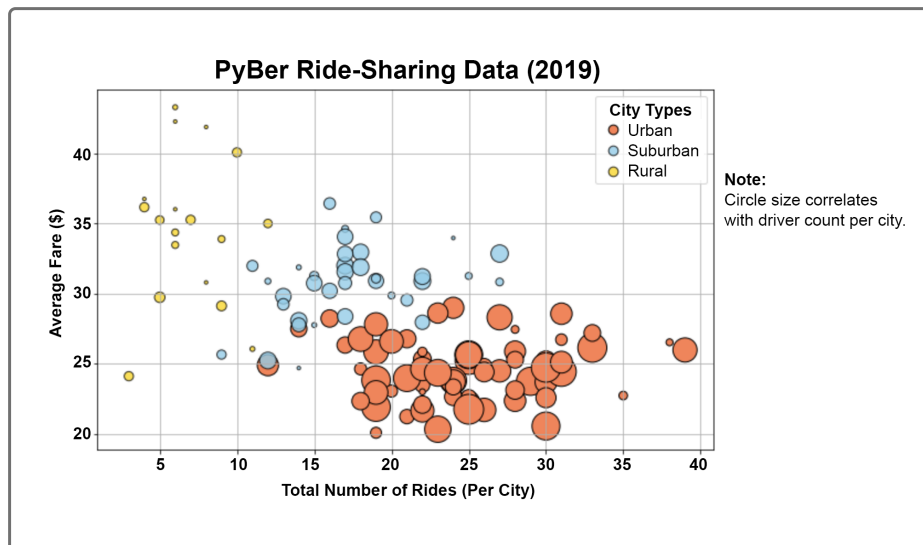


5.3.5 Create Bubble Charts

As you get ready to create a bubble chart, you start thinking that it would be cool to use the company's color scheme for each type of city. You know V. Isualize came up with the color scheme herself back in the first few days of the company: gold for profitability, sky blue for strategy, and coral because she loves the ocean! You can use each color for a city type.

You know that it's details like this that will set your presentation apart. Early promotion, here you come!

If we look at the final product, we can see that the bubble chart contains three different scatter plots, one for each type of city:



Our first task will be to create a scatter plot for each type of city where the following conditions are met:

- The x-axis is the number of rides for each city.
- The y-axis is the average fare for each city.
- The size of each marker is the average number of drivers in each city.

Let's create each scatter plot individually and add them all to one chart.

The first scatter plot we'll make is for urban cities. We'll create our plots using the MATLAB method.

REWIND

To create a scatter plot using the MATLAB method, use the `plt.scatter()` function.

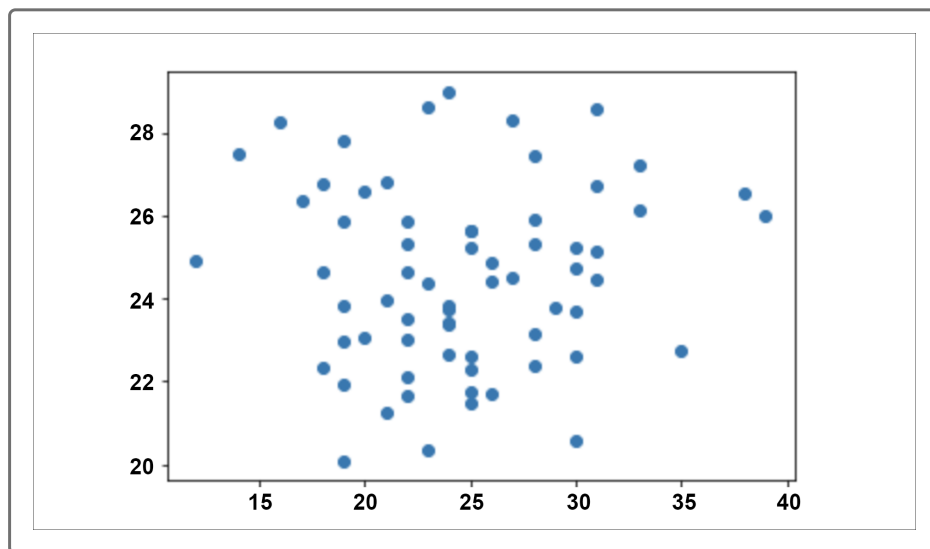
Create the Urban Cities Bubble Chart

To the `plt.scatter()` function, let's add the x- and y-axis data, where the x-axis is the `urban_ride_count` and the y-axis is the `urban_avg_fare`.

Add the following code to a new cell:

```
# Build the scatter plots for urban cities.  
plt.scatter(urban_ride_count, urban_avg_fare)
```

When you run the cell, you'll see the following urban cities scatter plot:



Using the knowledge we gained about adding features to the chart, let's add a title, axes labels, and a legend, and change the color and size of the markers.

Let's start by editing the `plt.scatter(urban_ride_count, urban_avg_fare)` code to increase the size of the markers and add a label for the legend.

