**Andrew Wang**

**Homework 9**

1. **Code:**

################

# Author: Andrew Wang

# Date: 11/10/2019

# This programs allows user to input sales for each month of the year and creates a pie chart

#################

import matplotlib.pyplot as plt

#Main function

def main():

#Initializing variable

month = ['January', 'February', 'March', 'April', 'May', 'June', 'July', 'August', 'September', 'October', 'November', 'December']

sales = []

color = ('r', 'g', 'b', 'y', 'm', 'w', 'k', 'blue', 'pink', 'brown', 'ivory', 'black')

#For loop that allows user to input sales for each month

for i in range(12):

values = int(input('Enter the sales value of ' +month[i]+ ': '))

sales.append(values)

#Creates pie chart

plt.pie(sales, labels=month, colors=color)

plt.title('Monthly Sales Values')

plt.savefig('Q1.png')

plt.show()

#Calls main function

main()

**Output:**A close up of a logo

Description automatically generated

1. **Code:**

################

# Author: Andrew Wang

# Date: 11/10/2019

# This programs reads the file of US population during 1950 - 1990 and plots it

# in a line chart and bar chart

#################

import matplotlib.pyplot as plt

#Main function

def main():

#Opens and reads file

gas = open('USPopulation.txt', 'r')

line = gas.readlines()

#Initializing variable

year = []

#For loop that reads file and put it into a list

for i in range(len(line)):

line[i] = line[i].rstrip('\n')

line[i] = float(line[i])

year = list(range(1950, len(line) + 1950))

line\_graph(year, line)

bar\_graph(year, line)

gas.close()

#Function that plots data into line chart

def line\_graph(year, line):

plt.plot(year, line)

plt.title('U.S. Population from 1950 to 1990')

plt.xlabel('Year')

plt.ylabel('Population')

plt.grid()

plt.xticks([1950,1960,1970,1980,1990])

plt.yticks([150000, 170000, 190000, 210000, 230000, 250000], ['150M', '170M', '190M', '210M', '230M', '250M'])

plt.savefig('Q2\_line.png')

plt.show()

#Function that plots data into bar chart

def bar\_graph(year, line):

bar\_width = 0.5

colors = ('b')

plt.bar(year, line, bar\_width, color=colors, align='center')

plt.title('U.S. Population from 1950 to 1990')

plt.xlabel('Year')

plt.ylabel('Population')

plt.xticks([1950,1960,1970,1980,1990])

plt.yticks([150000, 170000, 190000, 210000, 230000, 250000], ['150M', '170M', '190M', '210M', '230M', '250M'])

plt.ylim(bottom = 150000, top = 250000)

plt.xlim(left = 1948, right = 1992)

plt.savefig('Q2\_bar.png')

plt.show()

#Calls main function

main()

**Output:**A screenshot of a cell phone

Description automatically generated

A close up of a logo

Description automatically generated

1. **Code:**

################

# Author: Andrew Wang

# Date: 11/10/2019

# This programs reads the file of gas prices for each week in 1994 and plots it

# in a line chart and bar chart

#################

import matplotlib.pyplot as plt

#Main function

def main():

#Opens and reads file

gas = open('1994\_Weekly\_Gas\_Averages.txt', 'r')

line = gas.readlines()

#Initializing variable

year = []

#For loop at reads file and put it into a list

for i in range(len(line)):

line[i] = line[i].rstrip('\n')

line[i] = float(line[i])

week = list(range(1, len(line) + 1))

line\_graph(week, line)

bar\_graph(week, line)

gas.close()

#Function that plots data into line chart

def line\_graph(week, line):

plt.plot(week, line)

plt.title('1994 Weekly Gas Prices')

plt.xlabel('Weeks (by number)')

plt.ylabel('Average Prices')

plt.grid()

plt.savefig('Q3\_line.png')

plt.show()

#Function that plots data into bar chart

def bar\_graph(week, line):

bar\_width = 0.5

colors = ('b')

plt.bar(week, line, bar\_width, color=colors, align='center')

plt.xlabel('Weeks (by number)')

plt.ylabel('Average Prices')

plt.xticks([0,10,20,30,40,50])

plt.yticks([1,1.025,1.050,1.075,1.1,1.125,1.15])

plt.ylim(bottom = 0.975, top = 1.17)

plt.savefig('Q3\_bar.png')

plt.show()

#Calls main function

main()

**Output:**

**A close up of text on a white background

Description automatically generatedA close up of a logo

Description automatically generated**

1. **Code:**

################

# Author: Andrew Wang

# Date: 11/10/2019

# This programs simulates a Magic 8 Ball and allows user to input question which

# it gives a random response

#################

import random

#Main function

def main():

#Opens and reads file

ball = open('8\_ball\_responses.txt', 'r',encoding='utf8',errors="ignore")

line = ball.readlines()

#For loop that reads file and puts it into a list

for i in range(len(line)):

line[i] = line[i].rstrip('\n')

#While function that keeps looping

while True:

number = random.randint(0,len(line))

response = input(str('Enter your question: '))

#If statement that determins if the user wants to quit or not

if response == 'Quit':

print('Thanks for playing.')

break

else:

print(line[number])

#Calls main function

main()

**Output:**

**A screenshot of a cell phone

Description automatically generated**