Question 1:

from matplotlib import pyplot as plt

import numpy as np

import json

f = open("number.txt", "r")

data = f.read()

#remove lines and replace with a comma, then split this string into a list

data\_into\_list = data.replace('\n', ',').split(',')

#convert the data type into integers

for i in range(0, len(data\_into\_list)):

data\_into\_list[i] = int(data\_into\_list[i])

#sort the list from smallest to largest

numbers\_sorted = sorted(data\_into\_list)

print(numbers\_sorted)

frequency = {}

#count the frequency

for item in numbers\_sorted:

#the number already exists in the dictionary, the frequency will plus one

if item in frequency:

frequency[item] += 1

#if the number does not exist in the dictionary, it will be added and frequence counted as one

else:

frequency[item] = 1

#iterate and print output in the required format

for key, value in frequency.items():

print ("% d : % d"%(key, value))

print('explanation: here ', end = '')

for key, value in frequency.items():

print (f'{key} occurs {value} times,', end = '')

#plot a histogram using the sorted list

counts, edges, bars = plt.hist(numbers\_sorted, bins=11)

plt.bar\_label(bars)

plt.show()

with open('C:/Users/fredl/Desktop/stat1129/data.json', 'w') as fp:

json.dump(frequency, fp)

Question 2:

iimport matplotlib.pyplot as plt

import numpy as np

import pandas as pd

df = pd.read\_csv('amazon\_orders.csv')

df = df.fillna(0)

df = df.drop(0)

#remove dollar sign and convert those strings to float

df["Item Total"] = df["Item Total"].str.replace('$','').astype(float)

df["Item Subtotal Tax"] = df["Item Subtotal Tax"].str.replace('$','').astype(float)

df["Item Subtotal"] = df["Item Subtotal"].str.replace('$','').astype(float)

#Convert Date to datetime data type

df['Order Date'] = pd.to\_datetime(df['Order Date'])

df

#statistical analysis

#Calculate the total cost for all orders

df["Item Total"].sum()

#Calculate the average cost for each order

df["Item Total"].mean()

#Calculate the median cost for all orders

df["Item Total"].median()

#Gives the cost of the most expensive order

df["Item Total"].max()

#Gives the cost of the cheapest order

df["Item Total"].min()

#Locate corresponding data using loc method and sum up using sum method

Feb = df.loc['1':'22', ['Item Total']].sum()["Item Total"]

Mar = df.loc['23':'35', ['Item Total']].sum()["Item Total"]

print (f'Feberuary total cost {Feb}')

print (f'March total cost {Mar}')

#Gives the standard deviation of cost of all orders

df["Item Total"].std()

#Gives the total cost to pay tax

df["Item Subtotal Tax"].sum()

#Gives the proportion of tax in total cost

df["Item Subtotal Tax"].sum() / df["Item Total"].sum()

#barchart

#Group total cost by date

daily\_orders = df.groupby('Order Date').sum()["Item Total"]

#Graph a bar chart to show the cost of all orders on specific dates

daily\_orders.plot.bar(x='Order Date', y='Item Total', rot=90)

#pie chart

explode = [0,0.1,0,0,0,0,0,0,0,0,0,0,0,0.2,0,0,0,0,0,0,0,0]

#graph a bar chart to show the amount paied to each Seller

sum\_by\_seller = df.groupby(['Seller']).sum().plot.pie(y='Item Subtotal', radius = 1.4, explode = explode,autopct = "%0.2f%%")

plt.legend(title = 'Payment to each Seller', bbox\_to\_anchor=(1.5,1.0),loc = 'upper left')

plt.show()

#multiple line comparison

#Create a sublist containing tax and total cost for each order

df1 = df[['Item Subtotal Tax', 'Item Total']]

#Plot a multiple line comparison graph with date on the x axis and cost on the y axis

plt.figure(figsize=(50,40))

df1.plot()

plt.legend()