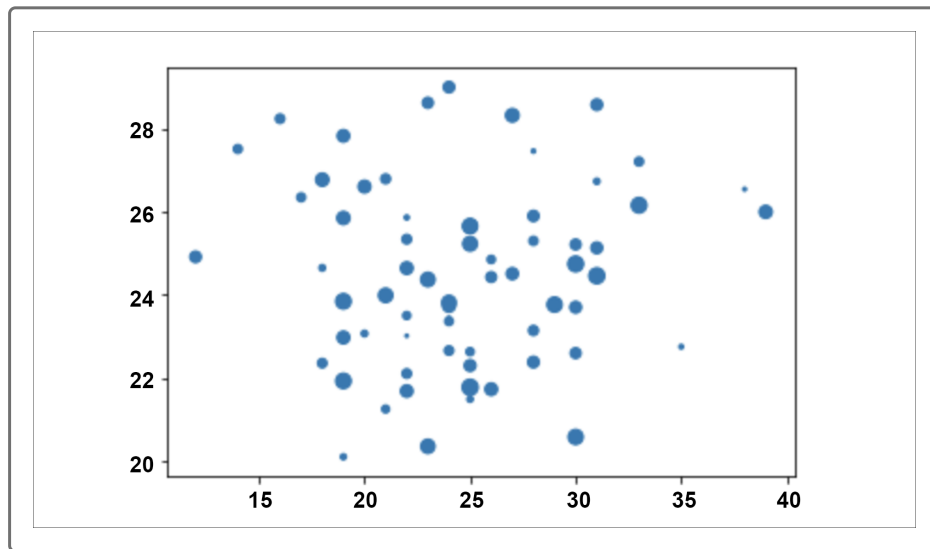
REWIND

To change the size of the markers, add the `s=` parameter to the `scatter()` function.

Go ahead and edit the existing code to look like this:

```
# Build the scatter plots for urban cities.
plt.scatter(urban_ride_count,
            urban_avg_fare,
            s=urban_driver_count)
```

When you run the cell, you'll see the following chart with markers of varying sizes:



Some of the bubbles are still too small, so let's increase the size by a factor of 10 and add a black edge color to the circles that have a line width of 1. While we're changing these features, let's pay homage to the company's color scheme and make the urban markers coral and 20% transparent. We can also add a title, labels for the axes, a legend, and a grid.

REWIND

To change the marker color, we add the `color=` parameter to the `scatter()` function.

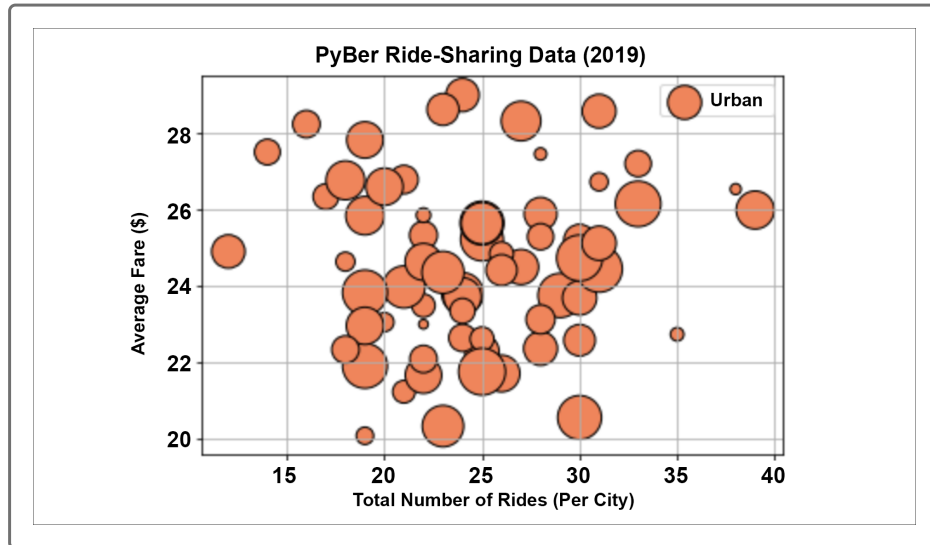
To add a title, x-axis and y-axis labels, and a legend, we use the `plt.title()`, `plt.ylabel()`, `plt.xlabel()`, and `plt.legend()` functions, respectively.

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

```
# Build the scatter plots for urban cities.
plt.scatter(urban_ride_count,
            urban_avg_fare,
            s=10*urban_driver_count, c="coral",
            edgecolor="black", linewidths=1,
            alpha=0.8, label="Urban")
plt.title("PyBer Ride-Sharing Data (2019)")
```

```
plt.ylabel("Average Fare ($)")
plt.xlabel("Total Number of Rides (Per City)")
plt.grid(True)
# Add the legend.
plt.legend()
```

The chart in the output cell will have coral markers that vary in diameter based on the average number of drivers in each city, a title, and axes labels.



Create the Suburban Cities Bubble Chart

Now that we know what the final individual bubble chart should look like, we can repurpose the code and change some variables to create the suburban cities bubble chart.

