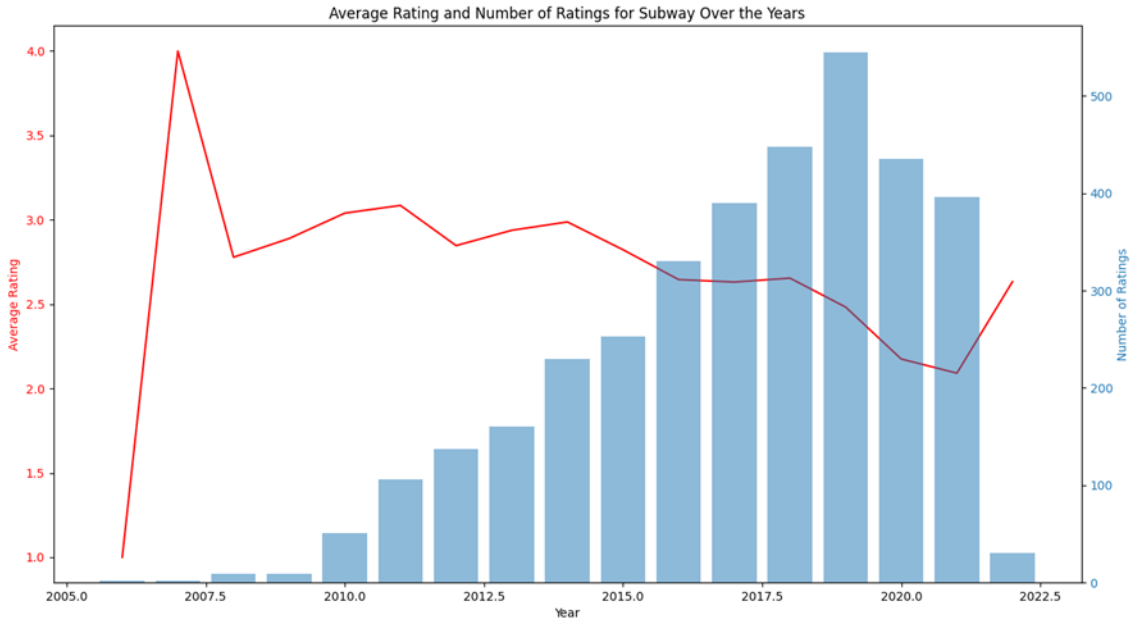
Q1:

```
import pandas as pd
import matplotlib.pyplot as plt

# Read the CSV files
reviews_df = pd.read_csv('C:/Users/Patrick/Downloads/reviews.csv')
restaurants_df = pd.read_csv('C:/Users/Patrick/Downloads/restaurants.csv')

# Merge the two datasets on the common column 'business_id'
merged_df = pd.merge(reviews_df, restaurants_df, on='business_id', how='inner')

# Filter data to include only "Subway" restaurants
subway_df = merged_df[merged_df['name'] == 'Subway']

# Convert the 'date' column to datetime format using pandas
subway_df['date'] = pd.to_datetime(subway_df['date'])

# Extract the year from the date using Pandas
subway_df['year'] = subway_df['date'].dt.year

# Calculate average rating and number of ratings per year
avg_rating_per_year = subway_df.groupby('year')['stars'].mean()
num_ratings_per_year = subway_df.groupby('year').size()

# Create a subplot with two y-axes
fig, ax1 = plt.subplots()

# Plot average rating on the primary y-axis
color = 'red'
```

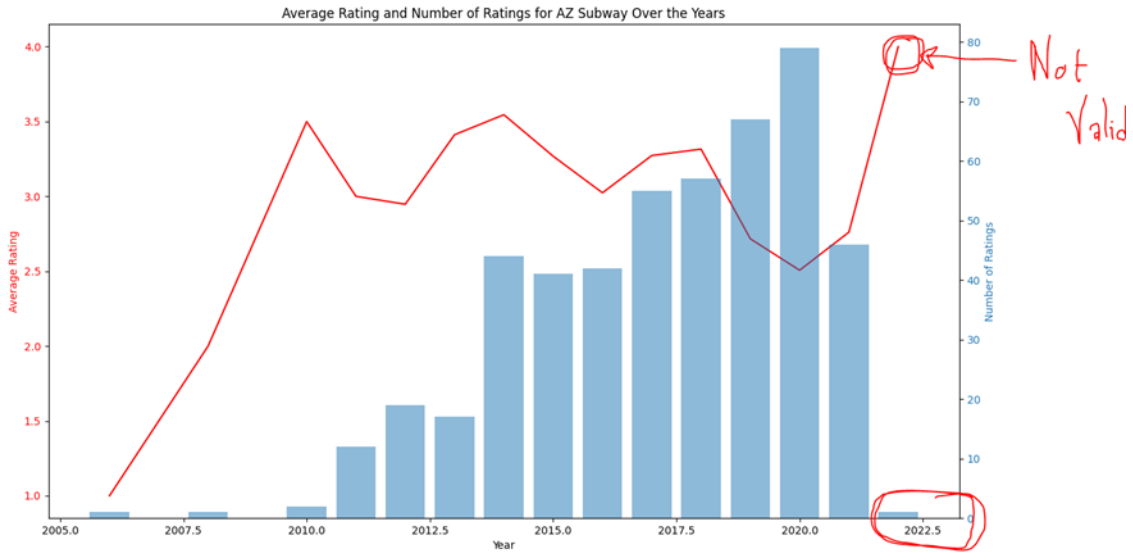
```
ax1.set_xlabel('Year')
ax1.set_ylabel('Average Rating', color=color)
ax1.plot(avg_rating_per_year.index, avg_rating_per_year, color=color)
ax1.tick_params(axis='y', labelcolor=color)

# Create a secondary y-axis to plot the number of ratings
ax2 = ax1.twinx()
color = 'tab:blue'
ax2.set_ylabel('Number of Ratings', color=color)
ax2.bar(num_ratings_per_year.index, num_ratings_per_year, alpha=0.5, color=color)
ax2.tick_params(axis='y', labelcolor=color)

# Show the plot
plt.title('Average Rating and Number of Ratings for Subway Over the Years')
plt.show()
```

