|  |
| --- |
| Create a Python script that analyzes the PyBank records to calculate each of the following: |
|  |

|  |
| --- |
| # -->> The total number of months included in the dataset |
|  |

|  |
| --- |
| # -->> The net total amount of "Profit/Losses" over the entire period |
|  |

|  |
| --- |
| # -->> The average of the changes in "Profit/Losses" over the entire period |
|  |

|  |
| --- |
| # -->> The greatest increase in profits (date and amount) over the entire period |
|  |

|  |
| --- |
| # -->> The greatest decrease in losses (date and amount) over the entire period |
|  |

|  |
| --- |
| # -->> Print the analysis to the terminal and export a text file with the results |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| # Import dependencies |
|  |

|  |
| --- |
| import os |
|  |

|  |
| --- |
| import csv |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| # Define PyBank's variables |
|  |

|  |
| --- |
| months = [] |
|  |

|  |
| --- |
| profit\_loss\_changes = [] |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| count\_months = 0 |
|  |

|  |
| --- |
| net\_profit\_loss = 0 |
|  |

|  |
| --- |
| previous\_month\_profit\_loss = 0 |
|  |

|  |
| --- |
| current\_month\_profit\_loss = 0 |
|  |

|  |
| --- |
| profit\_loss\_change = 0 |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| # Change directory to the directory of current python script |
|  |

|  |
| --- |
| os.chdir(os.path.dirname(\_\_file\_\_)) |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| # Path to collect data from the Resources folder |
|  |

|  |
| --- |
| budget\_data\_csv\_path = os.path.join("Resources", "budget\_data.csv") |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| # Open and read csv |
|  |

|  |
| --- |
| with open(budget\_data\_csv\_path, newline="") as csvfile: |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| csv\_reader = csv.reader(csvfile, delimiter=",") |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| # Read the header row first |
|  |

|  |
| --- |
| csv\_header = next(csvfile) |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| #print(f"Header: {csv\_header}") |
|  |

|  |
| --- |
| # This prints -->> Header: Date, Profit/Losses |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| # Read through each row of data after the header |
|  |

|  |
| --- |
| for row in csv\_reader: |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| # Count of months |
|  |

|  |
| --- |
| count\_months += 1 |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| # Net total amount of "Profit/Losses" over the entire period |
|  |

|  |
| --- |
| current\_month\_profit\_loss = int(row[1]) |
|  |

|  |
| --- |
| net\_profit\_loss += current\_month\_profit\_loss |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| if (count\_months == 1): |
|  |

|  |
| --- |
| # Make the value of previous month to be equal to current month |
|  |

|  |
| --- |
| previous\_month\_profit\_loss = current\_month\_profit\_loss |
|  |

|  |
| --- |
| continue |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| else: |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| # Compute change in profit loss |
|  |

|  |
| --- |
| profit\_loss\_change = current\_month\_profit\_loss - previous\_month\_profit\_loss |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| # Append each month to the months[] |
|  |

|  |
| --- |
| months.append(row[0]) |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| # Append each profit\_loss\_change to the profit\_loss\_changes[] |
|  |

|  |
| --- |
| profit\_loss\_changes.append(profit\_loss\_change) |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| # Make the current\_month\_loss to be previous\_month\_profit\_loss for the next loop |
|  |

|  |
| --- |
| previous\_month\_profit\_loss = current\_month\_profit\_loss |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| #sum and average of the changes in "Profit/Losses" over the entire period |
|  |

|  |
| --- |
| sum\_profit\_loss = sum(profit\_loss\_changes) |
|  |

|  |
| --- |
| average\_profit\_loss = round(sum\_profit\_loss/(count\_months - 1), 2) |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| # highest and lowest changes in "Profit/Losses" over the entire period |
|  |

|  |
| --- |
| highest\_change = max(profit\_loss\_changes) |
|  |

|  |
| --- |
| lowest\_change = min(profit\_loss\_changes) |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| # Locate the index value of highest and lowest changes in "Profit/Losses" over the entire period |
|  |

|  |
| --- |
| highest\_month\_index = profit\_loss\_changes.index(highest\_change) |
|  |

|  |
| --- |
| lowest\_month\_index = profit\_loss\_changes.index(lowest\_change) |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| # Assign best and worst month |
|  |

|  |
| --- |
| best\_month = months[highest\_month\_index] |
|  |

|  |
| --- |
| worst\_month = months[lowest\_month\_index] |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| # -->> Print the analysis to the terminal |
|  |

|  |
| --- |
| print("Financial Analysis") |
|  |

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

|  |
| --- |
| print(f"Total Months: {count\_months}") |
|  |

|  |
| --- |
| print(f"Total: ${net\_profit\_loss}") |
|  |

|  |
| --- |
| print(f"Average Change: ${average\_profit\_loss}") |
|  |

|  |
| --- |
| print(f"Greatest Increase in Profits: {best\_month} (${highest\_change})") |
|  |

|  |
| --- |
| print(f"Greatest Decrease in Losses: {worst\_month} (${lowest\_change})") |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| # -->> Export a text file with the results |
|  |

|  |
| --- |
| budget\_file = os.path.join("Output", "budget\_data.txt") |
|  |

|  |
| --- |
| with open(budget\_file, "w") as outfile: |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| outfile.write("Financial Analysis\n") |
|  |

|  |
| --- |
| outfile.write("----------------------------\n") |
|  |

|  |
| --- |
| outfile.write(f"Total Months: {count\_months}\n") |
|  |

|  |
| --- |
| outfile.write(f"Total: ${net\_profit\_loss}\n") |
|  |

|  |
| --- |
| outfile.write(f"Average Change: ${average\_profit\_loss}\n") |
|  |

|  |
| --- |
| outfile.write(f"Greatest Increase in Profits: {best\_month} (${highest\_change})\n") |
|  |

outfile.write(f"Greatest Decrease in Losses: {worst\_month} (${lowest\_change})\n")