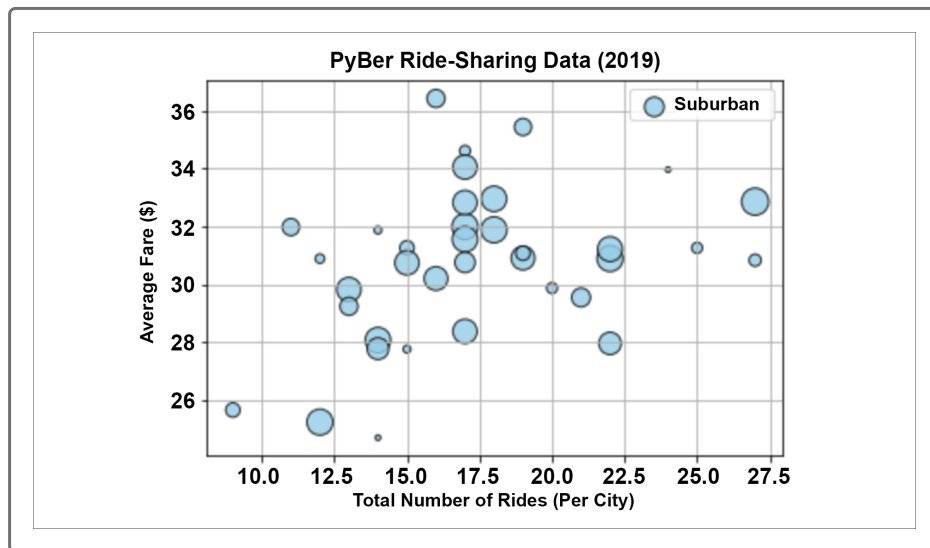
Let's create the same bubble chart as we did for the urban cities chart but change the color to sky blue.

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

```
# Build the scatter plots for suburban cities.
plt.scatter(suburban_ride_count,
            suburban_avg_fare,
            s=10*suburban_driver_count, c="skyblue",
            edgecolor="black", linewidths=1,
            alpha=0.8, label="Suburban")
plt.title("PyBer Ride-Sharing Data (2019)")
plt.ylabel("Average Fare ($)")
```

```
plt.xlabel("Total Number of Rides (Per City)")
plt.grid(True)
# Add the legend.
plt.legend()
```

The chart in the output cell will have sky-blue markers that vary in diameter based on the average number of drivers in each city, a title, and axes labels.



Create the Rural Cities Bubble Chart

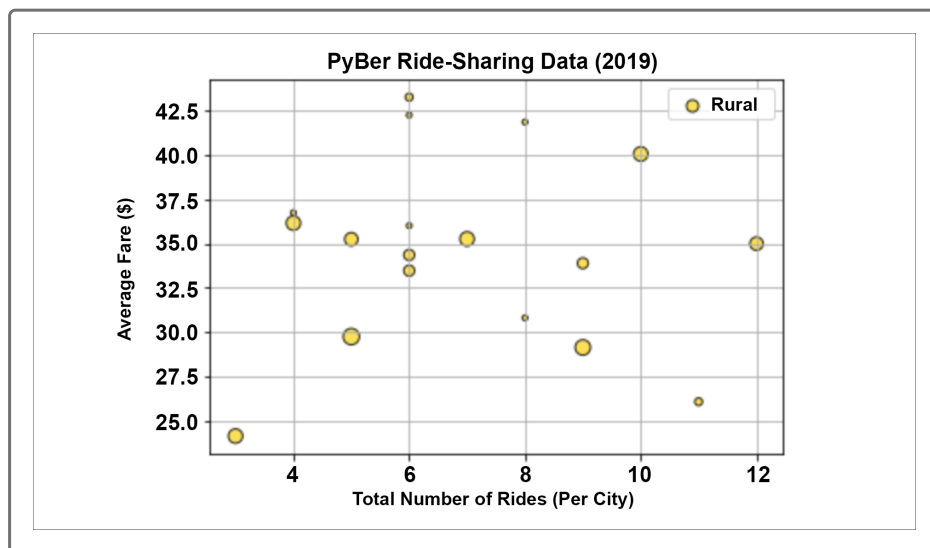
We have two of the three bubble charts. Let's create the final bubble chart: the rural cities bubble chart. Repurposing the code from the suburban cities bubble chart, we'll change the x- and y-axes variables and the size parameter to the rural city data, and we will change the color of the circle to gold.

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

```
# Build the scatter plots for rural cities.
plt.scatter(rural_ride_count,
            rural_avg_fare,
            s=10*rural_driver_count, c="gold",
            edgecolor="black", linewidths=1,
            alpha=0.8, label="Rural")
plt.title("PyBer Ride-Sharing Data (2019)")
plt.ylabel("Average Fare ($)")
plt.xlabel("Total Number of Rides (Per City)")
plt.grid(True)
```

```
# Add the legend.  
plt.legend()
```

When you run the cell, you'll see the following chart with gold markers that vary in diameter based on the average number of drivers in each city, a title, and axes labels.



Congratulations on creating the individual bubble charts for each city type! Next, we will combine these three charts into one chart.