Final Assignment – Konstantinos Karagiovanis

Following is the report (in steps) for the final assignment of the Python Data Bootcamp 2023.

Steps

1. I set up a new connection in Workbench in order to load the provided sql file.

2. After loading the sql file, I ran the following query to the get all the columns of the table between the years 2016-2019:

```
SELECT * FROM finance_liquor_sales WHERE date BETWEEN '2016-01-01 00:00:00' AND '2019-12-31 00:00:00'
```

3. I exported the returned data from workbench, to csv format.

4. I opened VSCode, which I used to build the task, and created a new project.

5. After that I created a virtual environment into the project folder by running the following command, and activated it by running the activate.bat file that was generated inside script folder:

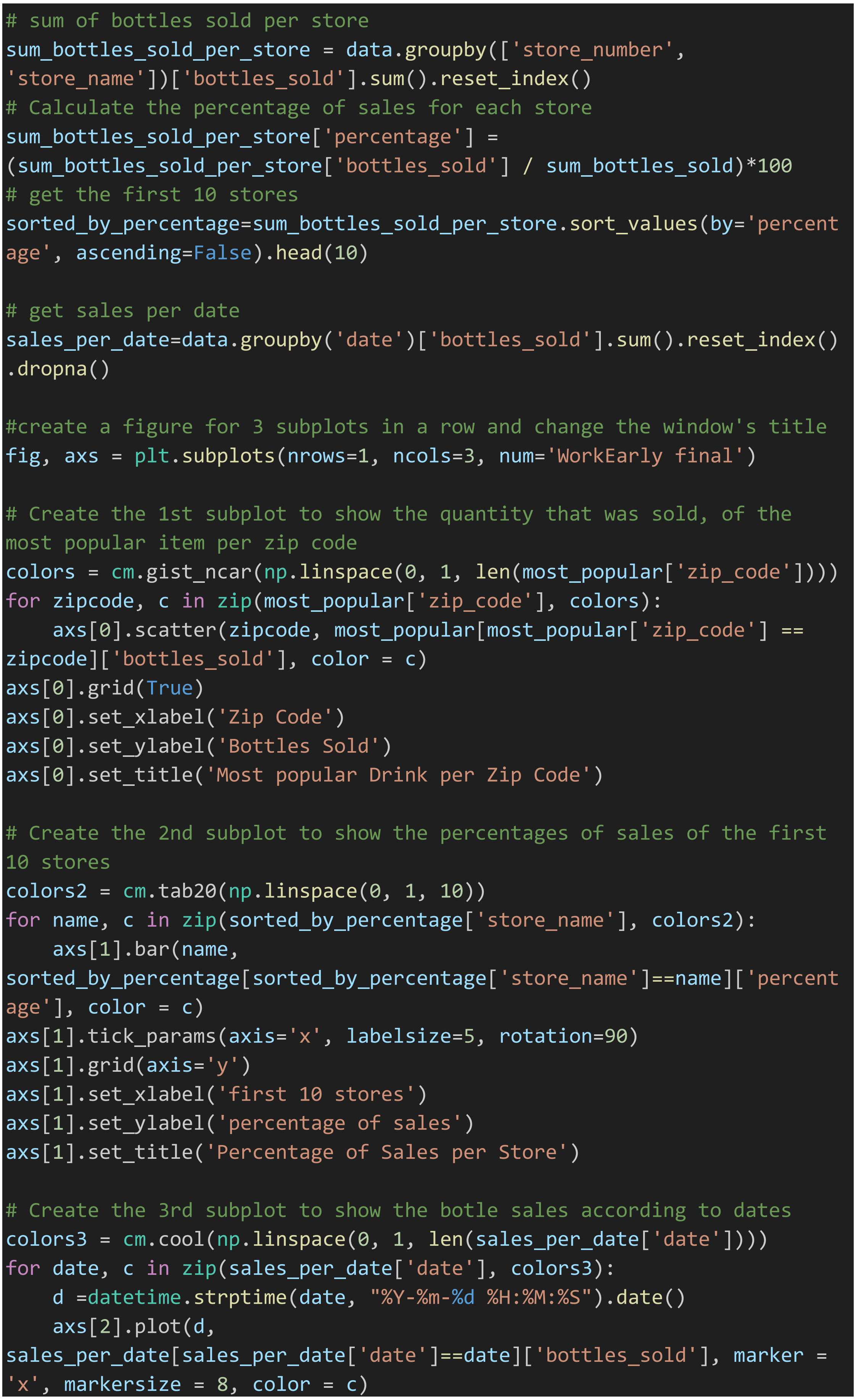
