HomeWorl03 – PyBank & PyPoll

PyBank

Output File:

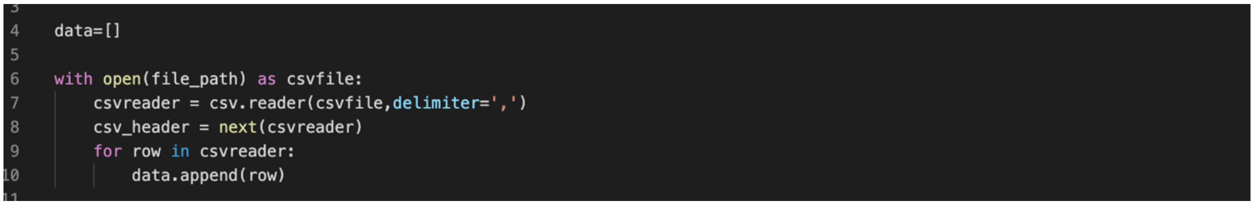
The output file will describe the overall Profit Analysis based on the CSV sent. Please find below a printout of the TXT file generated.

```
Financial Analysis
--------------------------------------------
Total Months: 86
Total: 38382578.0
Average Change: -2315.12
Greatest Increase in Profits: Feb-2012 ($1926159.0)
Greatest Decrease in Profits: Sep-2013 ($-2196167.0)
```

Code:

Although the code has comments, I have divided it into three main parts:

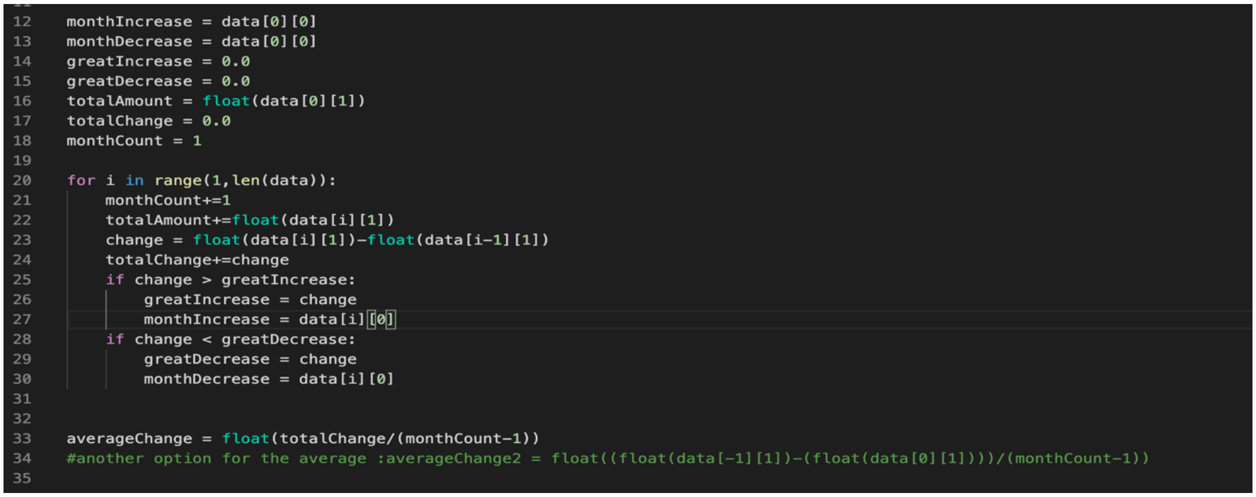
- Read the CSV File



```python
data=[]

with open(file_path) as csvfile:
    csvreader = csv.reader(csvfile,delimiter=',')
    csv_header = next(csvreader)
    for row in csvreader:
        data.append(row)
```

- Calculate and iterate across all data

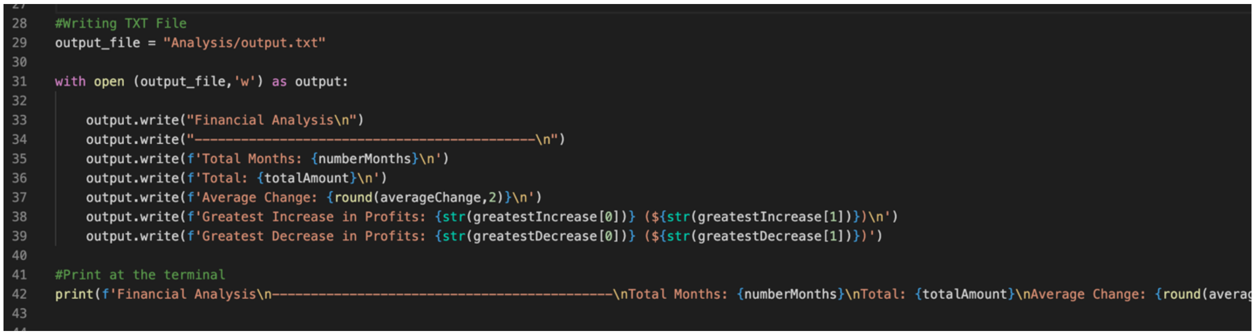


```python
monthIncrease = data[0][0]
monthDecrease = data[0][0]
greatIncrease = 0.0
greatDecrease = 0.0
totalAmount = float(data[0][1])
totalChange = 0.0
monthCount = 1

for i in range(1,len(data)):
    monthCount+=1
    totalAmount+=float(data[i][1])
    change = float(data[i][1])-float(data[i-1][1])
    totalChange+=change
    if change > greatIncrease:
        greatIncrease = change
        monthIncrease = data[i][0]
    if change < greatDecrease:
        greatDecrease = change
        monthDecrease = data[i][0]


averageChange = float(totalChange/(monthCount-1))
#another option for the average :averageChange2 = float((float(data[-1][1])-(float(data[0][1])))/(monthCount-1))
```

