3.5.2

Get the Candidates in the Election

Great job getting the total votes in the election! Next, you will retrieve the names of the individual candidates in the election. This is important, because later you will need to determine how many votes each candidate received and their percentage of the total vote. So go grab some coffee (or your beverage of choice) and get ready to write some code with Tom.

Remember that when we inspected the data in `election_results.csv`, using the first 10 rows and the last 10 rows, there were at two candidate names: Charles Casper Stockham and Raymon Anthony Doane.

REWIND

Here's what we found when we looked at the first 10 and last 10 rows of `election_results.csv`:

First 10 Rows

	A	B	C
1	Ballot ID	County	Candidate
2	1323913	Jefferson	Charles Casper Stockham
3	1005842	Jefferson	Charles Casper Stockham
4	1880345	Jefferson	Charles Casper Stockham
5	1600337	Jefferson	Charles Casper Stockham
6	1835994	Jefferson	Charles Casper Stockham
7	1772756	Jefferson	Charles Casper Stockham
8	1920023	Jefferson	Charles Casper Stockham
9	1040408	Jefferson	Charles Casper Stockham
10	1018412	Jefferson	Charles Casper Stockham

Last 10 Rows

369703	4762851	Arapahoe	Raymon Anthony Doane
369704	4768093	Arapahoe	Raymon Anthony Doane
369705	4196905	Arapahoe	Raymon Anthony Doane
369706	4299985	Arapahoe	Raymon Anthony Doane
369707	4620283	Arapahoe	Raymon Anthony Doane
369708	4714953	Arapahoe	Raymon Anthony Doane
369709	4497542	Arapahoe	Raymon Anthony Doane
369710	4085849	Arapahoe	Raymon Anthony Doane
369711	4592018	Arapahoe	Raymon Anthony Doane
369712	4660518	Arapahoe	Raymon Anthony Doane

You may have scrolled through the CSV file using Excel or VS Code and found three candidate's names. However, this method isn't very efficient, as it takes a long time to scroll through 369,712 rows. With Python, we can iterate through the rows in the CSV file and get the candidates from the "Candidate" column, and then add their names to a list.

REWIND

Remember, when we iterated through the rows of `election_results.csv`, the last 10 rows that were printed to the terminal had the structure of Python list.

```
['4762851', 'Arapahoe', 'Raymon Anthony Doane']
['4678093', 'Arapahoe', 'Raymon Anthony Doane']
['4196905', 'Arapahoe', 'Raymon Anthony Doane']
['4299985', 'Arapahoe', 'Raymon Anthony Doane']
['4620283', 'Arapahoe', 'Raymon Anthony Doane']
['4714953', 'Arapahoe', 'Raymon Anthony Doane']
['4497542', 'Arapahoe', 'Raymon Anthony Doane']
['4085849', 'Arapahoe', 'Raymon Anthony Doane']
['4592018', 'Arapahoe', 'Raymon Anthony Doane']
['4660518', 'Arapahoe', 'Raymon Anthony Doane']
```

To get the candidate from each list when we iterate through the row, we can use indexing on the `for` loop variable, `row`. The Candidate column is the third column that has the second index, so we would use, `row[2]` to reference the Candidate column.

Let's test this to make sure. Follow these steps:

1. Declare a new list, `candidate_options = []` by adding it before the `with open()` statement in our script.
2. Add the following code to get the candidate's name from the row within the `for` loop.

```
# Print the candidate name from each row
        candidate_name = row[2]
```

3. Add the `candidate_name` to the `candidate_options` list using the `append()` method.

REWIND

To add an item to a list, use the `append()` method.

4. Add a print statement that is flush with the left margin to print out the `candidate_options` list.

Your file should look like this:

```
# 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
candidate_options = []

# 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

        # Print the candidate name from each row.
        candidate_name = row[2]

        # Add the candidate name to the candidate list.
        candidate_options.append(candidate_name)
```

```
# Print the candidate list.
print(candidate_options)
```

When we run this file, we will see all the elements, or candidates' names, from each row in the `candidate_options` list:

```
ny Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon A
nthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raym
on Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', '
Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane
', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony D
oane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Antho
ny Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon A
nthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raym
on Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', '
Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane
', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony D
oane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Antho
ny Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon A
nthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raymon Anthony Doane', 'Raym
```

However, we do not want every candidate from each row. Instead, we need to get only the unique candidate names.

To get the unique names in the `candidate_options` list, we can use an `if` statement with the `not in` membership operator to check if the candidate has been added to the list. If the candidate's name has been added to the list, then the next time the candidate's name is found in a row using the `for` loop, it will not be added to the list.

SHOW HINT

Inside the `for` loop, we will need to check if the candidate has been added to the `candidate_options` list. Therefore, add the following code to `PyPoll.py` inside the `for` loop.

```
# If the candidate does not match any existing candidate...
        if candidate_name not in candidate_options:
            # Add it to the list of candidates.
            candidate_options.append(candidate_name)
```

The code in the `for` loop should look like this:

```
# Print each row in the CSV file.
    for row in file_reader:
        # Add to the total vote count.
        total_votes += 1

        # Print the candidate name from each row.
        candidate_name = row[2]

        # If the candidate does not match any existing candidate...
        if candidate_name not in candidate_options:
            # Add it to the list of candidates.
            candidate_options.append(candidate_name)

# Print the candidate list.
print(candidate_options)
```

Run the file in the VS Code terminal.

FINDING

The output will be a list of the candidates in the election.

```
['Charles Casper Stockham', 'Diana DeGette', 'Raymon Anthony Doane']
```