```
python -m venv liquorvenv
```

6. Next step was to install all the needed libraries.

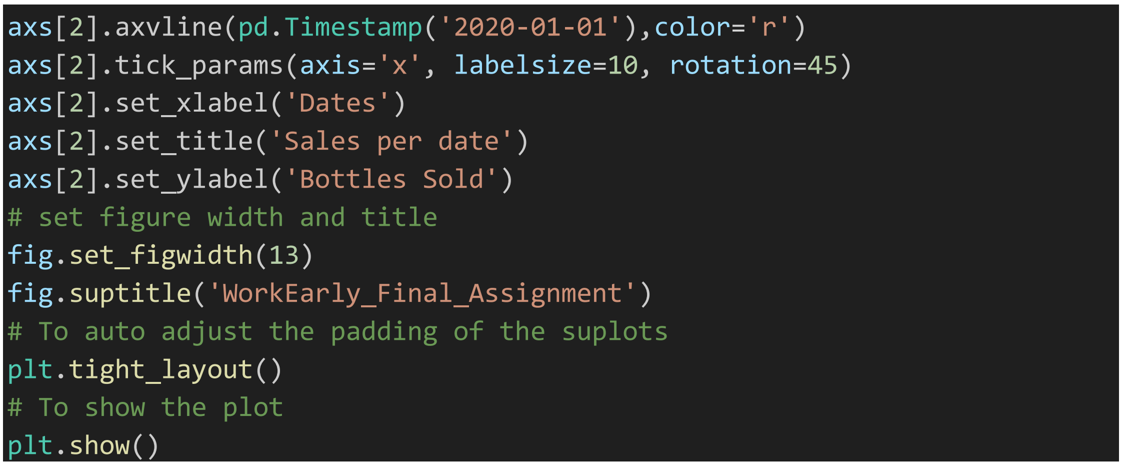
```
pip install pandas

pip install matplotlib
```

7. At this point we are ready to write the code for the assignment.

```
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib.cm as cm
from datetime import datetime
import numpy as np

# read data from csv
data = pd.read_csv('finance_liquor_sales.csv')

# get the most popular item per zip code
most_popular_zip_code = data.groupby(['zip_code', 'item_description'])['bottles_sold'].max().reset_index()
most_popular = most_popular_zip_code.loc[most_popular_zip_code.groupby('zip_code')['bottles_sold'].idxmax()]

# get the percentage of sales per store
# sum of all bottles sold
sum_bottles_sold = data['bottles_sold'].sum()
```

