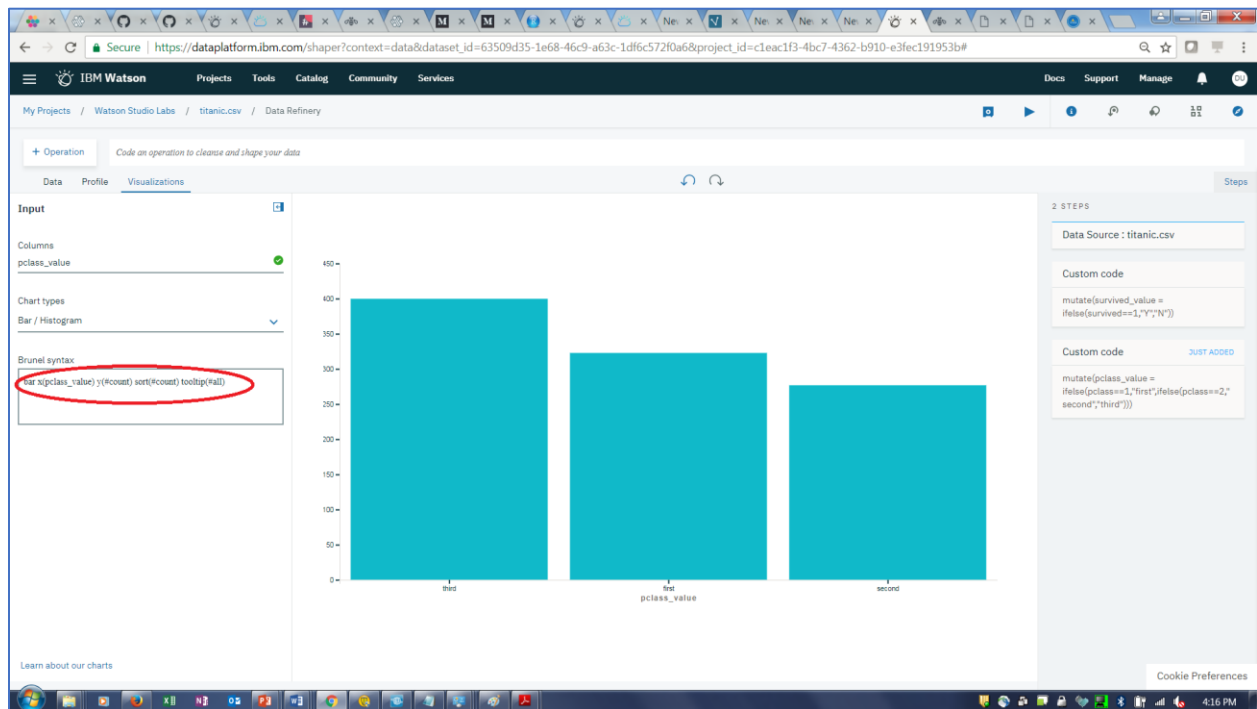


4. We can switch this to a bar chart, by switching the Chart type.

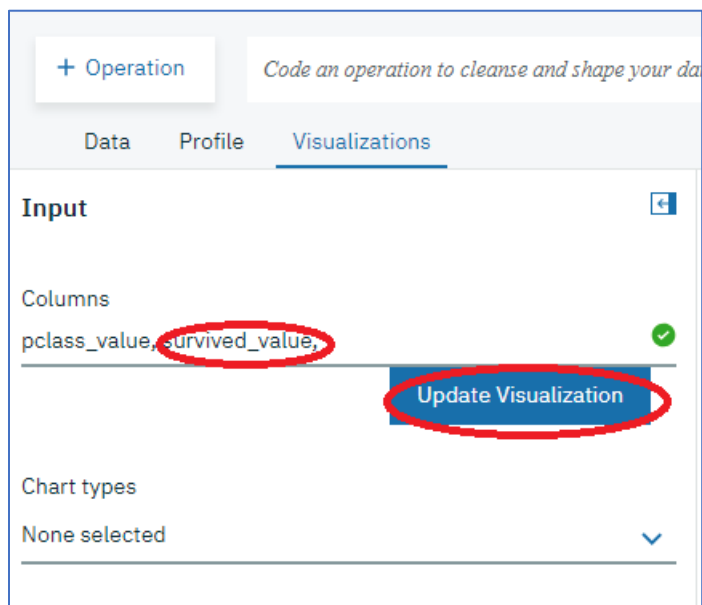


5. The result is shown below. Note the Brunel coding syntax. According to the Brunel github repo, *Brunel defines a highly succinct and novel language that defines interactive data visualizations based on tabular data. The language is well suited for both data scientists and more aggressive business users. The system interprets the language and produces visualizations using the user's choice of existing lower-level visualization technologies typically used by application engineers such as RAVE or D3. It can operate stand-alone and integrated into Jupyter (IPython) notebooks with further integrations as well as other low-level rendering support depending on the desires of the community.*

