

# 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 you 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.

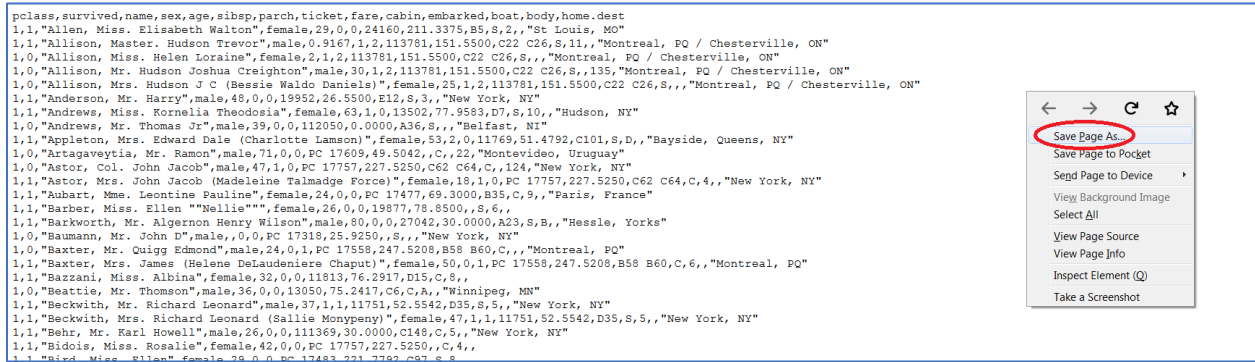
As you interact with the Data Refinery tool, it will perform the ordered operations on a subset of the data. When you are satisfied with the flow of operations, you save the Data Refinery flow and then run a job to apply the series of operations on the entire dataset.

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:

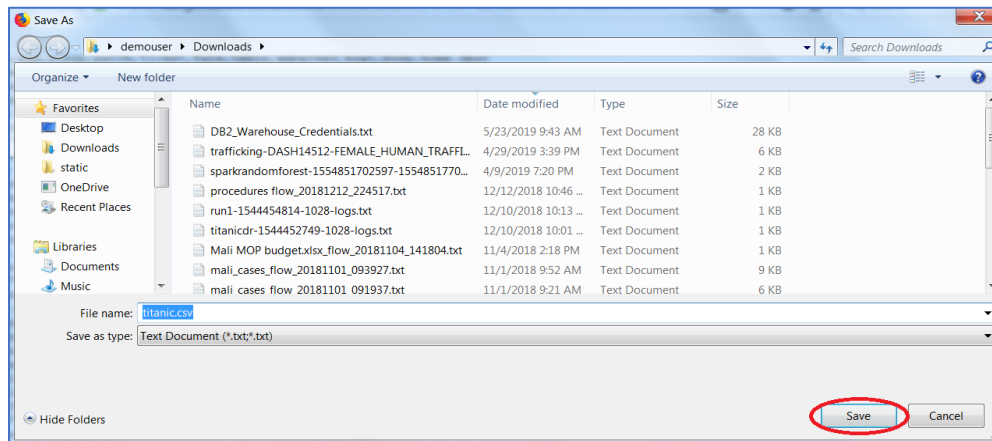
1. Use the Data Refinery Tool to:
  - a. Profile the data to help determine missing values
  - b. Visualize the data to gain a better understanding
  - c. Prepare the data for modeling
  - d. Run the sequence of data preparation operations on the entire data set.

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

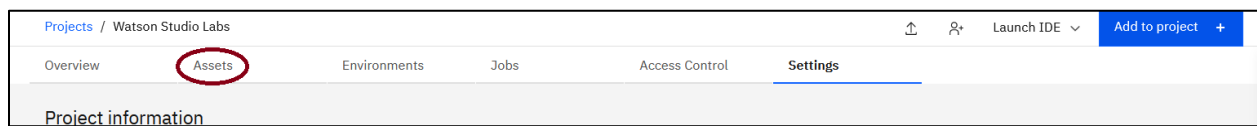
1. Download the Titanic data file from the following location by clicking [here](#).
2. Right-click on the screen and click on Save Page As ...



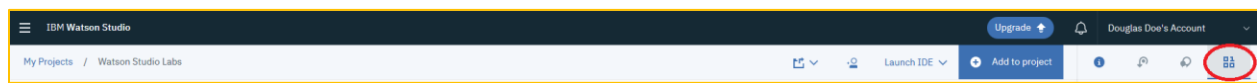
- Click on **Save** to save the titanic.csv file (Note, if the file shown is titanic.csv.txt, remove the .txt).




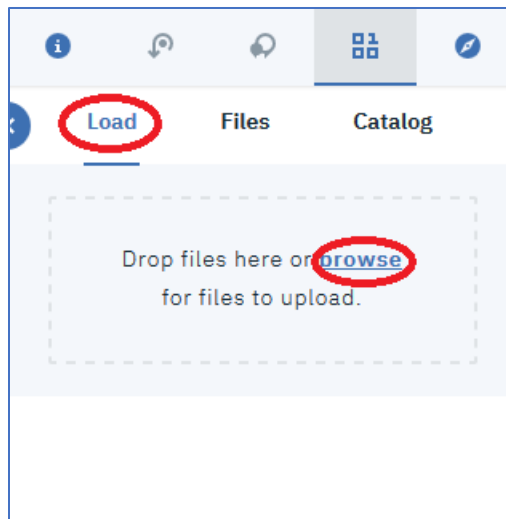
- Go back to your Watson Studio Labs project. Click on the **Assets** tab.



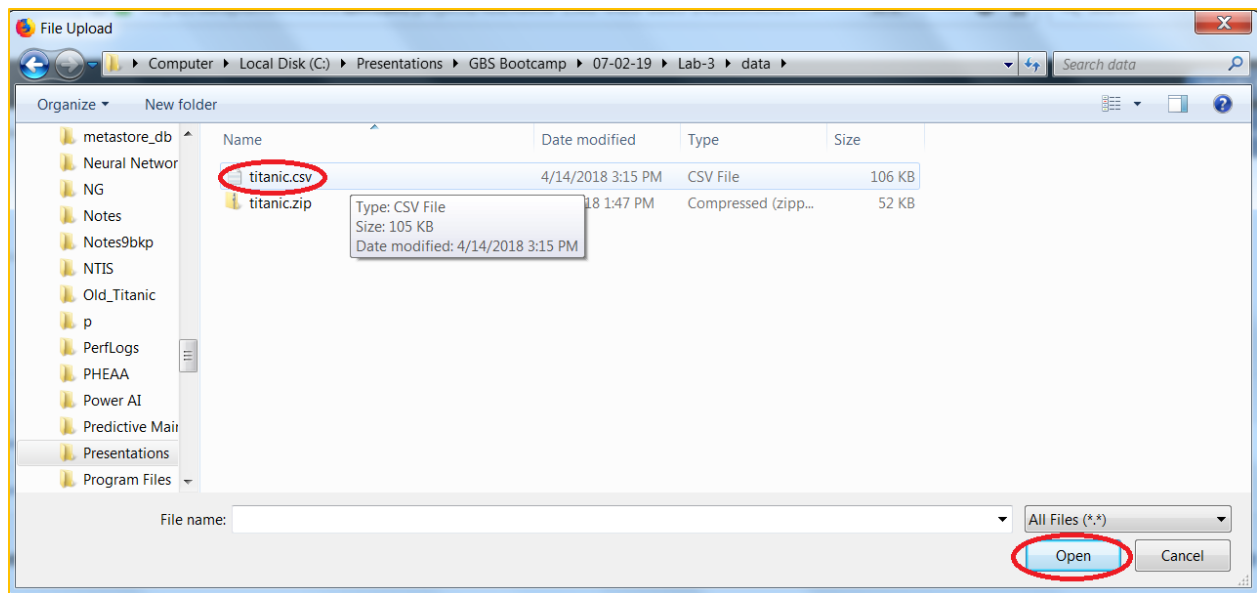
- Click on the  icon.