- Print and Create output file



```
#Writing TXT File
output_file = "Analysis/output.txt"

with open (output_file,'w') as output:

    output.write("Financial Analysis\n")
    output.write("--------------------------------------------\n")
    output.write(f'Total Months: {numberMonths}\n')
    output.write(f'Total: {totalAmount}\n')
    output.write(f'Average Change: {round(averageChange,2)}\n')
    output.write(f'Greatest Increase in Profits: {str(greatestIncrease[0])} (${str(greatestIncrease[1])})\n')
    output.write(f'Greatest Decrease in Profits: {str(greatestDecrease[0])} (${str(greatestDecrease[1])})')

#Print at the terminal
print(f'Financial Analysis\n--------------------------------------------\nTotal Months: {numberMonths}\nTotal: {totalAmount}\nAverage Change: {round(averag
```

PyPoll

Output File:

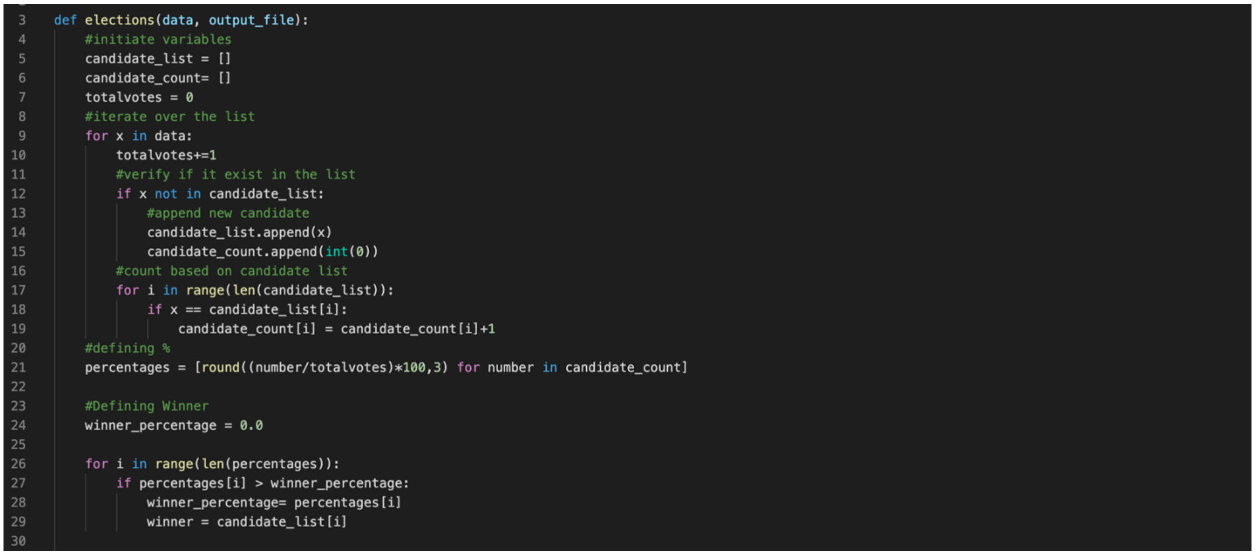
The output file describes the overall results of a Political campaign, highlighting the total number of votes, the % of each candidate, and the Winner of the election.

```
Elections Results
--------------------------------------------
Total Votes: 3521001
--------------------------------------------
Khan: 63.0% (2218231)
Correy: 20.0% (704200)
Li: 14.0% (492940)
O'Tooley: 3.0% (105630)
--------------------------------------------
Winner: Khan
--------------------------------------------
```

Code:

The strategy behind this code is different from the one used in the example above. Since the goal of this essay was not to use Pandas to summarize the data somehow and get the results, I have coded a function that runs all the main calculations we will need based on the "Candidate Name Field."