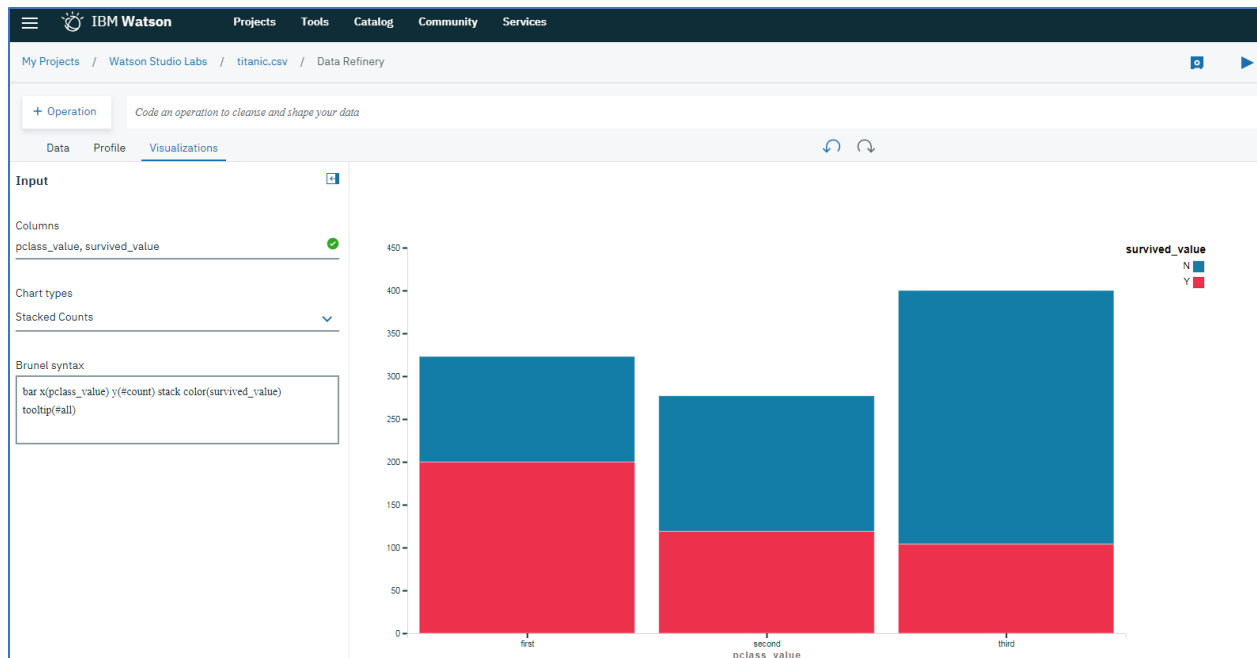
If you understand the syntax, you can make changes and update the visualization.



