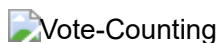


## Option 2: PyPoll



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 given two sets of poll data ( `election_data_1.csv` and `election_data_2.csv` ). Each 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
- 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.

As an example, your analysis should look similar to the one below:

```
Election Results
-----
Total Votes: 620100
-----
Rogers: 36.0% (223236)
Gomez: 54.0% (334854)
Brentwood: 4.0% (24804)
Higgins: 6.0% (37206)
-----
Winner: Gomez
-----
```

Your final script must be able to handle any such similarly-structured dataset in the future (i.e you have zero intentions of living in this hillbilly town -- so your script needs to work without massive re-writes). In addition, your final script should both print the analysis to the terminal and export a text file with the results.

```
In [1]: 1 import os
        2 import csv
```

```
In [2]: 1 # input and output files
        2 input_file = 'election_data_1-Copy1.csv'
        3 output_file = 'election_summary_1.txt'
```

```
In [3]: 1 # input and output paths
        2 csv_input_path = os.path.join('raw_data', input_file)
        3 txt_output_path = os.path.join('summary_doc', output_file)
```

```
In [4]: 1 candidates, total_candidates, candidate_perc, candidate_total, summaries = ([
2
3
4 with open(csv_input_path, mode='r', newline='') as poll_data:
5     reader = csv.reader(poll_data, delimiter=',')
6
7     next(reader)
8
9     num_rows = 0
10
11     for row in reader:
12         total_candidates.append(row[2])
13         num_rows += 1
14
```

```
In [5]: 1 # sorted list of total_candidates
2 sorted_candidates = sorted(total_candidates)
3
4 for i in range(num_rows):
5     if sorted_candidates[i - 1] != sorted_candidates[i]:
6         candidates.append(sorted_candidates[i])
```

```
In [6]: 1 # *-----*
2 # |  Output Summaries  |
3 # *-----*
4
5 print("\nElection Results")
6 print("-" * 40)
7 print("Total Votes:", num_rows)
8 print("-" * 40)
```

Election Results

```
-----
Total Votes: 803000
-----
```

```

In [7]: 1 for j in range(len(candidates)):
        2     candidate_count = 0
        3
        4     for k in range(len(sorted_candidates)):
        5         if candidates[j] == sorted_candidates[k]:
        6             candidate_count += 1
        7
        8     candidate_perc.append(round(candidate_count / num_rows * 100, 1))
        9     candidate_total.append(candidate_count)
       10
       11
       12 zipidy_doo_da = zip(candidates, candidate_perc, candidate_total)
       13
       14 for row in zipidy_doo_da:
       15     print(row[0] + ":", str(row[1]) + "%", "(" + str(row[2]) + ")")
       16     summary = (row[0] + ": ", str(row[1]) + "%", " (" + str(row[2]) + ")")
       17     summaries.append(summary)
       18
       19
       20 for k in range(len(candidate_perc)):
       21     if candidate_total[k] > candidate_total[k - 1]:
       22         chicken_dinner = candidates[k]
       23
       24
       25 print("-" * 40)
       26 print("Winner:", chicken_dinner)
       27 print("-" * 40)
       28 print("\n\n")

```

```

Cordin: 3.0% (24090)
Seth: 5.0% (40150)
Torres: 44.0% (353320)
Vestal: 48.0% (385440)

```

```

-----
Winner: Vestal
-----

```

```
In [8]: 1 with open(txt_output_path, mode='w', newline='') as posted_summaries:
2         writer = csv.writer(posted_summaries)
3
4         writer.writerows([
5             ["Election Results for: " + input_file],
6             ["-" * 40],
7             ["Total Votes: " + str(num_rows)],
8             ["-" * 40]
9         ])
10        writer.writerows(summaries)
11        writer.writerows([
12            ["-" * 40],
13            ["Winner: " + str(chicken_dinner)],
14            ["-" * 40]
15        ])
16
```

```

In [9]: 1 # Process second data set, election_data_2-Copy1.csv
2
3 # input and output files
4 input_file = 'election_data_2-Copy1.csv'
5 output_file = 'election_summary_2.txt'
6
7
8 # input and output paths
9 csv_input_path = os.path.join('raw_data', input_file)
10 txt_output_path = os.path.join('summary_doc', output_file)
11
12 candidates, total_candidates, candidate_perc, candidate_total, summaries = ([
13
14
15 with open(csv_input_path, mode='r', newline='') as poll_data:
16     reader = csv.reader(poll_data, delimiter=',')
17
18     next(reader)
19
20     num_rows = 0
21
22     for row in reader:
23         total_candidates.append(row[2])
24         num_rows += 1
25
26
27 # sorted list of total_candidates
28 sorted_candidates = sorted(total_candidates)
29
30 for i in range(num_rows):
31     if sorted_candidates[i - 1] != sorted_candidates[i]:
32         candidates.append(sorted_candidates[i])
33
34
35 # *-----*
36 # | Output Summaries |
37 # *-----*
38
39 print("\nElection Results")
40 print("-" * 40)
41 print("Total Votes:", num_rows)
42 print("-" * 40)
43
44
45 for j in range(len(candidates)):
46     candidate_count = 0
47
48     for k in range(len(sorted_candidates)):
49         if candidates[j] == sorted_candidates[k]:
50             candidate_count += 1
51
52     candidate_perc.append(round(candidate_count / num_rows * 100, 1))
53     candidate_total.append(candidate_count)
54
55
56 zipidy_doo_da = zip(candidates, candidate_perc, candidate_total)

```

```

57
58 for row in zipidy_doo_da:
59     print(row[0] + ":", str(row[1]) + "%", "(" + str(row[2]) + ")")
60     summary = (row[0] + ": ", str(row[1]) + "%", " (" + str(row[2]) + ")")
61     summaries.append(summary)
62
63
64 for k in range(len(candidate_perc)):
65     if candidate_total[k] > candidate_total[k - 1]:
66         chicken_dinner = candidates[k]
67
68
69 print("-" * 40)
70 print("Winner:", chicken_dinner)
71 print("-" * 40)
72 print("\n\n")
73
74
75 with open(txt_output_path, mode='w', newline='') as posted_summaries:
76     writer = csv.writer(posted_summaries)
77
78     writer.writerows([
79         ["Election Results for: " + input_file],
80         ["-" * 40],
81         ["Total Votes: " + str(num_rows)],
82         ["-" * 40]
83     ])
84     writer.writerows(summaries)
85     writer.writerows([
86         ["-" * 40],
87         ["Winner: " + str(chicken_dinner)],
88         ["-" * 40]
89     ])
90

```

## Election Results

-----  
Total Votes: 3521001  
-----

Correy: 20.0% (704200)  
Khan: 63.0% (2218231)  
Li: 14.0% (492940)  
O'Tooley: 3.0% (105630)  
-----

Winner: Khan  
-----