**Challenge 3**

# Add our dependencies.

import csv

import os

# Assign a variable to load a file from a path.

file\_to\_load = os.path.join("Resources\election\_results.csv")

# Assign a variable to save the file to a path.

file\_to\_save = os.path.join("analysis","election\_analysis.txt")

# Initialize a total vote counter.

total\_votes = 0

# Candidate options and candidate votes.

candidate\_options = []

candidate\_votes = {}

# Track the winning candidate, vote count, and percentage.

winning\_candidate = ""

winning\_count = 0

winning\_percentage = 0

# Open the election results and read the file.

with open(file\_to\_load) as election\_data:

    file\_reader = csv.reader(election\_data)

    # Read the header row.

    headers = next(file\_reader)

    # Print each row in the CSV file.

    for row in file\_reader:

        # Add to the total vote count.

        total\_votes += 1

        # Get the candidate name from each row.

        candidate\_name = row[2]

        # If the candidate does not match any existing candidate, add the

        # the candidate list.

        if candidate\_name not in candidate\_options:

            # Add the candidate name to the candidate list.

            candidate\_options.append(candidate\_name)

            # And begin tracking that candidate's voter count.

            candidate\_votes[candidate\_name] = 0

        # Add a vote to that candidate's count.

        candidate\_votes[candidate\_name] += 1

# Save the results to our text file.

with open(file\_to\_save, "w") as txt\_file:

    # Print the final vote count to the terminal.

    election\_results = (

        f"\nElection Results\n"

        f"-------------------------\n"

        f"Total Votes: {total\_votes:,}\n"

        f"-------------------------\n")

    print(election\_results, end="")

    # Save the final vote count to the text file.

    txt\_file.write(election\_results)

    for candidate in candidate\_votes:

        # Retrieve vote count and percentage.

        votes = candidate\_votes[candidate]

        vote\_percentage = float(votes) / float(total\_votes) \* 100

        candidate\_results = (

            f"{candidate}: {vote\_percentage:.1f}% ({votes:,})\n")

        # Print each candidate's voter count and percentage to the terminal.

        print(candidate\_results)

        #  Save the candidate results to our text file.

        txt\_file.write(candidate\_results)

        # Determine winning vote count, winning percentage, and winning candidate.

        if (votes > winning\_count) and (vote\_percentage > winning\_percentage):

            winning\_count = votes

            winning\_candidate = candidate

            winning\_percentage = vote\_percentage

    # Print the winning candidate's results to the terminal.

    winning\_candidate\_summary = (

        f"-------------------------\n"

        f"Winner: {winning\_candidate}\n"

        f"Winning Vote Count: {winning\_count:,}\n"

        f"Winning Percentage: {winning\_percentage:.1f}%\n"

        f"-------------------------\n")

    print(winning\_candidate\_summary)

    # Save the winning candidate's results to the text file.

    txt\_file.write(winning\_candidate\_summary)