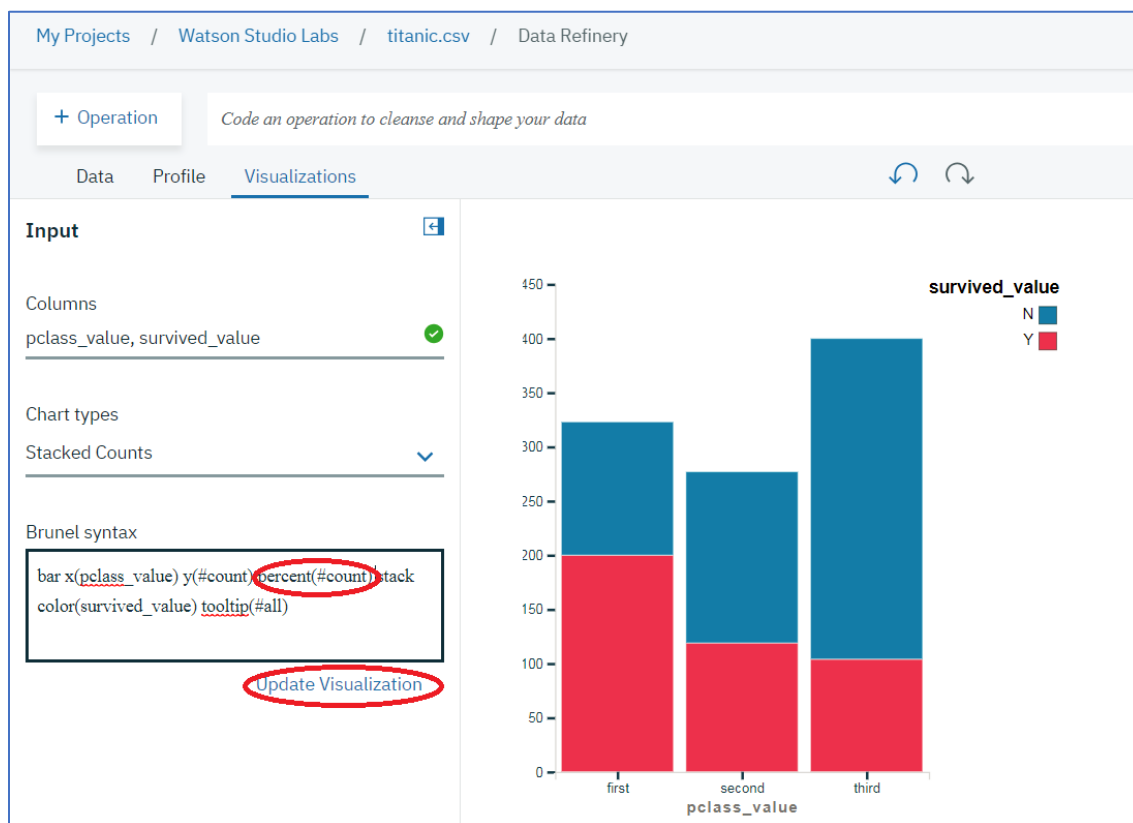
- Let's examine the relationship between survival and the passenger class. We will add the survived_value and click **Update Visualization**.



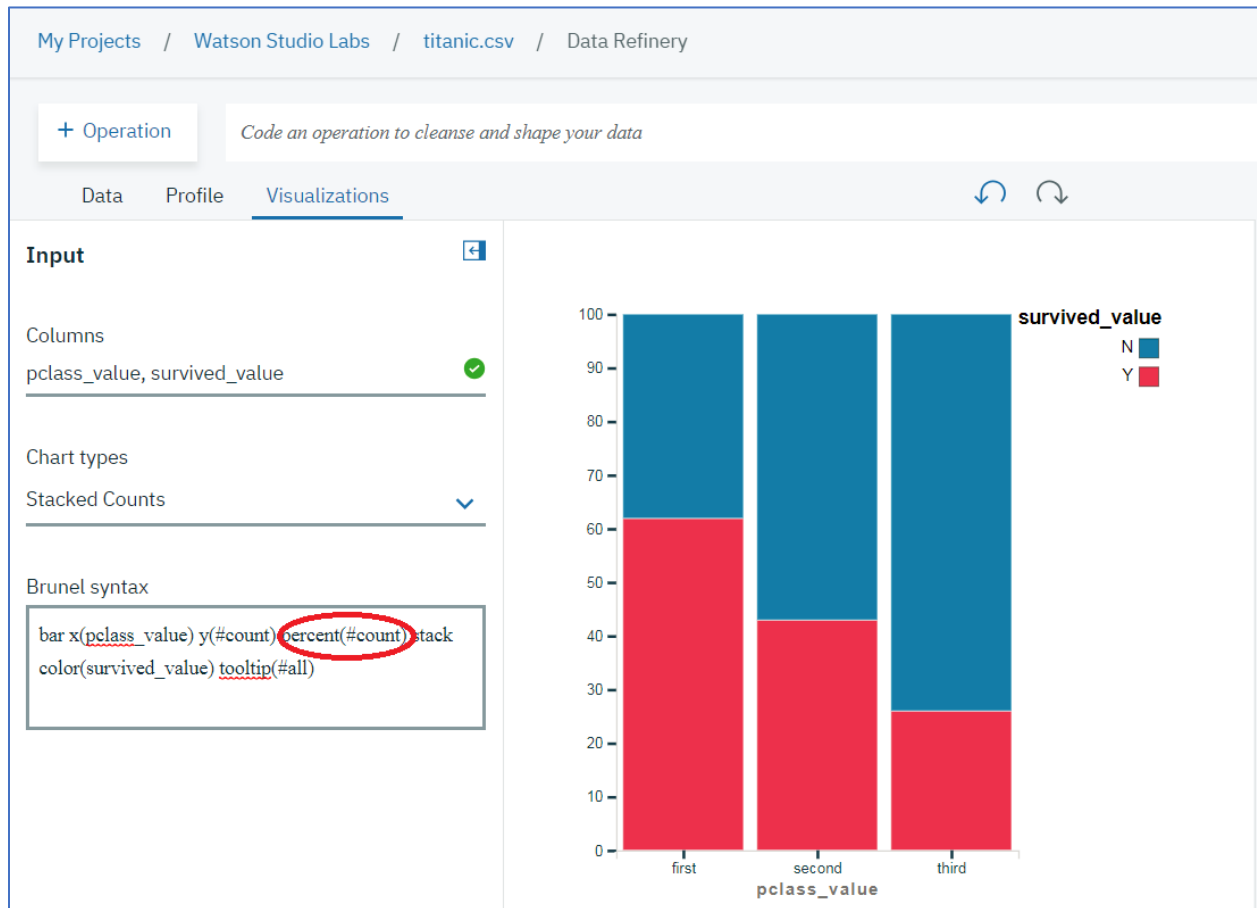
- The result is shown below. We can see that survival probability for first class customers is significantly better.