- 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**.

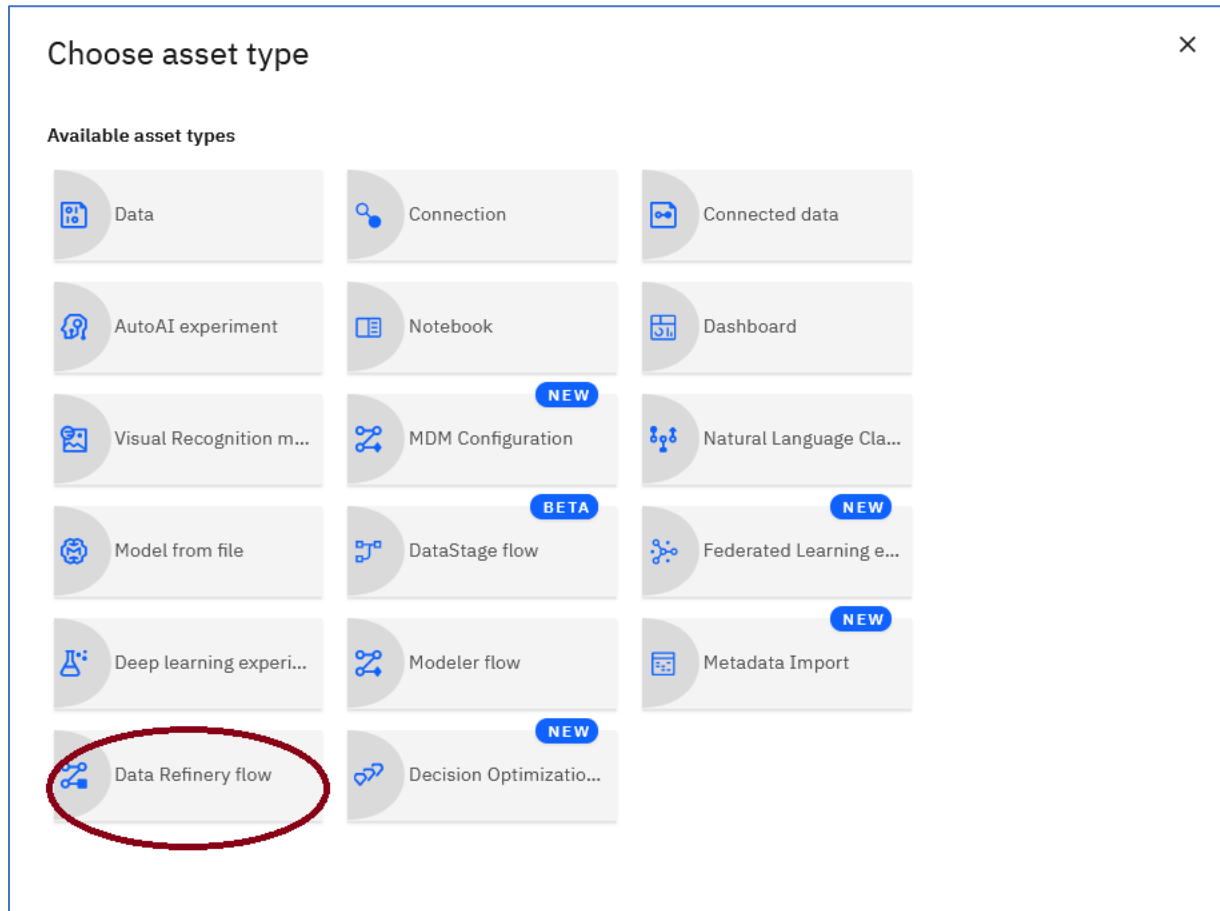
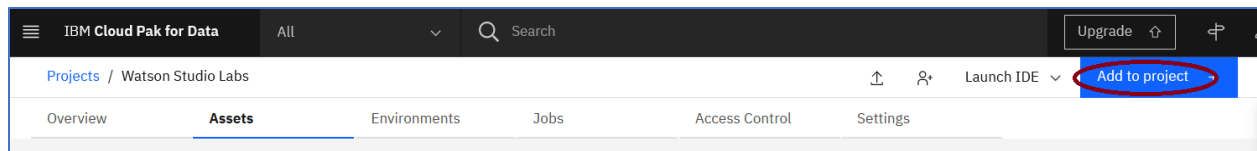


8. The file is now added as a Data Asset.

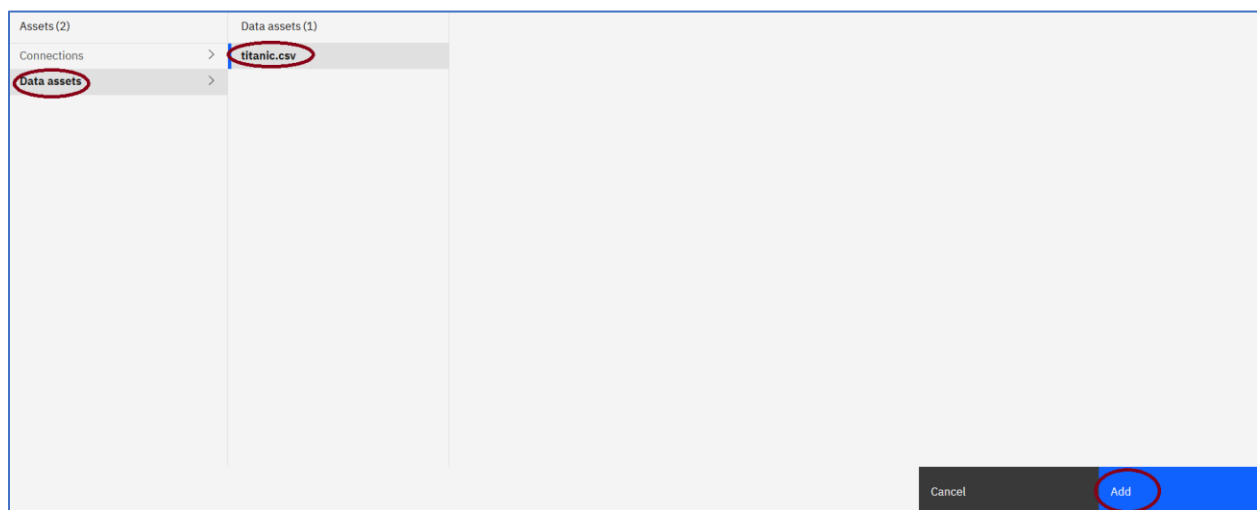


Step 2: Profile the data to help determine missing values.

