Option 2: PyPoll

 Vote-Counting

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]:
```
import os
import csv
```

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

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

In [4]:
```
candidates, total_candidates, candidate_perc, candidate_total, summaries = ([


with open(csv_input_path, mode='r', newline='') as poll_data:
    reader = csv.reader(poll_data, delimiter=',')

    next(reader)

    num_rows = 0

    for row in reader:
        total_candidates.append(row[2])
        num_rows += 1

```

In [5]:
```
# sorted list of total_candidates
sorted_candidates = sorted(total_candidates)

for i in range(num_rows):
    if sorted_candidates[i - 1] != sorted_candidates[i]:
        candidates.append(sorted_candidates[i])
```

In [6]:
```
# *--------------------*
# |  Output Summaries  |
# *--------------------*

print("\nElection Results")
print("-" * 40)
print("Total Votes:", num_rows)
print("-" * 40)
```

```
Election Results
----------------------------------------
Total Votes: 803000
----------------------------------------
```

In [7]:

```
for j in range(len(candidates)):
    candidate_count = 0

    for k in range(len(sorted_candidates)):
        if candidates[j] == sorted_candidates[k]:
            candidate_count += 1

    candidate_perc.append(round(candidate_count / num_rows * 100, 1))
    candidate_total.append(candidate_count)


zipidy_doo_da = zip(candidates, candidate_perc, candidate_total)

for row in zipidy_doo_da:
    print(row[0] + ":", str(row[1]) + "%", "(" + str(row[2]) + ")")
    summary = (row[0] + ": ", str(row[1]) + "%", " (" + str(row[2]) + ")")
    summaries.append(summary)


for k in range(len(candidate_perc)):
    if candidate_total[k] > candidate_total[k - 1]:
        chicken_dinner = candidates[k]


print("-" * 40)
print("Winner:", chicken_dinner)
print("-" * 40)
print("\n\n")
```

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

In [8]:

```
with open(txt_output_path, mode='w', newline='') as posted_summaries:
    writer = csv.writer(posted_summaries)

    writer.writerows([
        ["Election Results for: " + input_file],
        ["-" * 40],
        ["Total Votes: " + str(num_rows)],
        ["-" * 40]
    ])
    writer.writerows(summaries)
    writer.writerows([
        ["-" * 40],
        ["Winner: " + str(chicken_dinner)],
        ["-" * 40]
    ])

```

In [9]:

```
# Process second data set, election_data_2-Copy1.csv

# input and output files
input_file = 'election_data_2-Copy1.csv'
output_file = 'election_summary_2.txt'


# input and output paths
csv_input_path = os.path.join('raw_data', input_file)
txt_output_path = os.path.join('summary_doc', output_file)

candidates, total_candidates, candidate_perc, candidate_total, summaries = ([


with open(csv_input_path, mode='r', newline='') as poll_data:
    reader = csv.reader(poll_data, delimiter=',')

    next(reader)

    num_rows = 0

    for row in reader:
        total_candidates.append(row[2])
        num_rows += 1


# sorted list of total_candidates
sorted_candidates = sorted(total_candidates)

for i in range(num_rows):
    if sorted_candidates[i - 1] != sorted_candidates[i]:
        candidates.append(sorted_candidates[i])


# *--------------------*
# |  Output Summaries  |
# *--------------------*

print("\nElection Results")
print("-" * 40)
print("Total Votes:", num_rows)
print("-" * 40)


for j in range(len(candidates)):
    candidate_count = 0

    for k in range(len(sorted_candidates)):
        if candidates[j] == sorted_candidates[k]:
            candidate_count += 1

    candidate_perc.append(round(candidate_count / num_rows * 100, 1))
    candidate_total.append(candidate_count)


zipidy_doo_da = zip(candidates, candidate_perc, candidate_total)
```

```

for row in zipidy_doo_da:
    print(row[0] + ":", str(row[1]) + "%", "(" + str(row[2]) + ")")
    summary = (row[0] + ": ", str(row[1]) + "%", " (" + str(row[2]) + ")")
    summaries.append(summary)


for k in range(len(candidate_perc)):
    if candidate_total[k] > candidate_total[k - 1]:
        chicken_dinner = candidates[k]


print("-" * 40)
print("Winner:", chicken_dinner)
print("-" * 40)
print("\n\n")


with open(txt_output_path, mode='w', newline='') as posted_summaries:
    writer = csv.writer(posted_summaries)

    writer.writerows([
        ["Election Results for: " + input_file],
        ["-" * 40],
        ["Total Votes: " + str(num_rows)],
        ["-" * 40]
    ])
    writer.writerows(summaries)
    writer.writerows([
        ["-" * 40],
        ["Winner: " + str(chicken_dinner)],
        ["-" * 40]
    ])

```

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