**Election\_Analysis**

**Overview of Election Audit:**

The main objective behind performing election audit /election analysis for election commission is analysis and find out results of election like total county in election, total candidate participates in election, total votes, largest county based on votes, votes received by each candidate and winning candidate etc.

**Election-Audit Results:**

Following are the election audit outcome

* + Count total votes:

It is necessary to know total votes in election. Following is code for count total votes:

* + - Initialize total vote counter as 0.
    - Read each row in election data and increment total\_vote counter by 1

1. #Initilize total vote counter.
2. total\_votes=0
3. for row in file\_reader:
4. #Add to the total vote\_count
5. total\_votes+=1
   * Count the number of votes and the percentage of total votes for each county in the precinct:

* To retrieve votes per county, add following code in which first initialize two counter named as a county\_options as a list and county votes as a dictionary.
* Find all unique county by appending county\_option dictionary and increment county\_votes list for particular county to get total votes for particular county.

#Create a county list and county votes dictionary.

county\_options=[]

county\_votes={}

    # If county does not mathch with exsting county

        #add to the county list

        if county\_name not in county\_options:

            #Add the existing county to the list of counties.

            county\_options.append(county\_name)

            #Start the county votes counter

            county\_votes[county\_name]=0

            #Add county votes.

        county\_votes[county\_name]+=1

* Code in below image for count total votes in county and how many vote percentage county has from total votes in election.

for county\_name in county\_votes:

            #Retrieve the county vote count.

            votescounty= county\_votes[county\_name]

            #Calculate the percentage of votes for the county.

            county\_vote\_percentage=float(votescounty)/float(total\_votes)\*100

            #Print the county results to the terminal.

            countywise\_vote\_count=(f"{county\_name}: {county\_vote\_percentage:.1f}% ({votescounty:,})\n")

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

            print(countywise\_vote\_count)

            # Save the county votes to a text file.

            election\_analysis.write(countywise\_vote\_count)

* + county had the largest number of votes: -
* To calculate which county had a largest number of votes first initialize variables .
* largest\_county to store county name with highest votes , largrst\_county\_count counter to store number of votes in largest county and third largest\_county\_percentag

to store percentage of votes largest county had.

* Now compare number of votes and percentage for particular county and find the county with highest number of votes and percentage.

#Track the largest county and county voter turnout.

largest\_county=""

largest\_county\_count=0

largest\_county\_percentage=0

if (votescounty>largest\_county\_count)and (county\_vote\_percentage>largest\_county\_percentage):

                largest\_county\_count=votescounty

                largest\_county\_percentage=county\_vote\_percentage

                largest\_county=county\_name

    largest\_county\_summary=(

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

            f"Largest county turnout:{largest\_county}\n"

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

    #Print the county with the largest turnout to the terminal.

    print(largest\_county\_summary)

    #Save the county with the largest turnout to a text file.

    election\_analysis.write(largest\_county\_summary)

* + Count number of votes and the percentage of the total votes each candidate received: -
* To find out votes and percentage of the votes each candidate received first declare candidate\_option list to find all candidate participate in election and candidate\_votes dictionary to calculate number of votes received by particular candidate from total votes in election.

# declare list for candidate options and dictionary for candidate.

candidate\_options=[]

candidate\_votes={}

* After initialize variable examine each row in election data to find out how many candidates participate in election. Append candidate\_option list when we found new candidate.
* Increment candidate\_votes dictionary by 1 to find total number of votes received by particular candidate from total votes in election.

# for each row in CSV file

    for row in file\_reader:

# get the candidate name from each row

        candidate\_name=row[2]

if candidate\_name not in candidate\_options:

            #Add candidate name in candidate list

            candidate\_options.append(candidate\_name)

            #Start the candidate votes counter

            candidate\_votes[candidate\_name]=0

            #Add candidate votes.

        candidate\_votes[candidate\_name]+=1

* Now use for loop for each candidate to calculate total votes he/she received and percentage of votes from total votes in election .

for candidate\_name in candidate\_votes:

        #Retrive vote count and percentage

            votes= candidate\_votes[candidate\_name]

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

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

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

            print(election\_results)

            #  Write each candidate's voter count and percentage to the text file.

            election\_analysis.write(candidate\_result)

* + Which candidate won the election, what was their vote count, and what was their percentage of the total votes?
* Now our last analysis and coding for winning candidate in election. For this first declare and initialize 3 variables. Winning candidate variable initialize with empty string to store winning candidate name , winning\_count initialize with value 0 to store votes received by winning candidate and winnig\_percentage also now initialize with 0 to store percentage of votes received from total votes by winning candidate.

#track winning candidate,vote\_count and vote percentage

winning\_candidate=""

winning\_count=0

winning\_percentage=0

* Now compare votes and votes percentage for all three candidate and store candidate who received height votes in winning\_candidate varible , store his/her votes in winning\_count varible and finally store percentage of votes in winning\_percentage.

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

                winning\_count=votes

                winning\_percentage=vote\_percentage

                winning\_candidate=candidate\_name

    # Print the winning candidate (to terminal)

    winning\_candidate\_summary=(

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

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

            f"winning vote count:{winning\_count:,}\n"

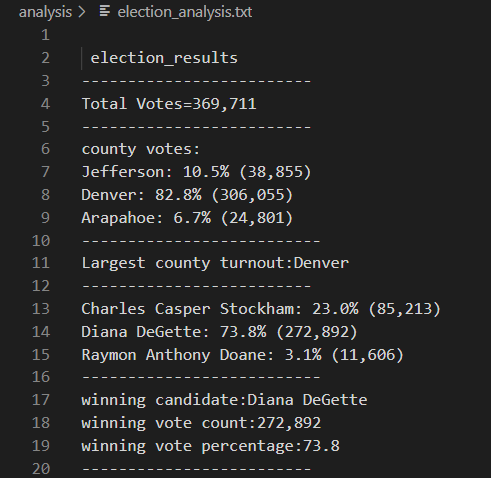
            f"winning vote percentage:{winning\_percentage:.1f}"

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

    #Save candidate result to text file.

    election\_analysis.write(winning\_candidate\_summary)

* Below image shows the election results saved in .txt file.



**Election-Audit Summary:**

We can perform this type of election analysis for other election or for other election commission. In future we can use this script with following modification.

* We can modify script to find how many votes each candidate received from each county.
* If we have more data than also we can find how many MALE and FEMALE voters in election. Also we can extend script to find out how many MALE and FEMALE give vote to particular candidate.