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



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



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



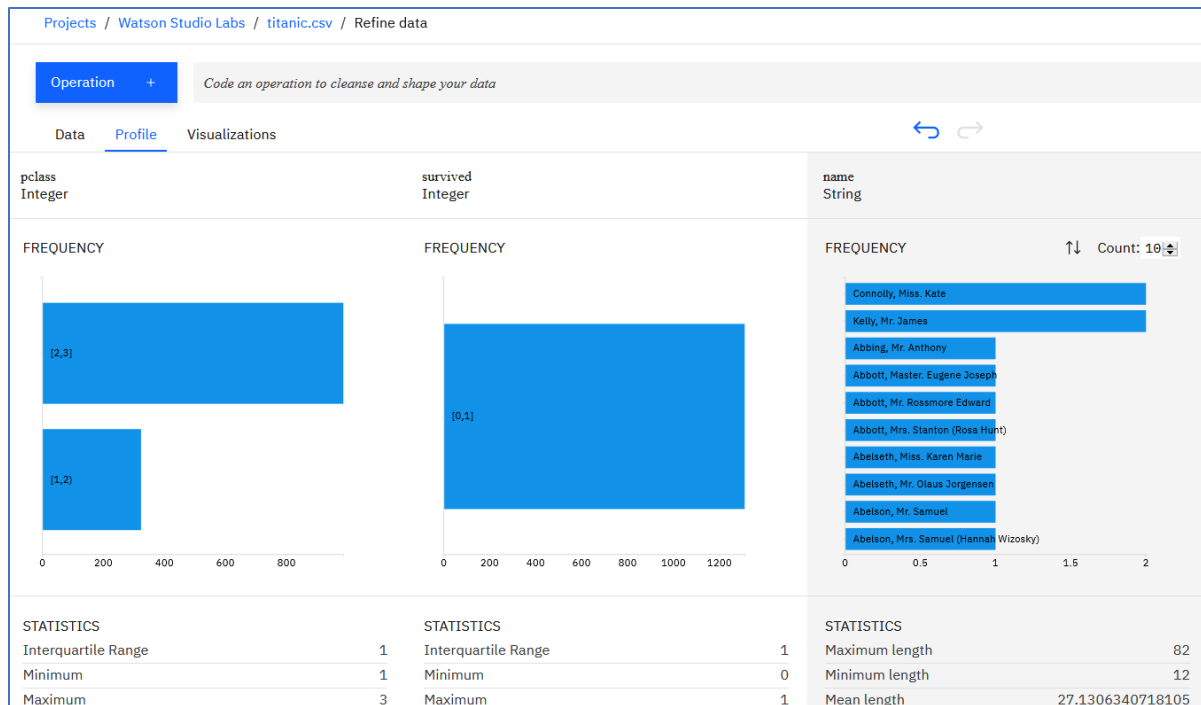
4. Click on the **Profile** tab.

The screenshot shows the IBM Cloud Pak for Data interface with the 'Data Refinery' panel. The 'Profile' tab is selected and circled in red. The table displays the top 10 count values for each column. The columns are: pclass, survived, name, sex, age, sibsp. The table has 11 rows of data.

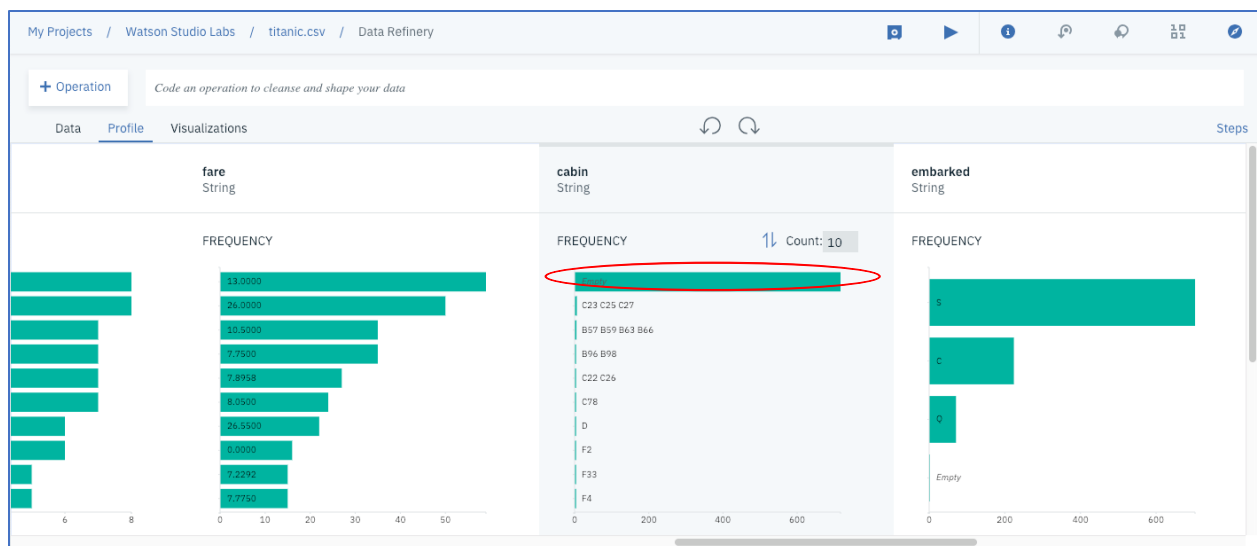
	pclass	survived	name	sex	age	sibsp
1	1	1	Allen, Miss. Elisabeth Walton	female	29	0
2	1	1	Allison, Master. Hudson Trevor	male	0.9167	1
3	1	0	Allison, Miss. Helen Loraine	female	2	1
4	1	0	Allison, Mr. Hudson Joshua Creighton	male	30	1
5	1	0	Allison, Mrs. Hudson J C (Bessie Waldo Daniels)	female	25	1
6	1	1	Anderson, Mr. Harry	male	48	0
7	1	1	Andrews, Miss. Kornelia Theodosia	female	63	1
8	1	0	Andrews, Mr. Thomas Jr	male	39	0
9	1	1	Appleton, Mrs. Edward Dale (Charlotte Lamson)	female	53	2
10	1	0	Artagaveytia, Mr. Ramon	male	71	0
11	1	0	Astor, Col. John Jacob	male	47	1