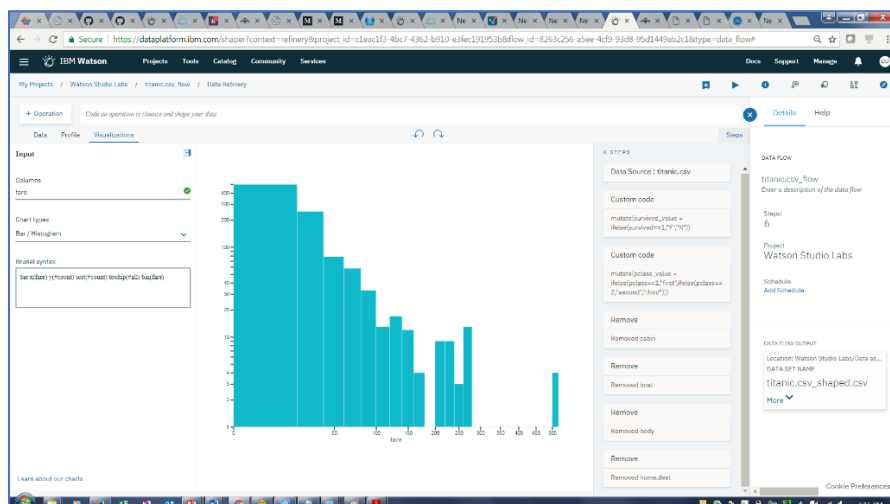
8. If we want to normalize the results so that each column is shown as a percentage to allow comparisons more easily, then add **percent(#count)** to the Brunel syntax and click on **Update Visualization**.



9. The result is shown below. We can see that the percentage of survival is greatest for first class and lowest for third class.



10. Plot the fare values. 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.



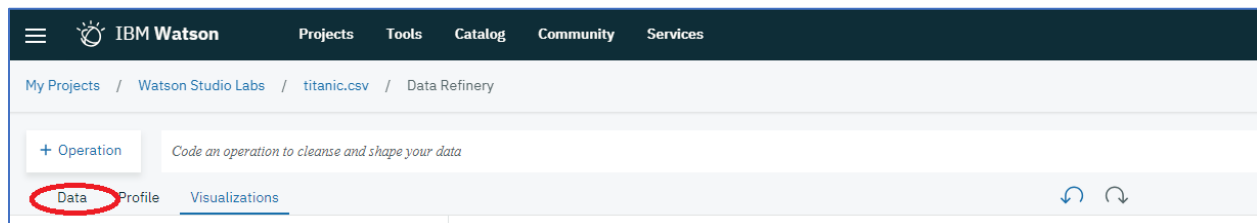
Step 4: Prepare the data for modeling

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.