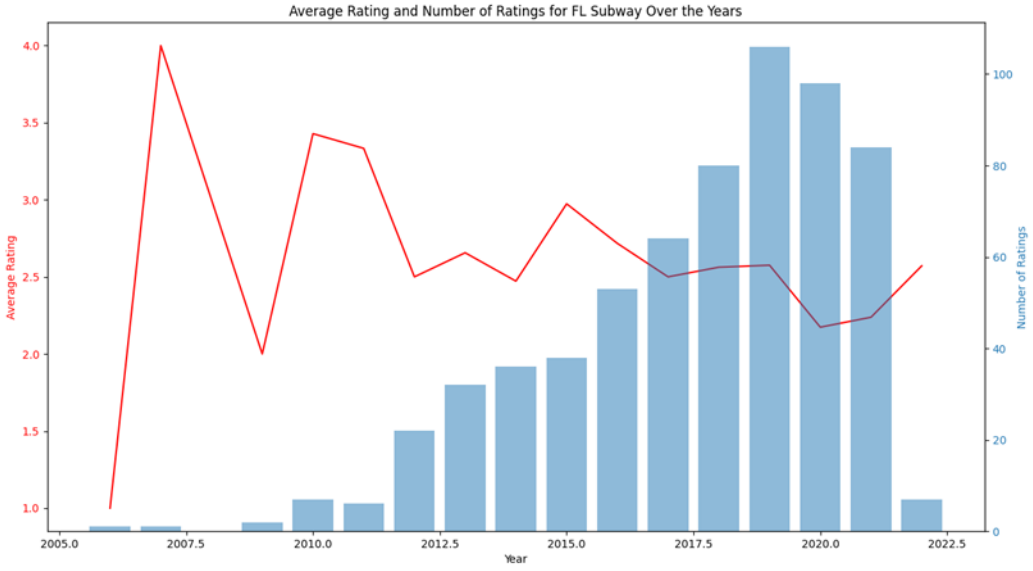


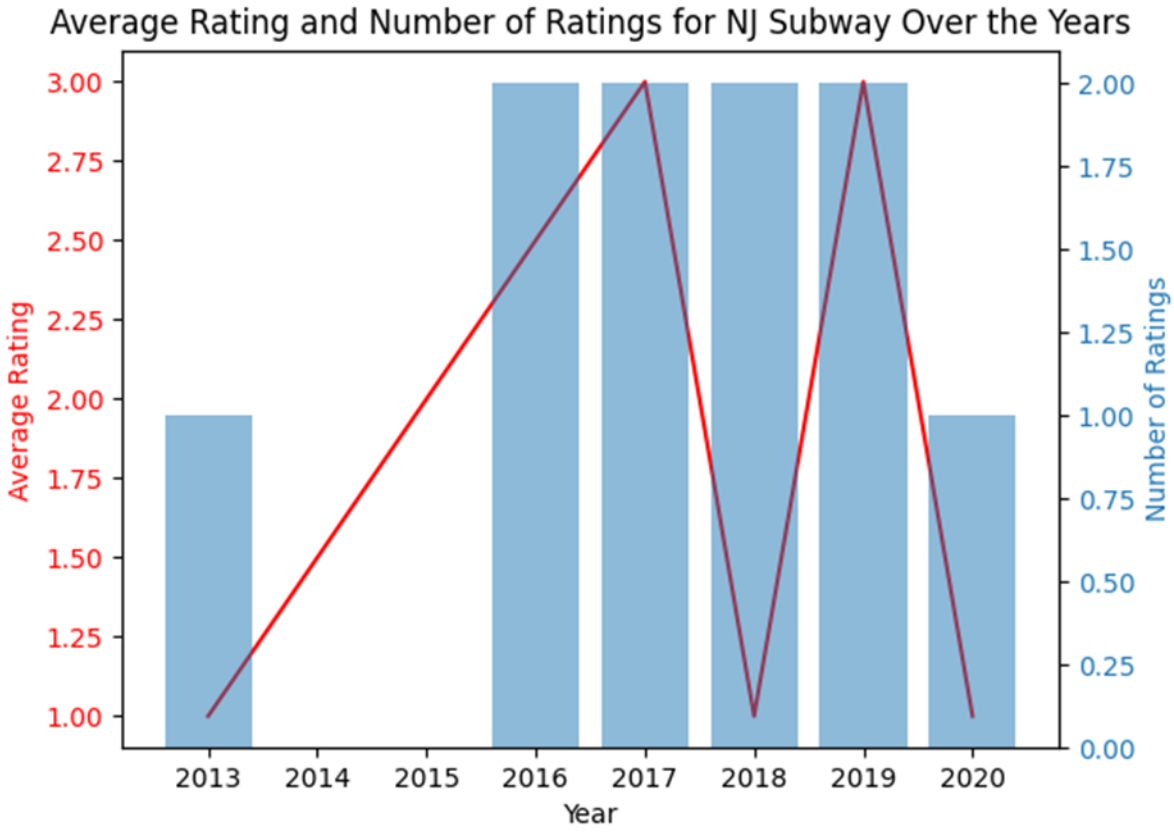
Q1 (Cont.):

```
import pandas as pd
import matplotlib.pyplot as plt

# Read the CSV files
reviews_df = pd.read_csv('C:/Users/Patrick/Downloads/reviews.csv')
restaurants_df = pd.read_csv('C:/Users/Patrick/Downloads/restaurants.csv')

# Merge the two datasets on the common column 'business_id'
merged_df = pd.merge(reviews_df, restaurants_df, on='business_id', how='inner')

# Filter data to include only "Subway" restaurants in the state of NJ
subway_df = merged_df[(merged_df['name'] == 'Subway') & (merged_df['state'] == 'AZ/NJ/FL')]

# Convert the 'date' column to datetime format using pandas
subway_df['date'] = pd.to_datetime(subway_df_nj['date'])

# Extract the year from the date using Pandas
subway_df['year'] = subway_df_nj['date'].dt.year

# Calculate average rating and number of ratings per year
avg_rating_per_year = subway_df.groupby('year')['stars'].mean()
num_ratings_per_year = subway_df.groupby('year').size()

# Create a subplot with two y-axes
fig, ax1 = plt.subplots()

# Plot average rating on the primary y-axis
color = 'red'
```

```
ax1.set_xlabel('Year')
ax1.set_ylabel('Average Rating', color=color)
ax1.plot(avg_rating_per_year.index, avg_rating_per_year, color=color)
ax1.tick_params(axis='y', labelcolor=color)

# Create a secondary y-axis to plot the number of ratings
ax2 = ax1.twinx()
color = 'tab:blue'
ax2.set_ylabel('Number of Ratings', color=color)
ax2.bar(num_ratings_per_year.index, num_ratings_per_year, alpha=0.5, color=color)
ax2.tick_params(axis='y', labelcolor=color)

# Show the plot
plt.title('Average Rating and Number of Ratings for AZ/NJ/FL Subway Over the Years')
plt.show()
```

