Project Title:- CU Modul3 3 Challenge

Description:- It’s time to put away the Excel sheet and enter the world of programming with Python. In this assignment, you’ll use the concepts you’ve learned to complete **two** (2) Python challenges, PyBank and PyPoll. Both tasks present a real-world situation where your newly developed Python scripting skills come in handy.

# PyBank

import pandas as pd

import numpy as np

# Load the CSV file into a pandas dataframe

budget\_data = pd.read\_csv('budget\_data.csv')

# Calculate the total number of months in the dataset

total\_months = len(budget\_data['Date'])

# Calculate the net total amount of "Profit/Losses" over the entire period

net\_total = budget\_data['Profit/Losses'].sum()

# Calculate the changes in "Profit/Losses" over the entire period

budget\_data['Profit/Losses Change'] = budget\_data['Profit/Losses'].shift(1) - budget\_data['Profit/Losses']

average\_change = budget\_data['Profit/Losses Change'].mean()

# Find the greatest increase and decrease in profits

greatest\_increase = budget\_data['Profit/Losses Change'].max()

greatest\_increase\_date = budget\_data.loc[budget\_data['Profit/Losses Change'] == greatest\_increase, 'Date'].iloc[0]

greatest\_decrease = budget\_data['Profit/Losses Change'].min()

greatest\_decrease\_date = budget\_data.loc[budget\_data['Profit/Losses Change'] == greatest\_decrease, 'Date'].iloc[0]

# Print out the results

print("Financial Analysis")

print("------------------")

print(f"Total Months: {total\_months}")

print(f"Total: ${net\_total}")

print(f"Average Change: ${average\_change:.2f}")

print(f"Greatest Increase in Profits: {greatest\_increase\_date} (${greatest\_increase})")

print(f"Greatest Decrease in Profits: {greatest\_decrease\_date} (${greatest\_decrease})")

**# PyPoll**

**import csv #importing required libraries**  
**candidates=[] #initialization of empty list for storing candidates**  
**all\_candidates=[] #initialization of empty list for storing votes**  
**percent\_list=[] #initialization of empty list for storing the percentages**  
**data={} #initialization of empty dictionary to store the name of candidate and their votes**  
**total=0 ##initialization variable total as 0 for total votes**  
**with open('election\_data.csv', 'r') as file: #opening the elections\_data.cvv**  
**reader = csv.reader(file) #reading the total file**  
**for row in reader: #iterating through each row**  
**total+=1 #increment the total based on rows**  
**all\_candidates.append(row[2]) #appending the vote to all\_candidates**  
**if row[2] in candidates: #matching**  
**continue**  
**else: #else condition starts**  
**candidates.append(row[2]) # appending the name of candidates i election**  
**total-=1**  
**for i in range(1,len(candidates)): #loop for counting the votes of each candidate**  
**count=0 #count variable to store votes count of individual candidate**  
**for j in range(len(all\_candidates)):**  
**if candidates[i]==all\_candidates[j]:**  
**count+=1**  
**data[candidates[i]]=count #inserting the name of candidate and their votes**  
**candidates.remove(candidates[0])**  
**print("Election Results")**  
**print("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n")**  
**print("Total Votes: %d"%total) #printing the total votes**  
**print("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n")**  
**for i in candidates:**  
**percentage=(data[i]/total)\*100 #finding the percentage of each candidate**  
**percent\_list.append(percentage) #appending the precentage to percent\_list**  
**print("%s : %f(%d)"%(i,percentage,data[i]))**  
**print("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n")**  
**print("Winner:",candidates[percent\_list.index(max(percent\_list))]) #declaring the winner of the election**  
**print("\t\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n")**