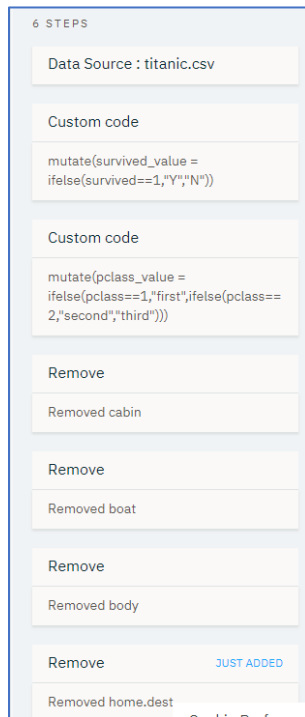
1. Return to the Data panel by clicking on the **Data** tab



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

cabin String	embarked String	boat String
B5		2
C22 C26		11
C22 C26		
C22 C26		
C22 C26		
E12		3
D7		10
A36		
C101		D
C62 C64		
C62 C64	C	4
B35	C	9
	S	6

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**.

embarked	survived_value	pclass
String	String	String
S		first
S		first
S		first
S		first
S		first
S		first
S		first
S		first
C		first
C		first
C	Y	first
C	Y	first
S	Y	first

- 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**.

fare	embarked	survived_value	pclass
String	String	String	String
211.3375		Y	
151.5500		Y	
151.5500		N	
151.5500		N	
151.5500		N	
26.5500		Y	
77.9583		Y	
0.0000		N	
51.4792			
49.5042			
227.5250			
227.5250	C		
69.3000	C		
78.8500	S		
30.0000	S		

- 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**.

+ Operation
mutate(log_fare=log10(fare))
Apply

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

age	sibsp	parch	ticket
Integer	String	String	String
29		0	24160
0		2	11378
2		2	11378
30		2	11378
25		2	11378
48		0	19952
63		0	13502
39		0	11205
53			11769
71			PC 176
47	1		PC 177
18	1		PC 177
24	0		PC 174
26	0	0	19877