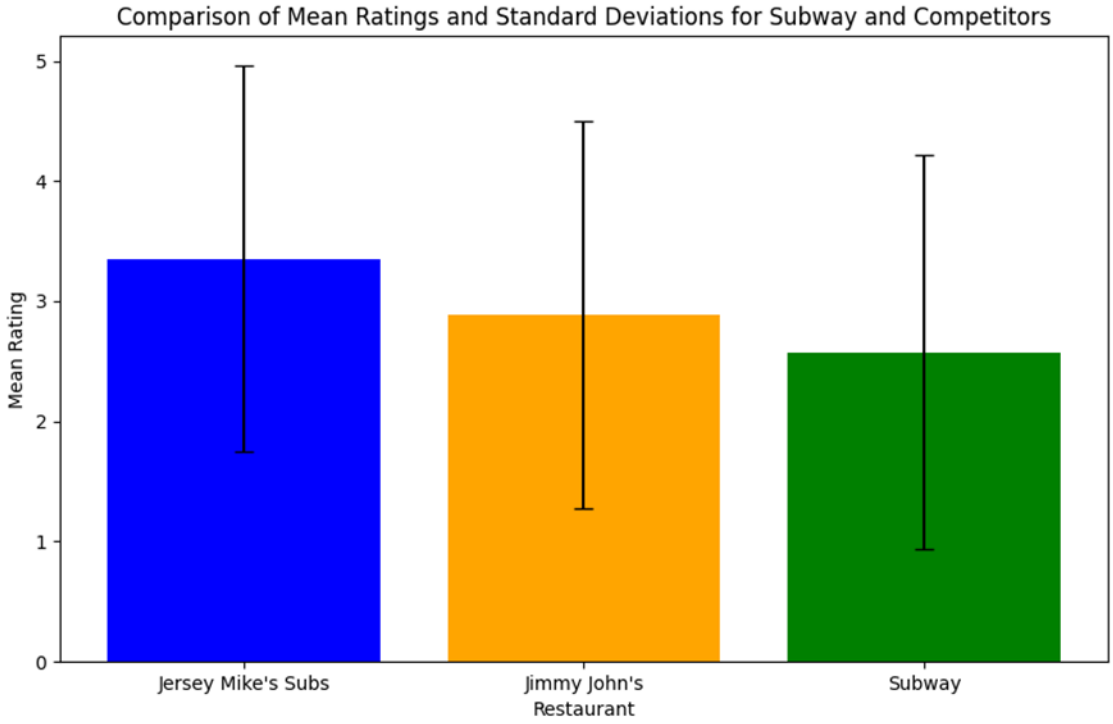



Average Rating and Number of Ratings for FL Subway Over the Years
Average Rating
Number of Ratings
Year
4.0
3.5
3.0
2.5
2.0
1.5
1.0
100
80
60
40
20
0
2005.0
2007.5
2010.0
2012.5
2015.0
2017.5
2020.0
2022.5


Average Rating and Number of Ratings for NJ Subway Over the Years
Average Rating
Number of Ratings
Year
3.00
2.75
2.50
2.25
2.00
1.75
1.50
1.25
1.00
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
2013
2014
2015
2016
2017
2018
2019
2020

Q2:

