5.3.3

Get the Average Fare for Each City Type

Now that you have the rides parsed by city type, you look at the bubble chart Omar gave you and realize that you need to get the average city fare for each city type. This will allow you to provide further insight into the data.

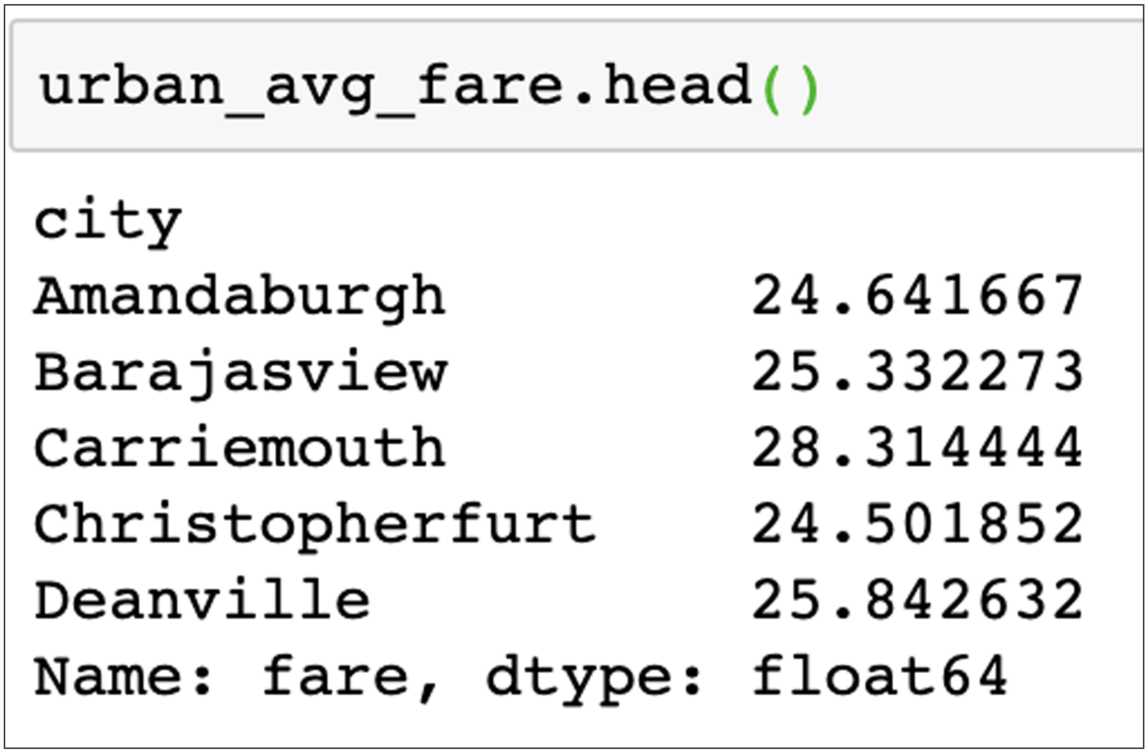
Using the separate DataFrames for each city type, we can calculate the average fare for each city in the urban, suburban, and rural cities.

Using the `groupby()` function, we can chain the `mean()` method to get the averages of the fare column.

Add the following code to a new cell and run the cell.

```
# Get average fare for each city in the urban cities.
urban_avg_fare = urban_cities_df.groupby(["city"]).mean()["fare"]
urban_avg_fare.head()
```

The output after running the cell will be a Series with the average fare for each city in the urban cities. This snapshot shows the first five rows of output:



```
urban_avg_fare.head()
```

```
city
Amandaburgh         24.641667
Barajasview         25.332273
Carriemouth         28.314444
Christopherfurt     24.501852
Deanville           25.842632
Name: fare, dtype: float64
```

Using the same approach, we can calculate the average fare for suburban and rural cities. Add the following code to a new cell.

```
# Get average fare for each city in the suburban and rural cities.
suburban_avg_fare = suburban_cities_df.groupby(["city"]).mean()["fare"]
rural_avg_fare = rural_cities_df.groupby(["city"]).mean()["fare"]
```

Now we have two of 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 number of drivers for each city type. This will be our third and final dataset.