**Import**

# Dependencies

%matplotlib inline

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

**DataFrame**

# File to Load (Remember to change these)

city\_data\_to\_load = "Resources/city\_data.csv"

ride\_data\_to\_load = "Resources/ride\_data.csv"

# Read the City and Ride Data

city\_data = pd.read\_csv(city\_data\_to\_load)

ride\_data = pd.read\_csv(ride\_data\_to\_load)

# Combine the data into a single dataset

city\_data\_drop = city\_data.drop\_duplicates('city',keep='first')

combine\_data = city\_data\_drop.merge(ride\_data,on='city',how='outer')

# Display the data table for preview

combine\_data.head()

**Calculation**

#groupby the city name to set as index

city\_group = combine\_data.groupby('city')

# count rides by city for x

ride\_count = city\_group['ride\_id'].count()

# ave fare by city for y

ave\_fare = city\_group.mean()['fare']

# count drivers

driver\_count = city\_group.mean()['driver\_count']

# add type as well to create three scatter plots by type

city\_type = city\_data.set\_index('city')['type']

# create dictionary dataframe

summary = pd.DataFrame({

"Number of Rides": ride\_count,

"Average Fare": ave\_fare,

"Number of Drivers": driver\_count,

"City Type": city\_type

})

summary.head()

#separate three dfs to create three categorized scatter plots

urban = summary[summary['City Type'] == 'Urban']

suburban = summary[summary['City Type'] == 'Suburban']

rural = summary[summary['City Type'] == 'Rural']

rural.head()

**Scatter**

# three scatter plots for each city type

plt.scatter(rural['Number of Rides'], rural['Average Fare'], s = rural['Number of Drivers']\*10, color = '#FFD700', edgecolor = 'black', label = 'Rural', alpha = .75)

plt.scatter(suburban['Number of Rides'], suburban['Average Fare'], s = suburban['Number of Drivers']\*10, color = '#87CEFA', edgecolor = 'black', label = 'Suburban', alpha = .75)

plt.scatter(urban['Number of Rides'], urban['Average Fare'], s = urban['Number of Drivers']\*10, color = '#F08080', edgecolor = 'black', label = 'Urban', alpha = .75)

#print scatter plot

plt.title('Pyber Ride Sharing Data (2016)')

plt.xlabel('Number of Rides per City')

plt.ylabel('Average Fare per City')

lgnd = plt.legend()

lgnd.set\_title('City Types')

lgnd.legendHandles[0].\_sizes = [50]

lgnd.legendHandles[1].\_sizes = [50]

lgnd.legendHandles[2].\_sizes = [50]

plt.grid(True)

**Pie1**

#group by type to be used by all pie charts

by\_type = combine\_data.groupby('type')['type', 'fare', 'ride\_id', 'driver\_count']

# sum fare by city

fare\_sum = by\_type.sum()['fare']

fare\_sum

# plot pie chart

plt.pie(fare\_sum, startangle = 150, colors = ["#FFD700","#87CEFA","#F08080"], explode = (0,0,0.1), labels = fare\_sum.index, autopct = "%1.1f%%", shadow = True, wedgeprops = {'linewidth': .1, 'edgecolor': 'black'})

# memo = plt.axis('equal') = change the shape

plt.title('% of Total Fares by City Type')

plt.show()

**Pie2**

# number of rides by city type

ride\_sum = by\_type.count()['ride\_id']

plt.pie(ride\_sum, startangle = 150, explode = (0,0,0.1), colors = ["#FFD700","#87CEFA","#F08080"], labels = ride\_sum.index, autopct = "%1.1f%%", shadow = True, wedgeprops = {'linewidth': .1, 'edgecolor': 'black'})

plt.title('% of Total Rides by City Type')

plt.show()

**Pie3**

# Number of driver by city type

driver\_sum = city\_data.groupby('type').sum()['driver\_count']

plt.pie(driver\_sum, startangle = 150, explode = (0,0,0.1), colors = ["#FFD700","#87CEFA","#F08080"], labels = ride\_sum.index, autopct = "%1.1f%%", shadow = True, wedgeprops = {'linewidth': .1, 'edgecolor': 'black'})

plt.title('% of Total Drivers by City Type')

plt.show()