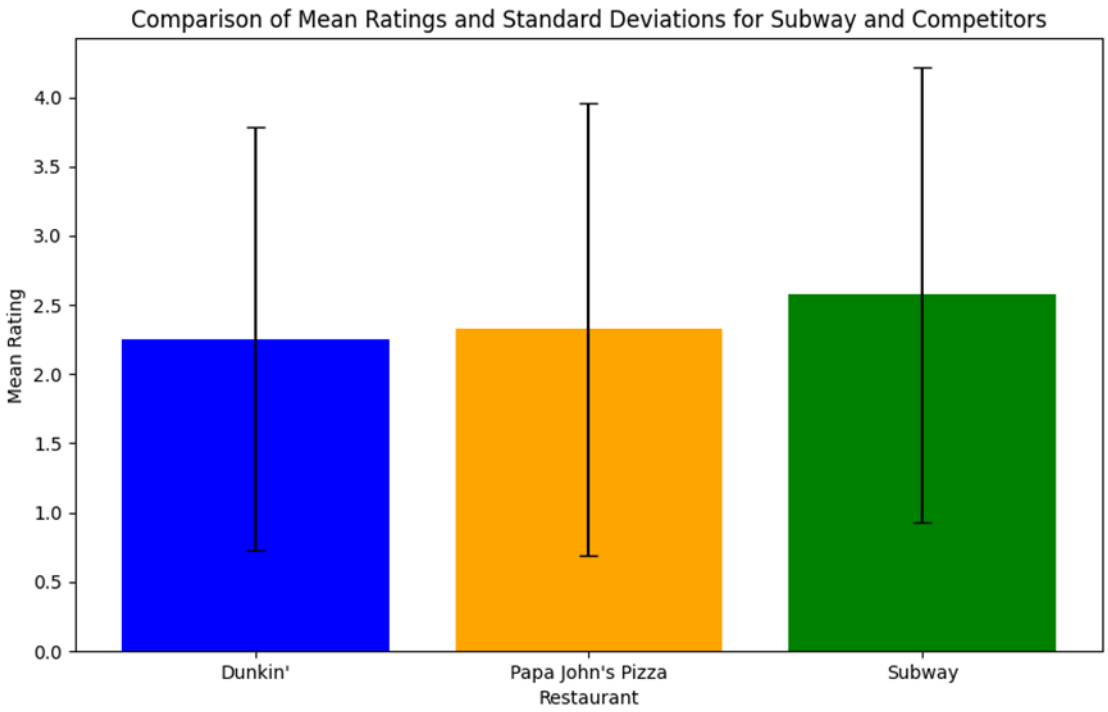
```
import pandas as pd
import matplotlib.pyplot as plt

# Read the CSV files
reviews_df = pd.read_csv('C:/Users/Patrick/Downloads/reviews.csv')
restaurants_df = pd.read_csv('C:/Users/Patrick/Downloads/restaurants.csv')

# Merge the two datasets on the common column 'business_id'
merged_df = pd.merge(reviews_df, restaurants_df, on='business_id', how='inner')

# Define Competitors
competitor1 = "Jersey Mike's Subs"
competitor2 = "Jimmy John's"

# Filter data to include reviews for Subway and its competitors
selected_data = merged_df[merged_df['name'].isin(['Subway', competitor1, competitor2])]

# Calculate mean and standard deviation of reviews for each restaurant
summary_stats = selected_data.groupby('name')['stars'].agg(['mean', 'std']).reset_index()

# Create a bar plot for comparison
plt.figure(figsize=(10, 6))
plt.bar(summary_stats['name'], summary_stats['mean'], yerr=summary_stats['std'], capsize=5,
color=['blue', 'orange', 'green'])
plt.xlabel('Restaurant')
plt.ylabel('Mean Rating')
plt.title('Comparison of Mean Ratings and Standard Deviations for Subway and Competitors')
plt.show()
```



Q2 (cont.):

```
import pandas as pd
import matplotlib.pyplot as plt

# Read the CSV files
reviews_df = pd.read_csv('C:/Users/Patrick/Downloads/reviews.csv')
restaurants_df = pd.read_csv('C:/Users/Patrick/Downloads/restaurants.csv')

# Merge the two datasets on the common column 'business_id'
merged_df = pd.merge(reviews_df, restaurants_df, on='business_id', how='inner')

# Define Competitors
competitor1 = "Dunkin'"
competitor2 = "Papa John's"
```