SOURCE FILE: titanic.csv SAMPLE SIZE: First 1000 rows

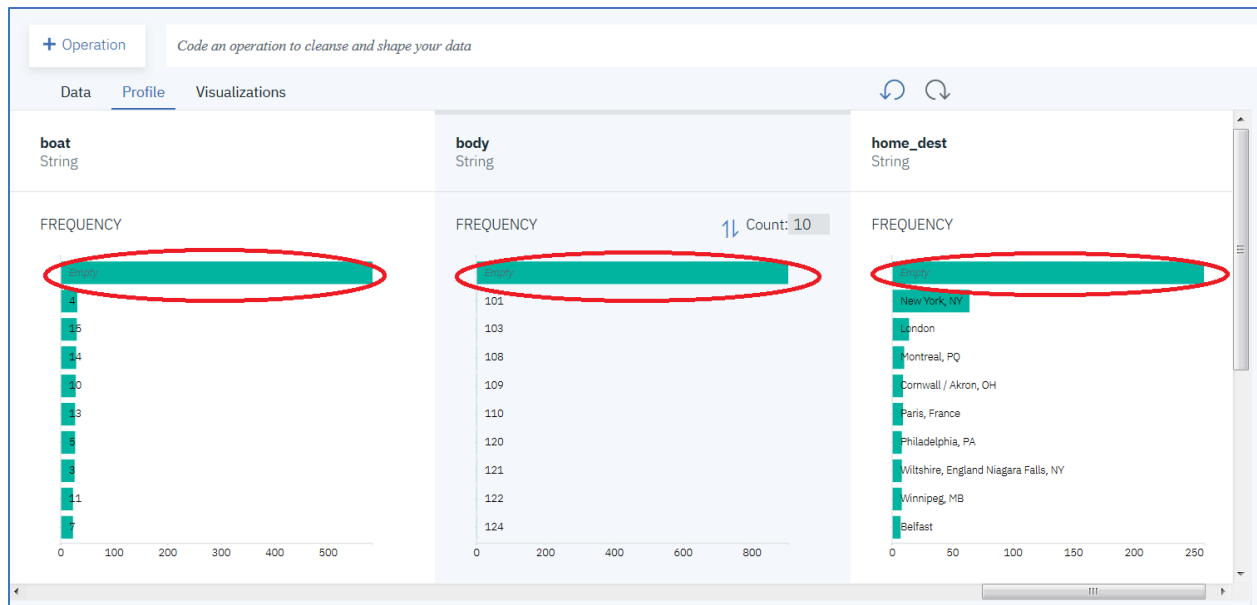
5. The Profile panel displays the counts of the top 10 count values for each column. Note that you can change 10 to another number if desired. You can also switch to the bottom 10 counts for a column. Scroll to the right to view the cabin column.



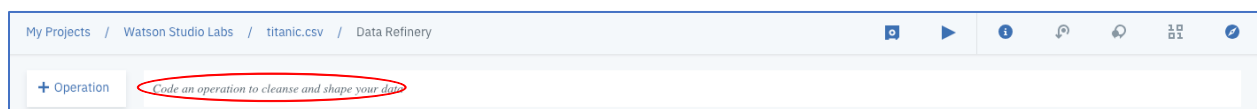
- Note that the cabin column has many missing values and should be removed as part of the data preparation step.



- 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.

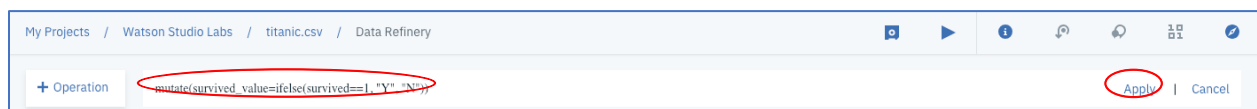


8. Age and Embarked also have missing values. Embarked has very few missing values. Age has over 200 missing values, but we will keep that column in the analysis. As part of data preparation, we will remove the rows that contain the missing age and embarked values.
9. Click on the **Data** tab. We will 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**.

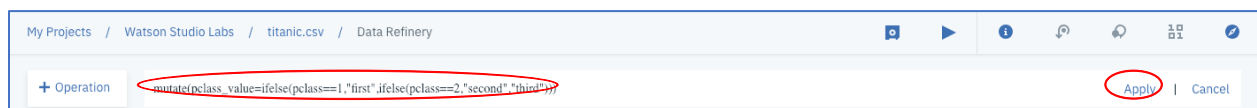


10. Copy and paste the following:  
`mutate(survived_value=ifelse(survived==1, "Y", "N"))`

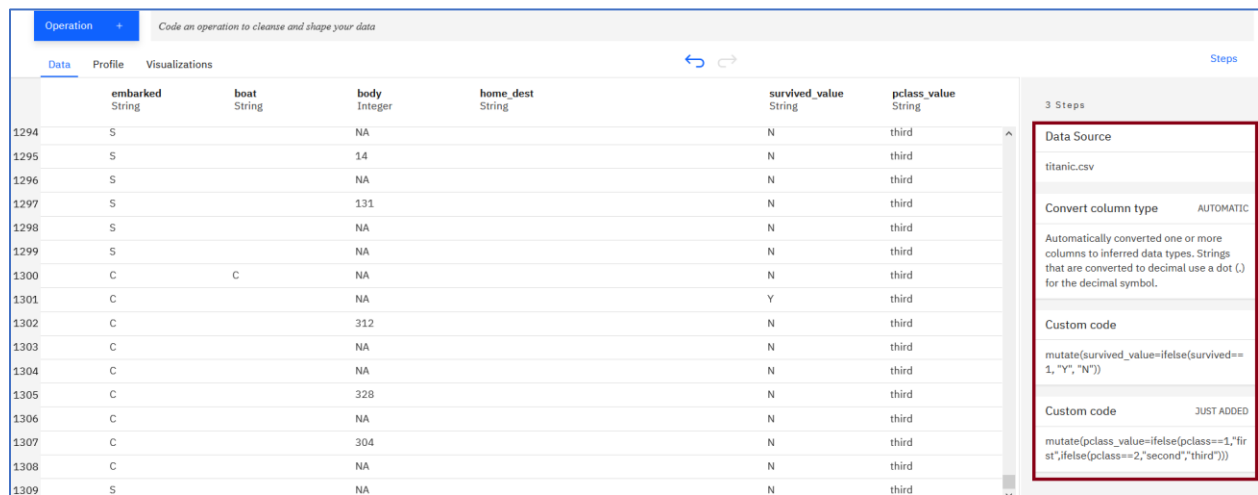
and then click Apply. If you scroll to the right, you should see the new column “survived\_value”.



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



12. 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.

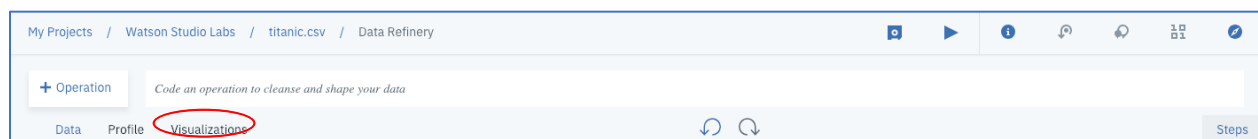


The screenshot shows the Data Refinery interface. The main table displays data for passengers 1294 through 1309. The columns are: embarked (String), boat (String), body (Integer), home\_dest (String), survived\_value (String), and pclass\_value (String). The 'Steps' panel on the right shows a list of transformations: 'Data Source' (titanic.csv), 'Convert column type' (AUTOMATIC), and two 'Custom code' blocks. The first custom code block contains the transformation: `mutate(survived_value=ifelse(survived==1,"Y","N"))`. The second custom code block contains the transformation: `mutate(pclass_value=ifelse(pclass==1,"first",ifelse(pclass==2,"second","third")))`.

	embarked	boat	body	home_dest	survived_value	pclass_value
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