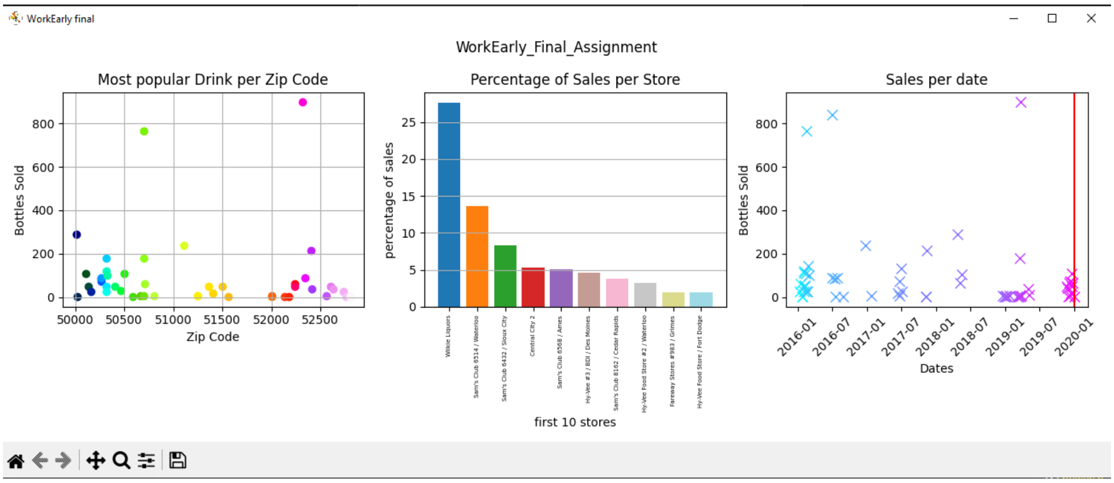


```python
# sum of bottles sold per store
sum_bottles_sold_per_store = data.groupby(['store_number', 'store_name'])['bottles_sold'].sum().reset_index()
# Calculate the percentage of sales for each store
sum_bottles_sold_per_store['percentage'] = (sum_bottles_sold_per_store['bottles_sold'] / sum_bottles_sold)*100
# get the first 10 stores
sorted_by_percentage=sum_bottles_sold_per_store.sort_values(by='percentage', ascending=False).head(10)

# get sales per date
sales_per_date=data.groupby('date')['bottles_sold'].sum().reset_index().dropna()

#create a figure for 3 subplots in a row and change the window's title
fig, axs = plt.subplots(nrows=1, ncols=3, num='WorkEarly final')

# Create the 1st subplot to show the quantity that was sold, of the most popular item per zip code
colors = cm.gist_ncar(np.linspace(0, 1, len(most_popular['zip_code'])))
for zipcode, c in zip(most_popular['zip_code'], colors):
    axs[0].scatter(zipcode, most_popular[most_popular['zip_code'] == zipcode]['bottles_sold'], color = c)
axs[0].grid(True)
axs[0].set_xlabel('Zip Code')
axs[0].set_ylabel('Bottles Sold')
axs[0].set_title('Most popular Drink per Zip Code')

# Create the 2nd subplot to show the percentages of sales of the first 10 stores
colors2 = cm.tab20(np.linspace(0, 1, 10))
for name, c in zip(sorted_by_percentage['store_name'], colors2):
    axs[1].bar(name, sorted_by_percentage[sorted_by_percentage['store_name']==name]['percentage'], color = c)
axs[1].tick_params(axis='x', labelsize=5, rotation=90)
axs[1].grid(axis='y')
axs[1].set_xlabel('first 10 stores')
axs[1].set_ylabel('percentage of sales')
axs[1].set_title('Percentage of Sales per Store')

# Create the 3rd subplot to show the botle sales according to dates
colors3 = cm.cool(np.linspace(0, 1, len(sales_per_date['date'])))
for date, c in zip(sales_per_date['date'], colors3):
    d =datetime.strptime(date, "%Y-%m-%d %H:%M:%S").date()
    axs[2].plot(d, sales_per_date[sales_per_date['date']==date]['bottles_sold'], marker = 'x', markersize = 8, color = c)
```



```
axs[2].axvline(pd.Timestamp('2020-01-01'),color='r')
axs[2].tick_params(axis='x', labelsize=10, rotation=45)
axs[2].set_xlabel('Dates')
axs[2].set_title('Sales per date')
axs[2].set_ylabel('Bottles Sold')
# set figure width and title
fig.set_figwidth(13)
fig.suptitle('WorkEarly_Final_Assignment')
# To auto adjust the padding of the suplots
plt.tight_layout()
# To show the plot
plt.show()
```

8. I decided to use Matplotlib to present the data. Following is the figure that is produced when you run the code.



Difficulties

1. The first challenge was to get the needed data from the csv file using pandas. I did a lot of research in the material of the course and also searched for examples on the Internet.

2. Another challenge was to make the graphs as user friendly as possible, which also included almost the same time in research.

Konstantinos Karagiovanis
Mobile: +30 6936711440
Email: k.n.karagiovanis@gmail.com