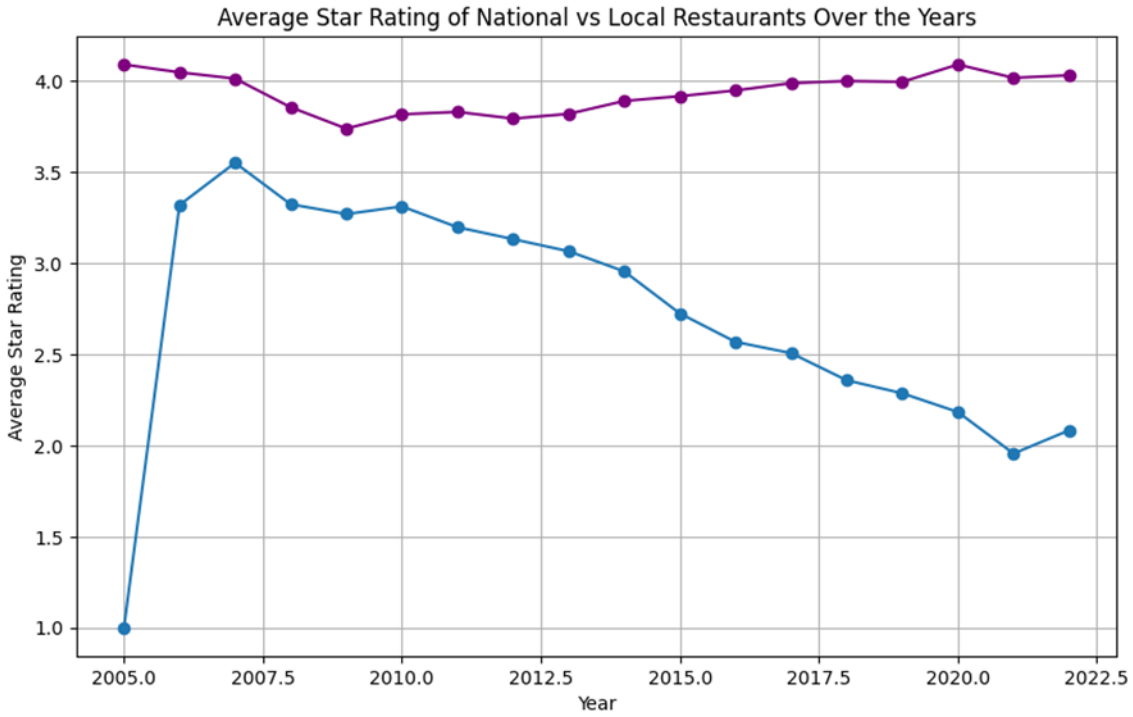
```
# Filter data to include reviews for Subway and its competitors
selected_data = merged_df[merged_df['name'].isin(['Subway', competitor1, competitor2])]

# Calculate mean and standard deviation of reviews for each restaurant
summary_stats = selected_data.groupby('name')['stars'].agg(['mean', 'std']).reset_index()

# Create a bar plot for comparison
plt.figure(figsize=(10, 6))
plt.bar(summary_stats['name'], summary_stats['mean'], yerr=summary_stats['std'], capsize=5, color=['blue', 'orange', 'green'])
plt.xlabel('Restaurant')
plt.ylabel('Mean Rating')
plt.title('Comparison of Mean Ratings and Standard Deviations for Subway and Competitors')
plt.show()
```



Q3:

```
import pandas as pd
import matplotlib.pyplot as plt

# Read the CSV files
reviews_df = pd.read_csv('C:/Users/Patrick/Downloads/reviews.csv')
restaurants_df = pd.read_csv('C:/Users/Patrick/Downloads/restaurants.csv')

# Merge the two datasets on the common column 'business_id'
merged_df = pd.merge(reviews_df, restaurants_df, on='business_id', how='inner')

# Filter restaurants with cities < 50 and category containing 'restaurant'
national_chains = merged_df[
   (merged_df.groupby('name')['city'].transform('nunique') > 50) &
   (merged_df['categories'].str.contains('restaurant', case=False, na=False))
]

local_chains = merged_df[
   (merged_df.groupby('name')['city'].transform('nunique') < 50) &
   (merged_df['categories'].str.contains('restaurant', case=False, na=False))
]

# Convert the 'date' column to datetime format
national_chains['date'] = pd.to_datetime(national_chains['date'])
local_chains['date'] = pd.to_datetime(local_chains['date'])



# Extract the year from the date
national_chains['year'] = national_chains['date'].dt.year
```

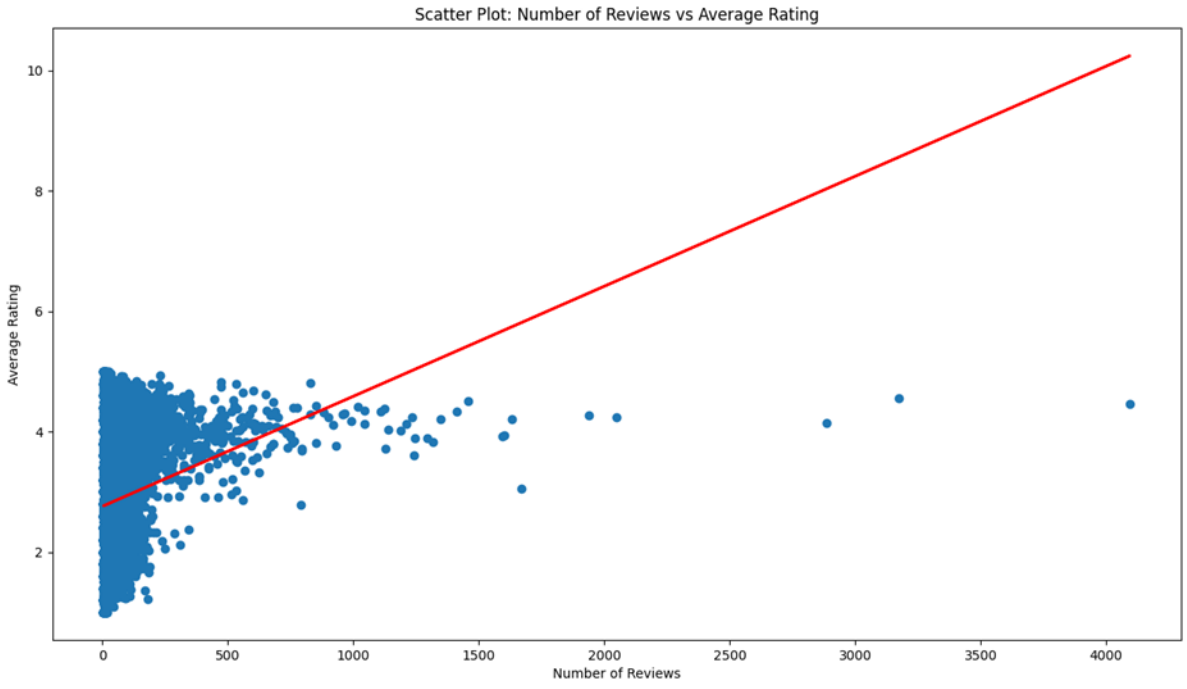
```
local_chains['year'] = local_chains['date'].dt.year


# Group by year and calculate the average star rating
avg_star_by_year_national = national_chains.groupby('year')['stars'].mean()
avg_star_by_year_local = local_chains.groupby('year')['stars'].mean()


# Plotting the results
plt.figure(figsize=(10, 6))
plt.plot(avg_star_by_year_national.index, avg_star_by_year_national, marker='o', linestyle='-')
plt.plot(avg_star_by_year_local.index, avg_star_by_year_local, marker='o', linestyle='-', color = 'purple')
plt.xlabel('Year')
plt.ylabel('Average Star Rating')
plt.title('Average Star Rating of National vs Local Restaurants Over the Years')
plt.grid(True)
plt.show()
```