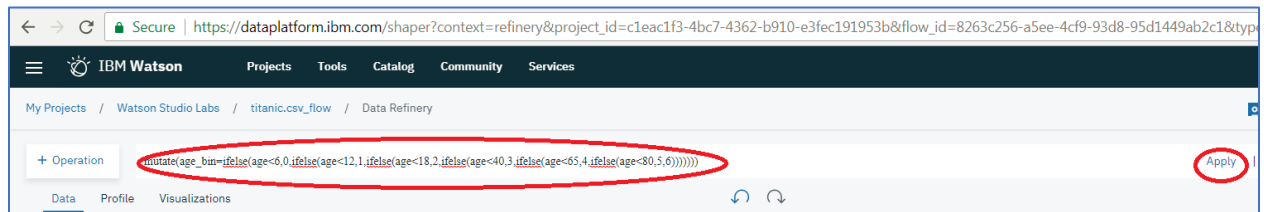
- Bin the age column into the following bins by clicking into the **Code an operation to cleanse and shape your data**, and entering

```
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**.

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

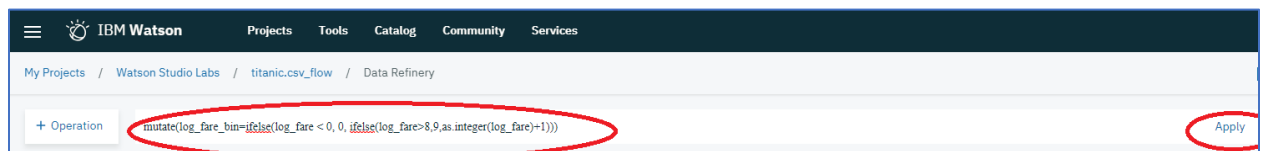
--	--



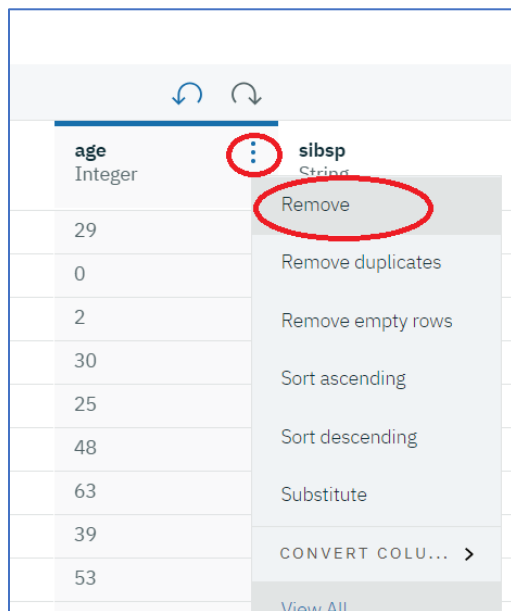
- Bin the `log_fare` column, by clicking into the **Code an operation to cleanse and shape your data**, and entering

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

and then clicking **Apply**




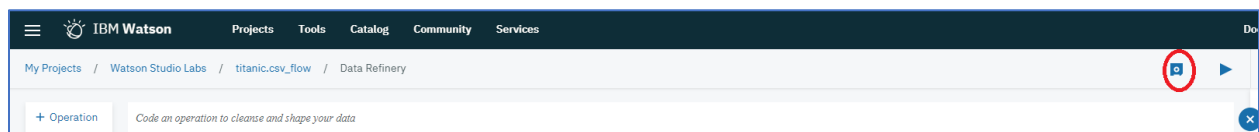
- Now we will drop the `age`, `fare`, and `log_fare` columns as they are no longer needed for modeling purposes. Select the vertical ellipse adjacent to the column and click on **Remove** as shown below.



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


log_fare	age_bin
Decimal	Decimal
2.32497656566603	Remove
2.18055594070364	Remove duplicates
2.18055594070364	Remove empty rows
2.18055594070364	Sort ascending
2.18055594070364	Sort descending
1.42406452541749	Substitute
1.89186236009324	CONVERT COLU... >
-Inf	View All
1.71163178923691	
1.69464204659912	
2.35702912303943	4
2.35702912303943	3
1.84073323461181	3

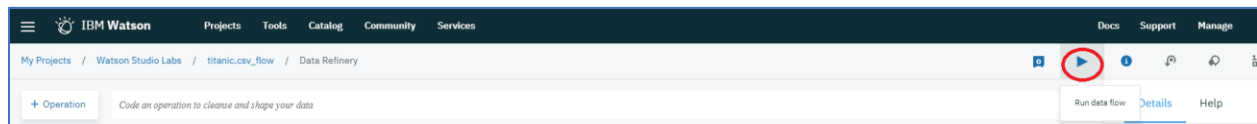
11. Save the Data Flow by clicking on the Save Data Flow icon .



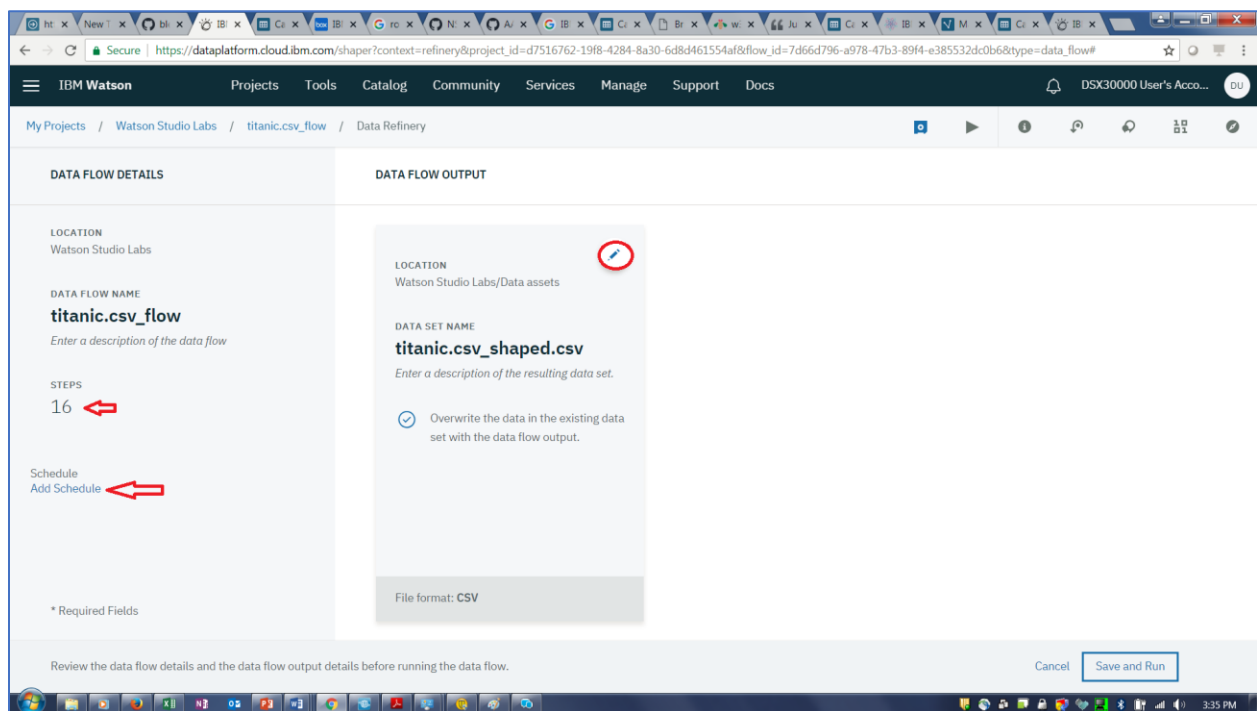
Step 5: Run the sequence of Data Flow operations on the entire data set.

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 run option.

