**Python Challenge**

Python scripts used to help analyze budget/financial data (PyBank) and to help a small, rural town modernize its vote-counting process (PyPoll).

**Background**

For this project, I created two Python scripts.

* [PyBank](https://github.com/philipstubbs13/python-challenge#pybank)
* [PyPoll](https://github.com/philipstubbs13/python-challenge#pypoll)

**PyBank**

Inside the **PyBank** folder, you will find a script that is used to help analyze budget/financial data for a company. The data is in a csv file and includes two columns, **Date** and **Profit/Losses**.

The csv file being analyzed is located [here](https://github.com/philipstubbs13/python-challenge/blob/master/PyBank/Resources/budget_data.csv). The script file is located [here](https://github.com/philipstubbs13/python-challenge/blob/master/PyBank/main.py).

When you run the script, the script analyzes the profit/losses numbers and calculates 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.

When the script is finished, the financial analysis will be printed to the terminal as well as exported to a [text file](https://github.com/philipstubbs13/python-challenge/blob/master/PyBank/financial_results.txt) in the same directory as the script.

**PyPoll**

Inside the **PyPoll** folder, you will find a script that is used to help a small, rural town modernize its vote-counting process. The data is in a csv file and includes three columns, **Voter ID**, **County**, and **Candidate**.

The csv file being analyzed is located [here](https://github.com/philipstubbs13/python-challenge/blob/master/PyPoll/Resources/election_data.csv). The script file is located [here](https://github.com/philipstubbs13/python-challenge/blob/master/PyPoll/main.py).

When you run the script, the script analyzes the votes and calculates the following:

* The total number of votes cast.
* A complete list of candidates who received votes.
* The percentage of votes each candidate won.
* The total number of votes each candidate won.
* The winner of the election based on popular vote.

When the script is finished, the election results will be printed to the terminal as well as exported to a [text file](https://github.com/philipstubbs13/python-challenge/blob/master/PyPoll/election_results.txt) in the same directory as the script.

**Running the scripts**

To run either one of the scripts:

1. Download or clone this repository to a local directory on your computer.
2. From a command line terminal (for example, Git Bash on Windows), change directory into the root directory (**python-challenge**) and then change directory into the **PyBank** or **PyPoll** directory, depending on which script you want to run.
3. Run the script by running the following command:

python main.py

When finished, the script prints the results to the terminal and exports the results to a text file in output directory.

**Sample output**

After running a script, the results are printed to the terminal, similar to the following examples:

* PyBank

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)

* PyPoll

Election Results

-------------------------

Total Votes: 3521001

-------------------------

Khan: 63.000% (2218231)

Correy: 20.000% (704200)

Li: 14.000% (492940)

O'Tooley: 3.000% (105630)

-------------------------

Winner: Khan

-------------------------

**Technologies used**

The following technologies were used to build the scripts:

* Python
* Pandas