Q3 (cont.):

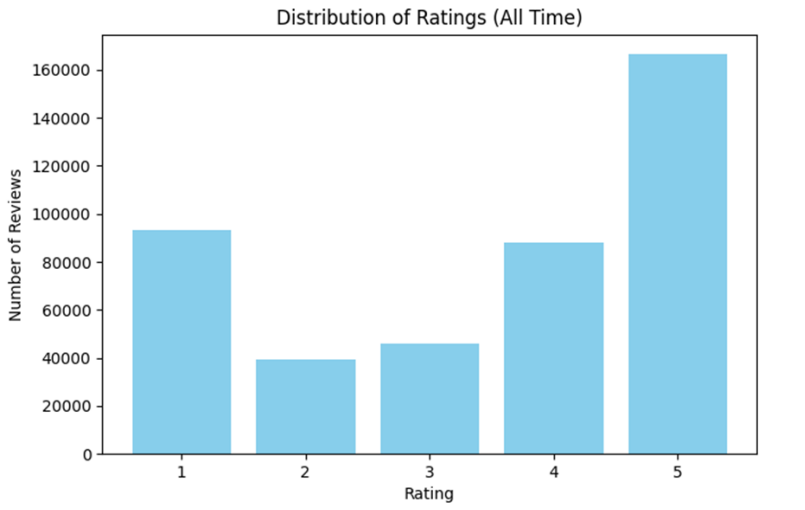
```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np

# Read the CSV files
reviews_df = pd.read_csv('C:/Users/Patrick/Downloads/reviews.csv')
restaurants_df = pd.read_csv('C:/Users/Patrick/Downloads/restaurants.csv')

# Merge the two datasets on the common column 'business_id'
merged_df = pd.merge(reviews_df, restaurants_df, on='business_id', how='inner')

# Calculate the number of reviews for each restaurant
num_reviews_per_restaurant = merged_df.groupby('business_id')['stars'].count()

# Calculate the average rating for each restaurant
avg_rating_per_restaurant = merged_df.groupby('business_id')['stars'].mean()

# Create a DataFrame with the number of reviews and average rating
data = pd.DataFrame({'num_reviews': num_reviews_per_restaurant, 'avg_rating':
avg_rating_per_restaurant})

# Calculate correlation
correlation = data['num_reviews'].corr(data['avg_rating'])
print(f'Correlation between number of reviews and average rating: {correlation}')

# Scatter plot
plt.scatter(data['num_reviews'], data['avg_rating'])
plt.xlabel('Number of Reviews')
```

```
plt.ylabel('Average Rating')
plt.title('Scatter Plot: Number of Reviews vs Average Rating')

# Fit a linear regression line using numpy
x = data['num_reviews'].values
y = data['avg_rating'].values

# Calculate the coefficients (slope and intercept) of the line
slope, intercept = np.polyfit(x, y, 1)

# Plot the line of best fit
plt.plot(x, slope * x + intercept, color='red', linewidth=2)

plt.show()
```



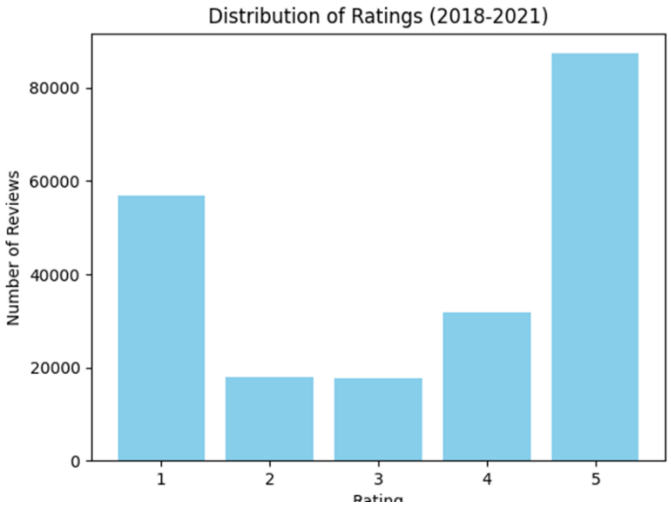
Q4:

```
import pandas as pd

import matplotlib.pyplot as plt


# Read the CSV files

reviews_df = pd.read_csv('C:/Users/Patrick/Downloads/reviews.csv')


# Convert 'date' column to datetime format

reviews_df['date'] = pd.to_datetime(reviews_df['date'])


# Count the number of reviews for each rating

rating_counts = reviews_df['stars'].value_counts().sort_index()


# Plotting

plt.bar(rating_counts.index, rating_counts.values, color='skyblue')

plt.xlabel('Rating')

plt.ylabel('Number of Reviews')

plt.title('Distribution of Ratings (All Time)')

plt.show()
```