1. Click on run icon 



2. Note the number of steps used to transform the data. It should be 16. Also, a schedule can be set up if the transformation process needs to run on a scheduled basis. We are just going to do a one-time run. Change the name of the output file by clicking on the edit option (pencil icon).



3. Type in **titanic_processed.csv** as the new file name, and click on the check mark.

DATA FLOW OUTPUT

Edit output

☒ ☐

LOCATION *

Watson Studio Labs/Data assets

Change Location

DATA SET NAME *

titanic_processed.csv

79

DESCRIPTION

Enter a description of the resulting data set.

300

FILE FORMAT

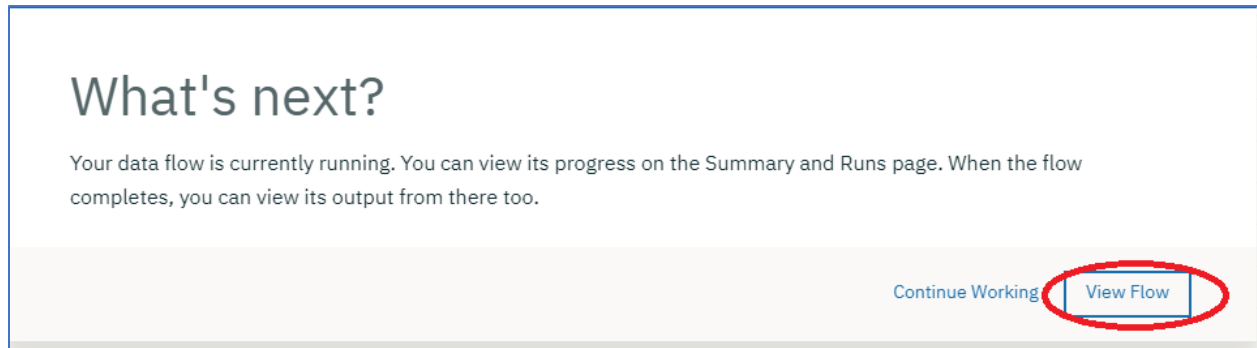
CSV

▼

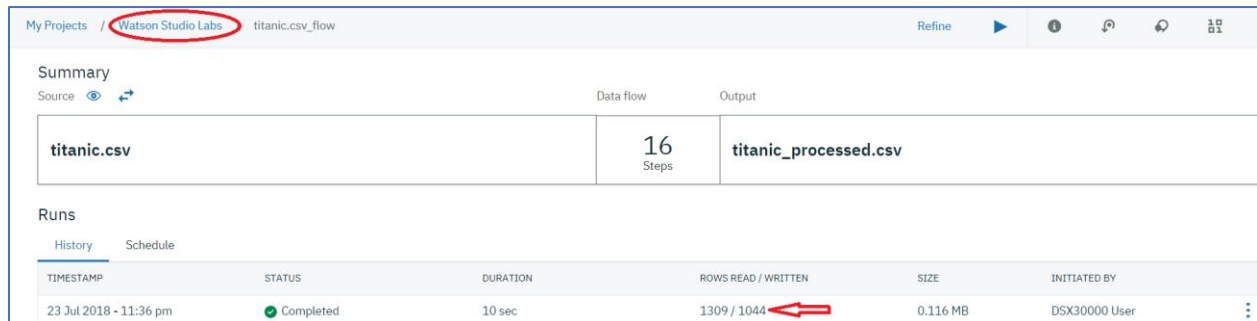
4. Click **Save and Run**.

DATA FLOW DETAILS	DATA FLOW OUTPUT
<div><div>LOCATION</div><div>Watson Studio Labs</div></div> <div><div>DATA FLOW NAME</div><div>titanic.csv_flow</div><div>Enter a description of the data flow</div></div> <div><div>STEPS</div><div>16</div></div> <div><div>Schedule</div><div>Add Schedule</div></div> <div><div>* Required Fields</div></div>	<div><div>LOCATION</div><div>Watson Studio Labs/Data assets</div><div><input checked="" type="checkbox"/></div></div> <div><div>DATA SET NAME</div><div>titanic_processed.csv</div><div>Enter a description of the resulting data set.</div></div> <div><div><input checked="" type="checkbox"/> Overwrite the data in the existing data set with the data flow output.</div></div> <div><div>File format: CSV</div></div>
