**Unit 3 | Assignment - Py Me Up, Charlie**

**Background**

Well... you've made it!

It's time to put away the Excel sheet and join the big leagues. Welcome to the world of programming with Python. In this homework assignment, you'll be using the concepts you've learned to complete **2** Python Challenges, PyBank and PyPoll. Both of these challenges encompasses a real-world situation where your newfound Python scripting skills can come in handy. These challenges are far from easy so expect some hard work ahead!

**Before You Begin**

1. Create a new GitHub repo called python-challenge. Then, clone it to your computer.
2. Inside your local git repository, create a directory for both of the Python Challenges. Use folder names corresponding to the challenges: **PyBank** and **PyPoll**.
3. Inside of each folder that you just created, add a new file called main.py. This will be the main script to run for each analysis.
4. Push the above changes to GitHub.

**PyBank**

* In this challenge, you are tasked with creating a Python script for analyzing the financial records of your company. You will give a set of financial data called [budget\_data.csv](https://github.com/UCF-Coding-Boot-Camp/UCFLM20190409DATA/blob/master/Homework/03-Python/Instructions/PyBank/Resources/budget_data.csv). The dataset is composed of two columns: Date and Profit/Losses. (Thankfully, your company has rather lax standards for accounting so the records are simple.)
* Your task is to create a Python script that analyzes the records to calculate each of the following:
  + The total number of months included in the dataset

Get: Date. Split out month. Count the months = Sum

* + The net total amount of "Profit/Losses" over the entire period

Get : Profit = Expenses/Revenue & 4 periods

row\_count = sum(1 for row in csvreader)

csvfile.seek(0)

next(csvreader)

* + The average of the changes in "Profit/Losses" over the entire period

Get: Calculate all the numbers/by how many numbers.

Sum=/Count

* + The greatest increase in profits (date and amount) over the entire period

Get: 1st = difference between 2 number comparing.

Increase = New Number – Original Number

Then, d/v the increase by the original number & \* 100

% increase = Increase/Original Number \*100

* + The greatest decrease in losses (date and amount) over the entire period

Get: 1st = difference between 2 number comparing.

Decrease = New Number – Original Number

Then, d/v the decrease by the original number & \* 100

% decrease = Decrease/Original Number \*100

* As an example, your analysis should look similar to the one below:
* Financial Analysis
* ----------------------------
* Total Months: 86
* Total: $38382578
* Average Change: $-2315.12
* Greatest Increase in Profits: Feb-2012 ($1926159)
* Greatest Decrease in Profits: Sep-2013 ($-2196167)
* In addition, your final script should both print the analysis to the terminal and export a text file with the results.

**PyPoll**

* In this challenge, you are tasked with helping a small, rural town modernize its vote-counting process. (Up until now, Uncle Cleetus had been trustfully tallying them one-by-one, but unfortunately, his concentration isn't what it used to be.)
* You will be give a set of poll data called [election\_data.csv](https://github.com/UCF-Coding-Boot-Camp/UCFLM20190409DATA/blob/master/Homework/03-Python/Instructions/PyPoll/Resources/election_data.csv). The dataset is composed of three columns: Voter ID, County, and Candidate. Your task is to create a Python script that analyzes the votes and calculates each of the following:
  + The total number of votes cast

Get: Count voters IDs = SUM

* + A complete list of candidates who received votes

Get: Candidate\_Name = SUM

* + The percentage of votes each candidate won

Get: Candidate\_Name

% of votes = Count Candidate\_Name/Total Votes

* + The total number of votes each candidate won

Get: Count each Candidate\_Name = SUM

+ (each) Candidate\_Name = SUM as number, no decimals

* + The winner of the election based on popular vote.

Get: Winner = Highest number of votes compared to all candidates

* As an example, your analysis should look similar to the one below:
* Election Results
* -------------------------
* Total Votes: 3521001
* -------------------------
* Khan: 63.000% (2218231)
* Correy: 20.000% (704200)
* Li: 14.000% (492940)
* O'Tooley: 3.000% (105630)
* -------------------------
* Winner: Khan
* -------------------------
* In addition, your final script should both print the analysis to the terminal and export a text file with the results.

**Hints and Considerations**

* Consider what we've learned so far. To date, we've learned how to import modules like csv; to read and write files in various formats; to store contents in variables, lists, and dictionaries; to iterate through basic data structures; and to debug along the way. Using what we've learned, try to break down you tasks into discrete mini-objectives. This will be a *much* better course of action than attempting to Google Search for a miracle.
* As you will discover, for some of these challenges, the datasets are quite large. This was done purposefully, as it showcases one of the limits of Excel-based analysis. While our first instinct, as data analysts, is often to head straight into Excel, creating scripts in Python can provide us with more robust options for handling "big data".
* Your scripts should work for each dataset provided. Run your script for each dataset separately to make sure that the code works for different data.
* Feel encouraged to work in groups, but don't shortchange yourself by copying someone else's work. You get what you put in, and the art of programming is extremely unforgiving to moochers. Dig your heels in, burn the night oil, and learn this while you can! These are skills that will pay dividends in your future career.
* Start early, and reach out for help often! Challenge yourself to identify *specific* questions for your instructors and TAs. Don't resign yourself to simply saying, "I'm totally lost." Come prepared to show your effort and thought patterns, we'll be happy to help along the way.
* Always commit your work and back it up with GitHub pushes. You don't want to lose hours of your work because you didn't push it to GitHub every half hour or so.
  + **Commit often**.

**Analysis Python Challenge**

**Teresina Toney**

**PyBank**

My plan was to read the files. This is the base of the code. Then, I set up the Financial Analysis Report. Next, I determined what information was trying to extract from the data file(s) to get the desired results of the report.

Get: Represents formula/code set development.

I used highest to lowest instead of revenue since I was looking for increases and decreases in profit.

Average Change was not working properly. My while loop was killed and was later determined that it was not needed. If you review the financial report, you have enough information to determine previous monthly activity which can be used to build a projection.

Assistance: James/TA on syntax errors; Tutor James Morris syntax errors

**PyPoll**

My plan was to read the files. This is the base of the code. Then, I set up the Election Results. Next, I determined what information was trying to extract from the data file(s) to get the desired results of the report.

Get: Represents formula/code set development.

**Troubleshooting my Code**

For both PyBank and PyPoll, I utilized “print” a lot. Each section of code was easier to determine the result after pulling information from the csv files. Most of my syntax errors were from typos and improper indentation.