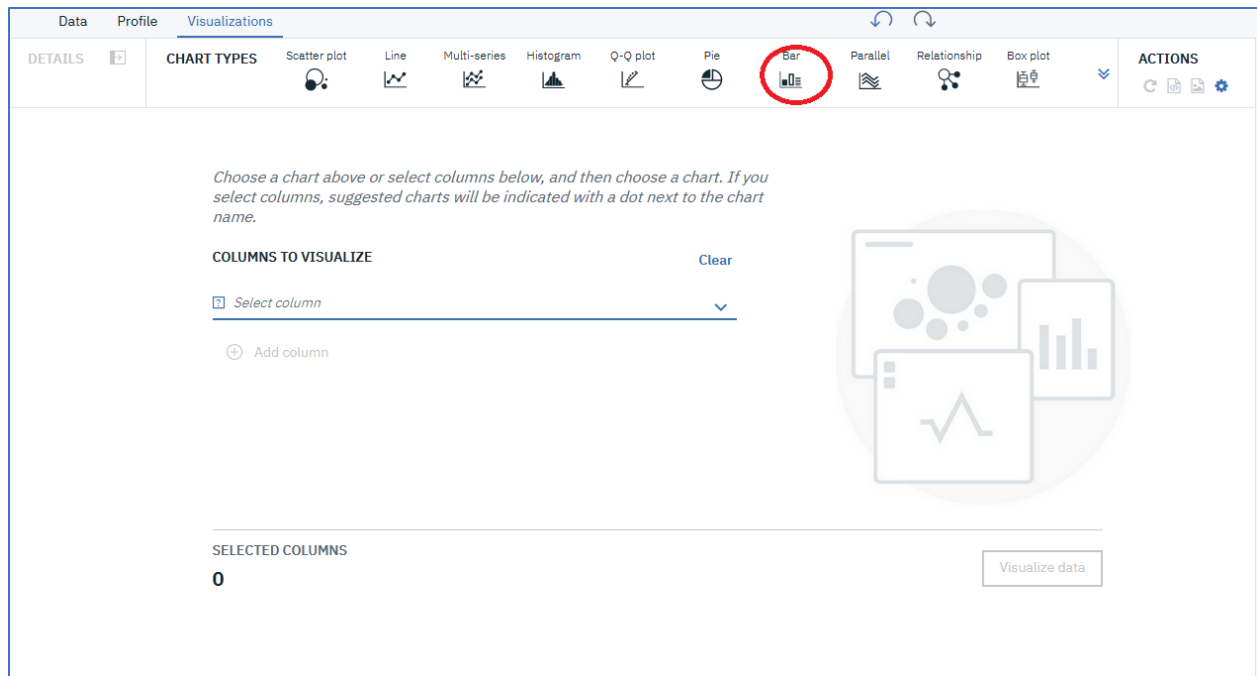
## Step 3: Visualize the data to get a better understanding

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

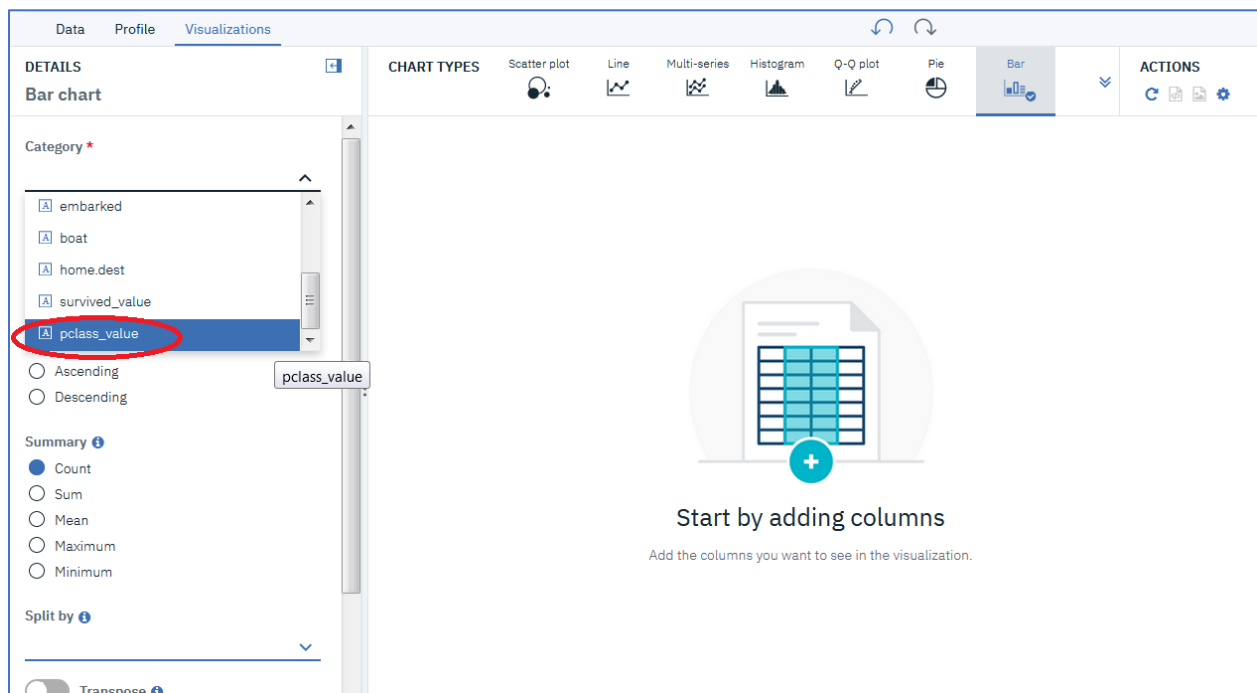


2. Let's take a look at the breakdown of passengers by passenger class. We will use our new pclass\_value field. Select the **Bar** Chart Type.