<div>Review the data flow details and the data flow output details before running the data flow.</div> <div><div>Cancel</div><div>Save and Run</div></div>	

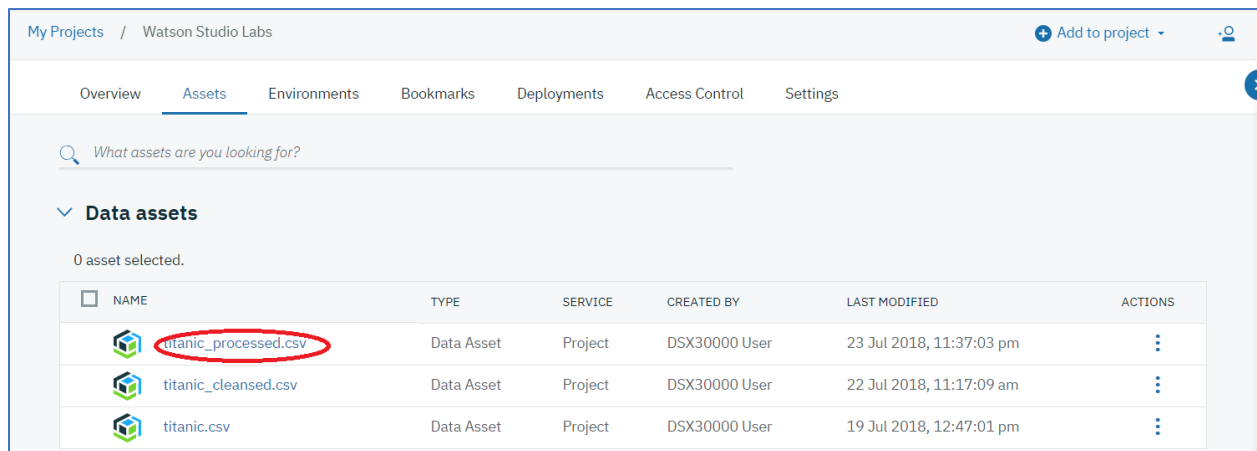
5. You can continue to work on other items, or monitor the Data Flow run status.



6. The completed flow is shown below. Note that 1044 records were written to the output file. Click on Watson Studio Labs to go back to the project Assets page.



7. The output of the Data Refinery process should be listed in the Data Assets. Click on the asset to view the contents.



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

My Projects / Watson Studio Labs / titanic_processed.csv

Refine

PreviewProfile

Schema: 12 Columns

Preview (1000 rows)

PCLASS	SURVIVED	NAME	SEX	SIBSP	PARCH	TICKET	EMBARKED	SURVIVED_VALUE	PCLASS_VALUE	AGE_BIN	LOG_FARE_BIN
Type: String	Type: String	Type: String	Type: String	Type: String	Type: String	Type: String	Type: String	Type: String	Type: String	Type: Decimal	Type: Decimal
1	1	Allen, Miss. Elisat	female	0	0	24160	S	Y	first	3.0	3.0
1	1	Allison, Master. H	male	1	2	113781	S	Y	first	0.0	3.0
1	0	Allison, Miss. Hel	female	1	2	113781	S	N	first	0.0	3.0
1	0	Allison, Mr. Hudsr	male	1	2	113781	S	N	first	3.0	3.0
1	0	Allison, Mrs. Hudr	female	1	2	113781	S	N	first	3.0	3.0
1	1	Anderson, Mr. Ha	male	0	0	19952	S	Y	first	4.0	2.0
1	1	Andrews, Miss. Ku	female	1	0	13502	S	Y	first	4.0	2.0
1	0	Andrews, Mr. Tho	male	0	0	112050	S	N	first	3.0	
1	1	Appleton, Mrs. Ed	female	2	0	11769	S	Y	first	4.0	2.0
1	0	Artagaveytia, Mr.	male	0	0	PC 17609	C	N	first	5.0	2.0