Q4 (cont.):

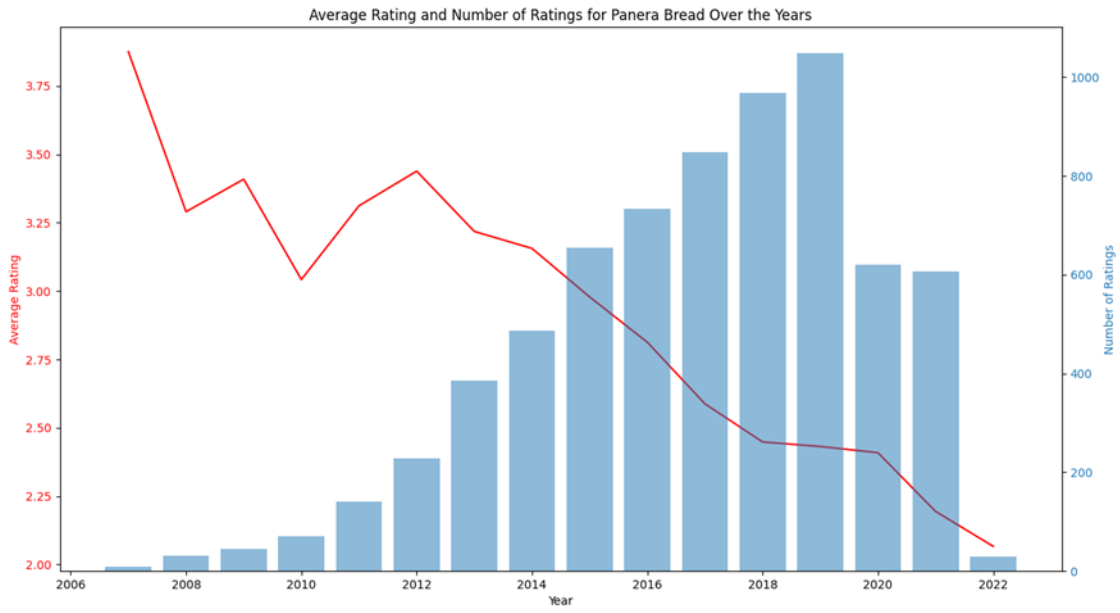
```
import pandas as pd

import matplotlib.pyplot as plt


# Read the CSV files

reviews_df = pd.read_csv('C:/Users/Patrick/Downloads/reviews.csv')


# Convert 'date' column to datetime format

reviews_df['date'] = pd.to_datetime(reviews_df['date'])


# Filter data for the years 2018 to 2021

filtered_reviews = reviews_df[(reviews_df['date'].dt.year >= 2018) & (reviews_df['date'].dt.year <= 2021)]


# Count the number of reviews for each rating

rating_counts = filtered_reviews['stars'].value_counts().sort_index()


# Plotting

plt.bar(rating_counts.index, rating_counts.values, color='skyblue')

plt.xlabel('Rating')

plt.ylabel('Number of Reviews')

plt.title('Distribution of Ratings (2018-2021)')

plt.show()
```