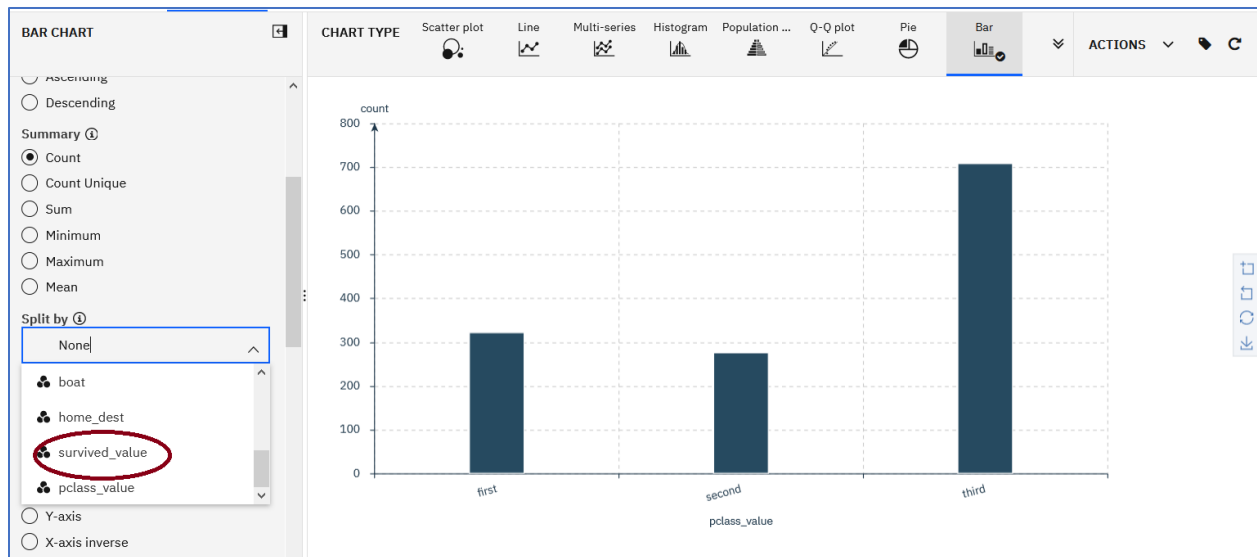




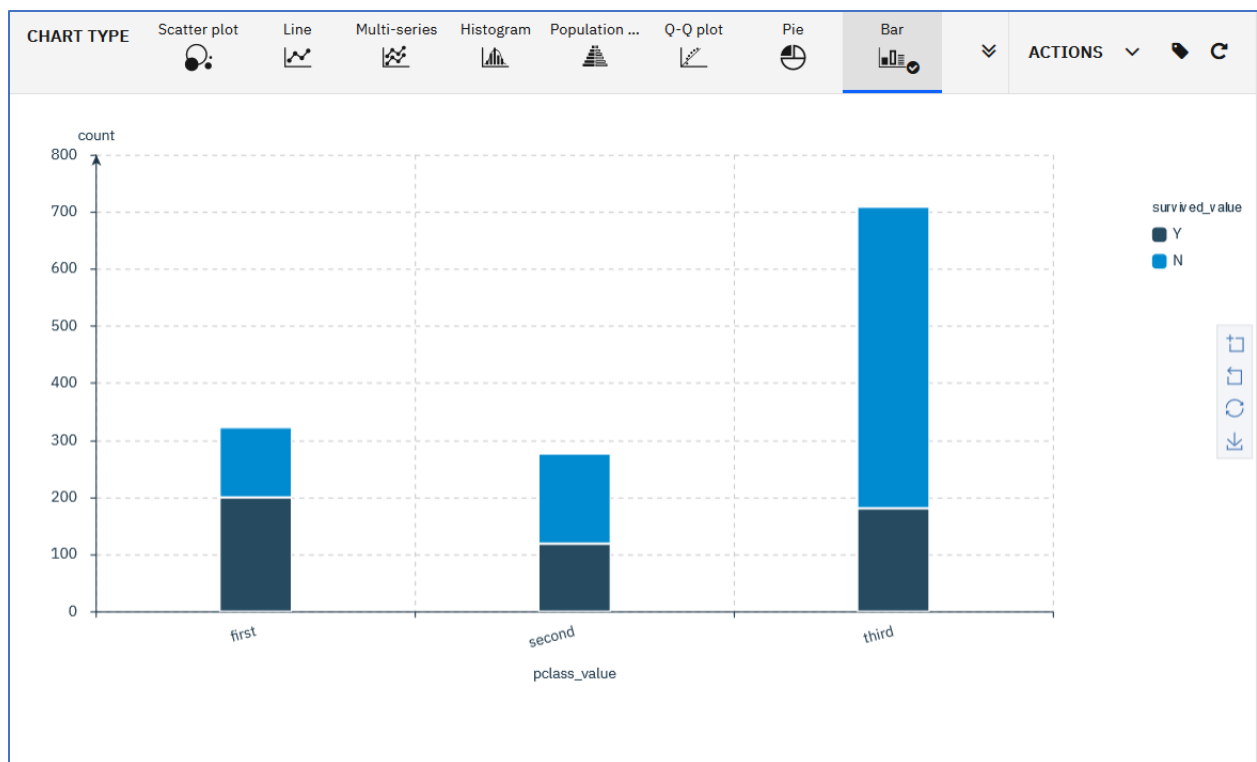
3. In the **Category** required field, select **pclass\_value**.



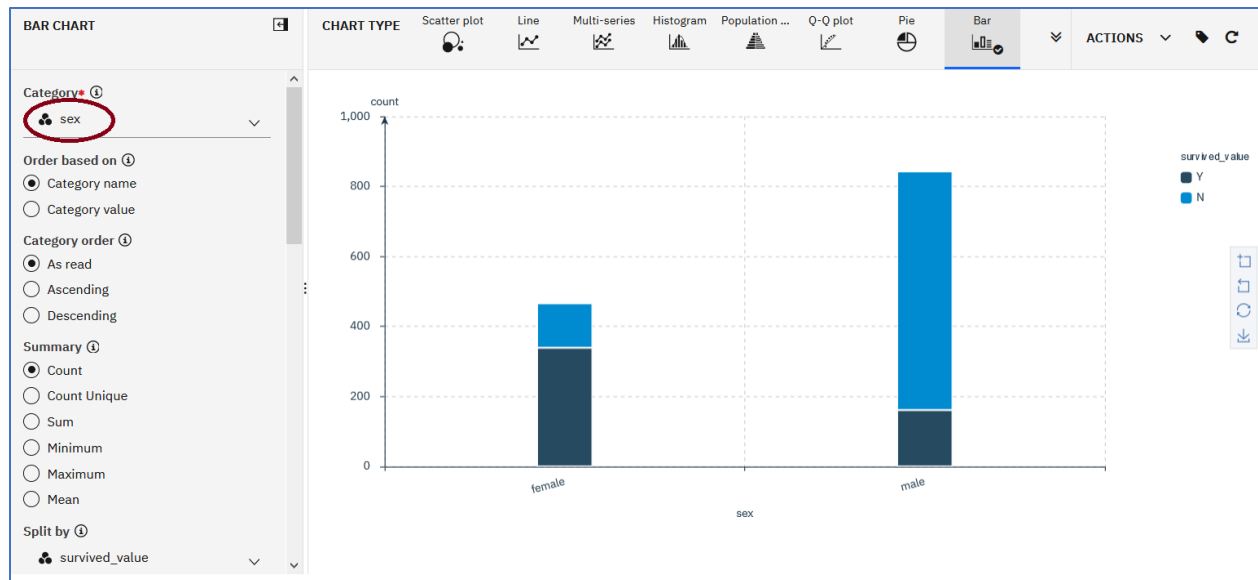
4. In the **Split by** field, select **survived\_value**.



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.



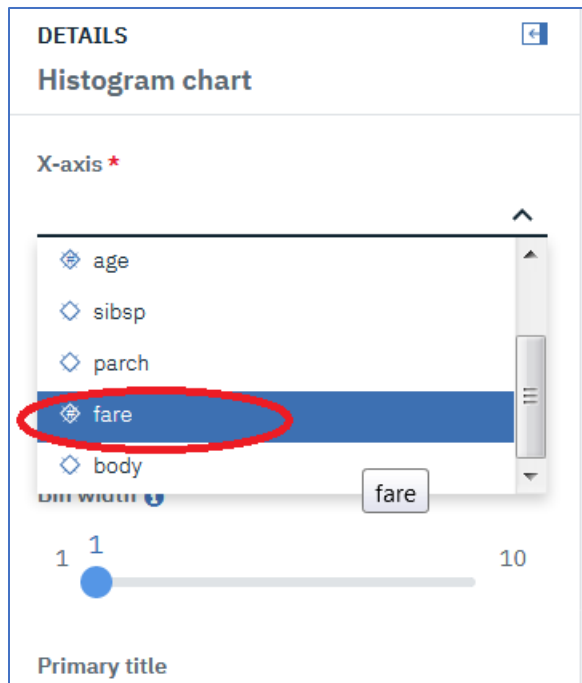
6. Change the **Category** to **sex**. We can see that survivorship for females is significantly greater than for males.