- Calculation & Output Function:
    - Iterate Part



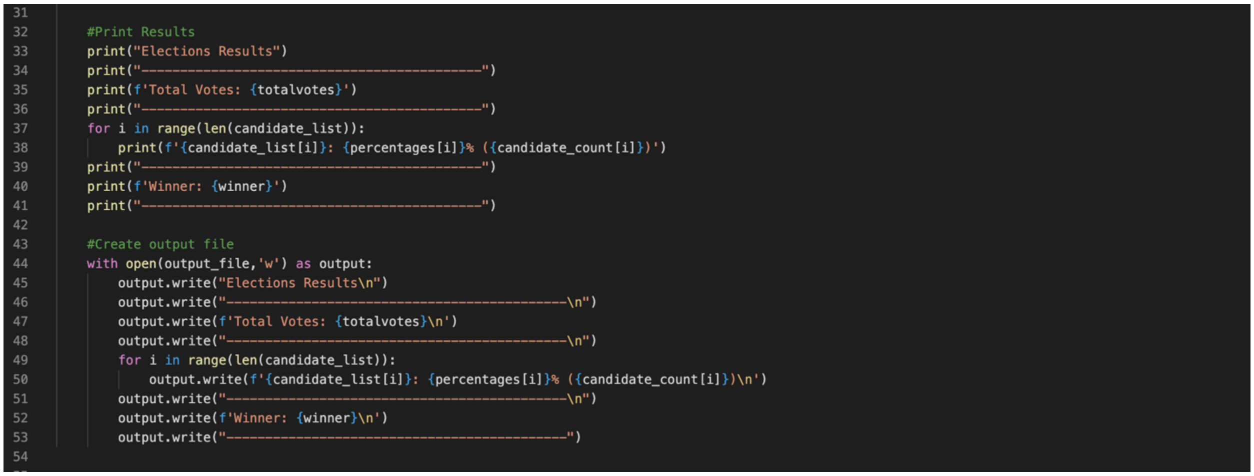
```python
def elections(data, output_file):
    #initiate variables
    candidate_list = []
    candidate_count= []
    totalvotes = 0
    #iterate over the list
    for x in data:
        totalvotes+=1
        #verify if it exist in the list
        if x not in candidate_list:
            #append new candidate
            candidate_list.append(x)
            candidate_count.append(int(0))
        #count based on candidate list
        for i in range(len(candidate_list)):
            if x == candidate_list[i]:
                candidate_count[i] = candidate_count[i]+1
    #defining %
    percentages = [round((number/totalvotes)*100,3) for number in candidate_count]

    #Defining Winner
    winner_percentage = 0.0

    for i in range(len(percentages)):
        if percentages[i] > winner_percentage:
            winner_percentage= percentages[i]
            winner = candidate_list[i]

```

- Output Part:



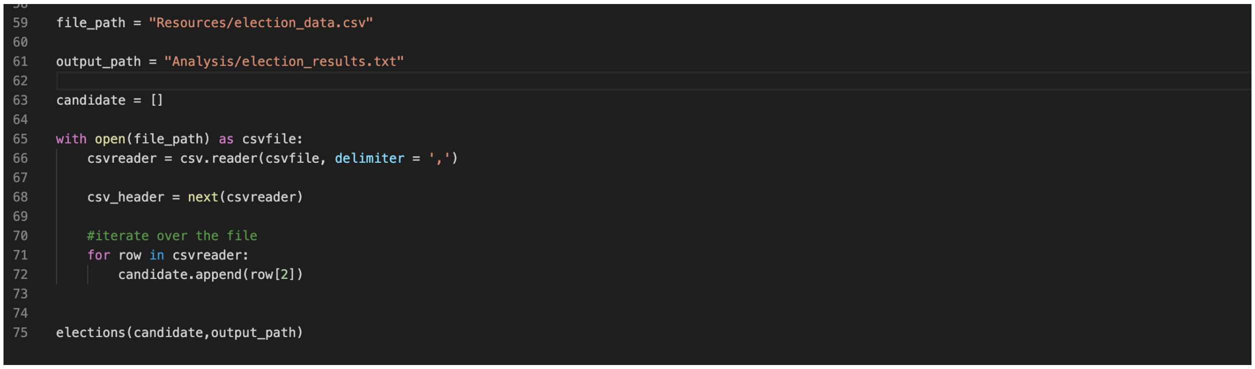
```python

    #Print Results
    print("Elections Results")
    print("--------------------------------------------")
    print(f'Total Votes: {totalvotes}')
    print("--------------------------------------------")
    for i in range(len(candidate_list)):
        print(f'{candidate_list[i]}: {percentages[i]}% ({candidate_count[i]})')
    print("--------------------------------------------")
    print(f'Winner: {winner}')
    print("--------------------------------------------")

    #Create output file
    with open(output_file,'w') as output:
        output.write("Elections Results\n")
        output.write("--------------------------------------------\n")
        output.write(f'Total Votes: {totalvotes}\n')
        output.write("--------------------------------------------\n")
        for i in range(len(candidate_list)):
            output.write(f'{candidate_list[i]}: {percentages[i]}% ({candidate_count[i]})\n')
        output.write("--------------------------------------------\n")
        output.write(f'Winner: {winner}\n')
        output.write("--------------------------------------------")

```

- Read the file and Function application:



```
file_path = "Resources/election_data.csv"

output_path = "Analysis/election_results.txt"

candidate = []

with open(file_path) as csvfile:
    csvreader = csv.reader(csvfile, delimiter = ',')

    csv_header = next(csvreader)

    #iterate over the file
    for row in csvreader:
        candidate.append(row[2])


elections(candidate,output_path)
```