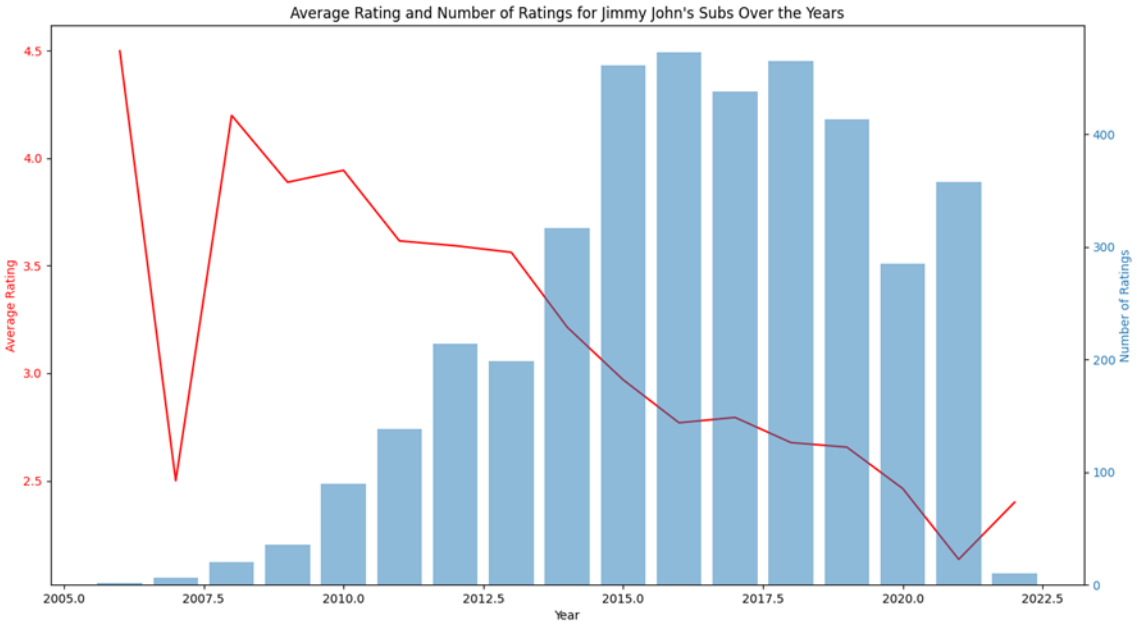


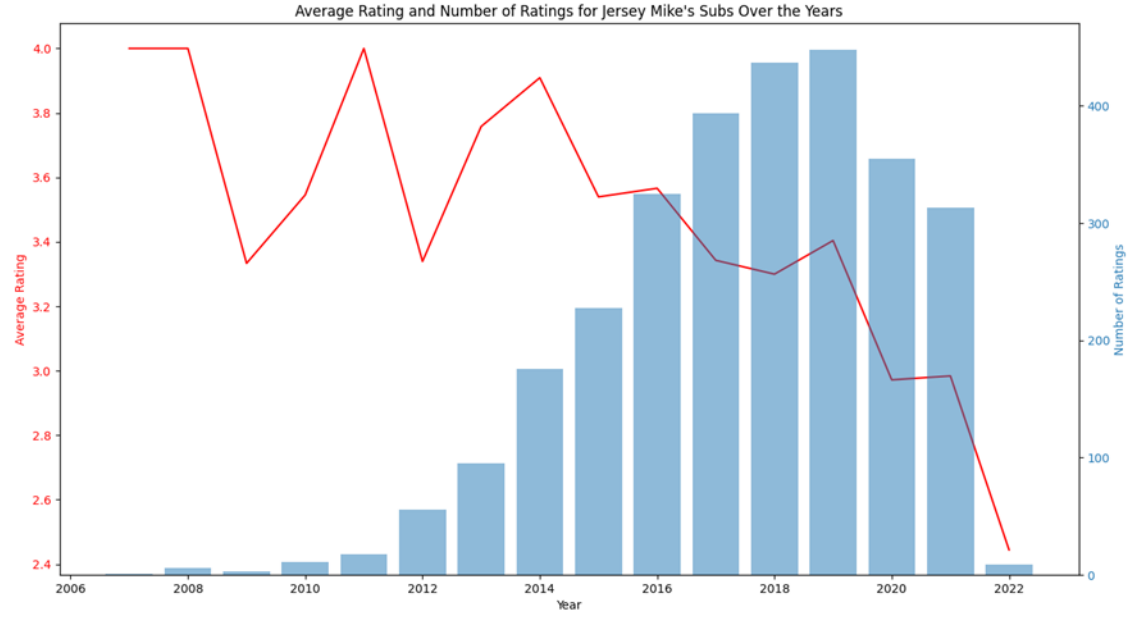
BONUS:

```
import pandas as pd
import matplotlib.pyplot as plt

# Read the CSV files
reviews_df = pd.read_csv('C:/Users/Patrick/Downloads/reviews.csv')
restaurants_df = pd.read_csv('C:/Users/Patrick/Downloads/restaurants.csv')

# Merge the two datasets on the common column 'business_id'
merged_df = pd.merge(reviews_df, restaurants_df, on='business_id', how='inner')

# Filter data to include only "Panera Bread" restaurants
panera_df = merged_df[merged_df['name'] == "Panera Bread"]

# Convert the 'date' column to datetime format using pandas
panera_df['date'] = pd.to_datetime(panera_df['date'])

# Extract the year from the date using Pandas
panera_df['year'] = panera_df['date'].dt.year

# Calculate average rating and number of ratings per year
avg_rating_per_year = panera_df.groupby('year')['stars'].mean()
num_ratings_per_year = panera_df.groupby('year').size()

# Create a subplot with two y-axes
fig, ax1 = plt.subplots()

# Plot average rating on the primary y-axis
color = 'red'
```

```
ax1.set_xlabel('Year')
ax1.set_ylabel('Average Rating', color=color)
ax1.plot(avg_rating_per_year.index, avg_rating_per_year, color=color)
ax1.tick_params(axis='y', labelcolor=color)

# Create a secondary y-axis to plot the number of ratings
ax2 = ax1.twinx()
color = 'tab:blue'
ax2.set_ylabel('Number of Ratings', color=color)
ax2.bar(num_ratings_per_year.index, num_ratings_per_year, alpha=0.5, color=color)
ax2.tick_params(axis='y', labelcolor=color)

# Show the plot
plt.title("Average Rating and Number of Ratings for Panera Bread Over the Years")
plt.show()
```




Average Rating and Number of Ratings for Jimmy John's Subs Over the Years
Average Rating
Number of Ratings
Year
4.5
4.0
3.5
3.0
2.5
400
300
200
100
0
2005.0
2007.5
2010.0
2012.5
2015.0
2017.5
2020.0
2022.5


Average Rating and Number of Ratings for Jersey Mike's Subs Over the Years
Average Rating
Number of Ratings
Year
4.0
3.8
3.6
3.4
3.2
3.0
2.8
2.6
2.4
400
300
200
100
0
2006
2008
2010
2012
2014
2016
2018
2020
2022