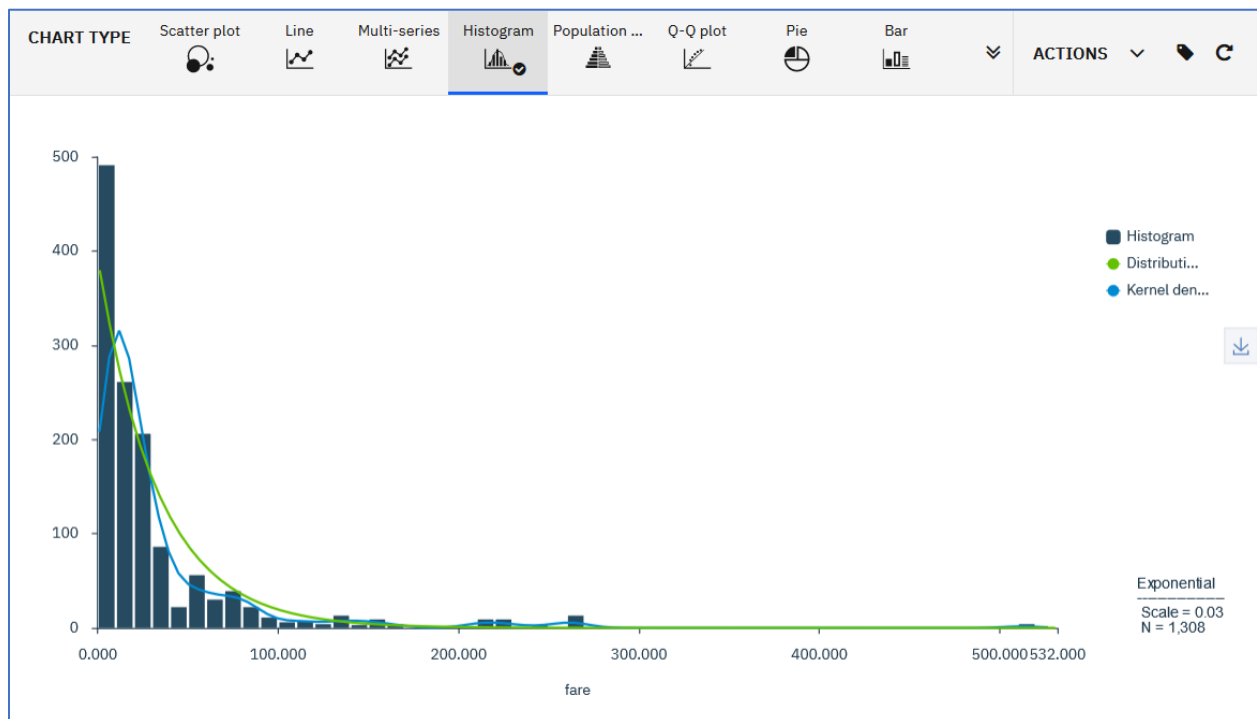
7. Click on the **Histogram** Chart Type.



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.



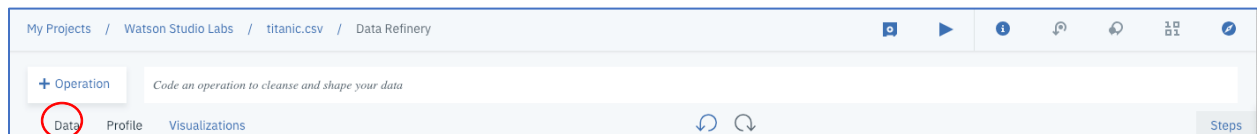
Step 4: Prepare the data for modeling

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**.

The screenshot shows a data table with columns 'cabin', 'embarked', and 'boat'. The 'cabin' column is highlighted with a red circle, and a vertical ellipse next to it is also circled in red. A context menu is open over the 'cabin' column, with the 'Remove' option selected and circled in red. The menu includes options like 'Remove duplicates', 'Remove empty rows', 'Sort ascending', 'Sort descending', 'Substitute', 'CONVERT COLU...', 'TEXT', and 'View All'. The table data is as follows:

cabin	embarked	boat
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.

6 STEPS

Data Source : titanic.csv

Custom code

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

Custom code

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

Remove

Removed cabin

Remove

Removed boat

Remove

Removed body

Remove JUST ADDED

Removed home.dest

- 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
S		first
C		first
C		first
C	Y	first
C	Y	first
S	Y	first

- 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**.

fare	embarked	survive	6 STEPS
String	String	String	
211.3375		Y	Data Sour
151.5500		Y	Custom co
151.5500		N	mutate(sur
151.5500		N	ifelse(survi
151.5500		N	Custom co
26.5500		Y	mutate(pcl
77.9583		Y	ifelse(pclas
0.0000		N	d"
51.4792			a
49.5042			d c
227.5250			a
227.5250			d b
69.3000			
78.8500			
30.0000			

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**.

+ Operation	mutate(log_fare=log10(fare))	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**.

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

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.

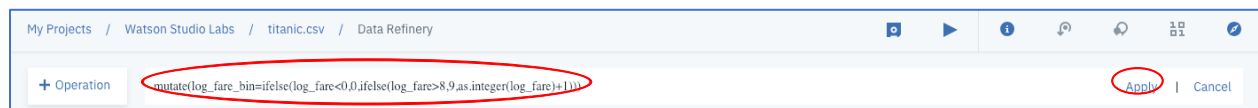
Bin	Age Range
0	0-5
1	6-11
2	12-17
3	18-39
4	40-64
5	65-79
6	Over 79



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.




