

5.3.2 Get the Number of Rides for Each City Type

Looking at the x-axis title that was given to you, you see that you will need to get the number of rides for each city for each city type. Omar says there is a quick way to do this using Pandas. Thinking back to your previous experience using Pandas, your mind quickly does a rewind to the time you did something similar.

To get the number of rides for each city by each type of city, we have to create a Series where the index is the name of the city and the column is the number of rides for that city.

REWIND

To create a Data Series with one of the columns in a DataFrame, we can use the `groupby()` function and add the column inside the parentheses.

Using the `groupby()` function can be used to group large amounts of data when we want to compute mathematical operations on these groups.

We'll use the `groupby()` function to create a Series of data that has the name of the city as the index, apply the `count()` method to the Series for

each city, and select the `ride_id` column.

Add the following code to a new cell:

```
# Get the number of rides for urban cities.  
urban_ride_count = urban_cities_df.groupby(["city"]).count()["ride_id"]  
urban_ride_count.head()
```

Run the cell, and you'll see that the output is a Series with the number of rides shown for each city. Here's a snapshot of the first five rows of the output:

```
city  
Amandaburgh      18  
Barajasview      22  
Carriemouth      27  
Christopherfurt  27  
Deanville        19  
Name: ride_id, dtype: int64
```

Using the same approach, we can create the `suburban_ride_count` and the `rural_ride_count` Series. To do so, add the following code to a new cell and run the cell.

```
# Create the suburban and rural ride count.  
suburban_ride_count = suburban_cities_df.groupby(["city"]).count()["ride_id"]  
  
rural_ride_count = rural_cities_df.groupby(["city"]).count()["ride_id"]
```

We now have one of the three datasets we need to create a bubble chart. Next, we'll use the `groupby()` function on the city type DataFrames to get the average fare for each city type. This will be our second dataset.