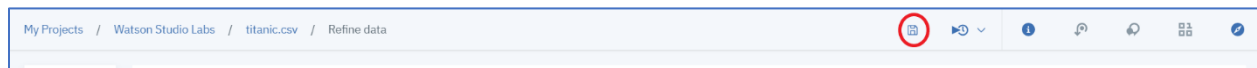
age	sibsp		
Integer	String		
29		Remove	
0		Remove duplicates	
2		Remove empty rows	
30		Sort ascending	
25		Sort descending	
48		Substitute	
63		CONVERT COLU... >	
39			
53			
		View All	

fare	embarked		
Decimal	String		
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			
51.4792		View All	
49.5042			
227.525	C		
227.525	C		




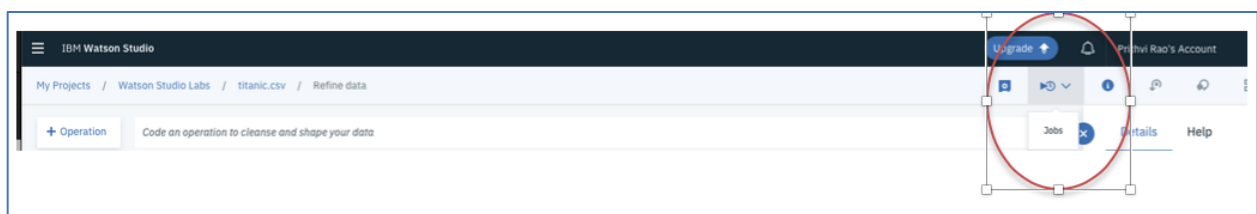
log_fare Decimal	age_bin Decimal
2.32497656566603	
2.18055594070364	
2.18055594070364	
2.18055594070364	
2.18055594070364	
1.42406452541749	
1.89186236009324	
-Inf	
1.71163178923691	
1.69464204659912	
2.35702912303943	4
2.35702912303943	3
1.84073323461181	3

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

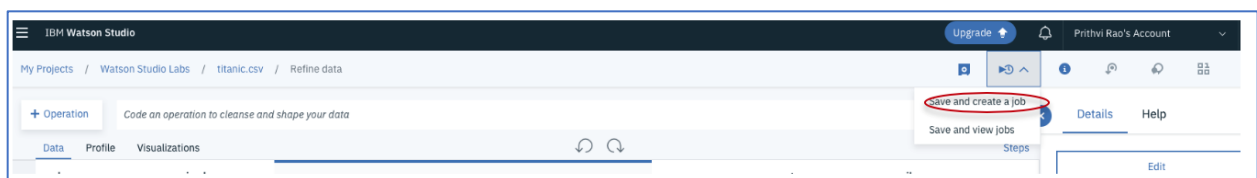


Step 5: Run the sequence of Data Flow operations on the entire data set.

- 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 **Jobs** icon .



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



- This action results in the following page display. Fill in the **Name**, for example **titantic\_refinery\_flow**, and click on the **Next** button.

**Define details**

Associated asset  
📁 titanic.csv\_flow (16 Steps)

Name

Description (optional)

[Cancel](#) [Next](#)

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

**Configure**

Data assets

Input	Output
titanic.csv <small>CSV</small>	titanic_csv_shaped <small>CSV</small>

Environment

[Cancel](#) [Back](#) [Next](#)

- 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.

**Schedule**

☐ Schedule off

[Cancel](#) [Back](#) [Next](#)

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

**Review and create**

[Details](#) [Data assets](#) [Configuration](#)

Associated asset  
titanic.csv\_flow (16 Steps)

Name  
titanic\_refinery\_flow

Description  
[Add Description](#)

Schedule [Schedule](#)

**Scheduled to run**  
No schedule created

**Input**

titanic.csv CSV

→

**Output**

titanic\_csv\_shaped CSV

Environment:  
Default Data Refinery XS

[Cancel](#) [Back](#) [Create](#) [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.

Projects / Watson Studio Labs / titanic.csv\_flow

Operation + Code an operation to cleanse and shape your data

The job was successfully created. See job details.

	pclass Integer	survived Integer	name String	sex String	sibsp Integer	parch Integer	ticket String
1	1	1	Allen, Miss. Elisabeth Walton	female	0	0	24160
2	1	1	Allison, Master. Hudson Trevor	male	1	2	113781
3	1	0	Allison, Miss. Helen Loraine	female	1	2	113781
4	1	0	Allison, Mr. Hudson Joshua Creighton	male	1	2	113781
5	1	0	Allison, Mrs. Hudson J C (Bessie Waldo Daniels)	female	1	2	113781
6	1	1	Anderson, Mr. Harry	male	0	0	19952
7	1	1	Andrews, Miss. Kornelia Theodosia	female	1	0	13502
8	1	0	Andrews, Mr. Thomas Jr	male	0	0	112050
9	1	1	Appleton, Mrs. Edward Dale (Charlotte Lamson)	female	2	0	11769
10	1	0	Artagaveytia, Mr. Ramon	male	0	0	PC 17609
11	1	0	Astor, Col. John Jacob	male	1	0	PC 17757
12	1	1	Astor, Mrs. John Jacob (Madeleine Talmadge Force)	female	1	0	PC 17757
13	1	1	Aubart, Mme. Leontine Pauline	female	0	0	PC 17477

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

titanic\_refinery\_flow

No description

Runs (1)

Start time	Status	Duration	Started by	Action
Mar 19, 2021 9:36:37 PM	Running	---	Mathias Doe	

titanic\_refinery\_flow

No description

Runs (1)

Start time	Status	Duration	Started by	Action
Mar 19, 2021 9:36:37 PM	Completed	22 seconds	Mathias Doe	

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

My projects	Watson Studio Labs	titanic_refinery_flow
titanic_refinery_flow		
No description		

10. Click on **titanic.csv\_shaped** to view the asset contents.

Data assets				
0 assets selected.				
<input type="checkbox"/>	Name	Type	Created by	Last modified ↓
<input type="checkbox"/>	CSV <b>titanic.csv_shaped</b>	Data Asset	Steven Doe	Oct 25, 2020, 6:04 PM
<input type="checkbox"/>	CSV titanic.csv	Data Asset	Steven Doe	Oct 25, 2020, 12:32 PM

11. 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.csv\_shaped.csv

Preview

Lineage

Schema: 12 Columns

Preview: 1000 rows | Last refresh: 14 minutes ago | [Refresh](#)

Refine

pclass	survived	name	sex	sibsp	parch	ticket	embarked	survived_v...	pclass_val...
Type: String	Type: String	Type: String	Type: String	Type: String	Type: String	Type: String	Type: String	Type: String	Type: String
1	0	Allison, Miss. Hel	female	1	2	113781	S	N	first
1	1	Anderson, Mr. He	male	0	0	19952	S	Y	first
1	1	Appleton, Mrs. E	female	2	0	11769	S	Y	first
1	1	Astor, Mrs. John	female	1	0	PC 17757	C	Y	first
1	1	Barkworth, Mr. A	male	0	0	27042	S	Y	first
1	1	Baxter, Mrs. Jam	female	0	1	PC 17558	C	Y	first
1	1	Beckwith, Mr. Ric	male	1	1	11751	S	Y	first
1	1	Bidois, Miss. Ros	female	0	0	PC 17757	C	Y	first
1	1	Bishop, Mr. Dickli	male	1	0	11967	C	Y	first
1	1	Bjornstrom-Steff	male	0	0	110564	S	Y	first
1	1	Bonnell, Miss. Ca	female	0	0	36928	S	Y	first
1	1	Bowen, Miss. Gra	female	0	0	PC 17608	C	Y	first
1	0	Brady, Mr. John E	male	0	0	113054	S	N	first
1	1	Brown, Mrs. Jam	female	0	0	PC 17610	C	Y	first
1	1	Burns, Miss. Eliz	female	0	0	16966	C	Y	first
1	1	Calderhead, Mr. I	male	0	0	PC 17476	S	Y	first
1	1	Cardeza, Mrs. Jai	female	0	1	PC 17755	C	Y	first
1	0	Carrau, Mr. Jose	male	0	0	113059	S	N	first

**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.