final project check point

Project code

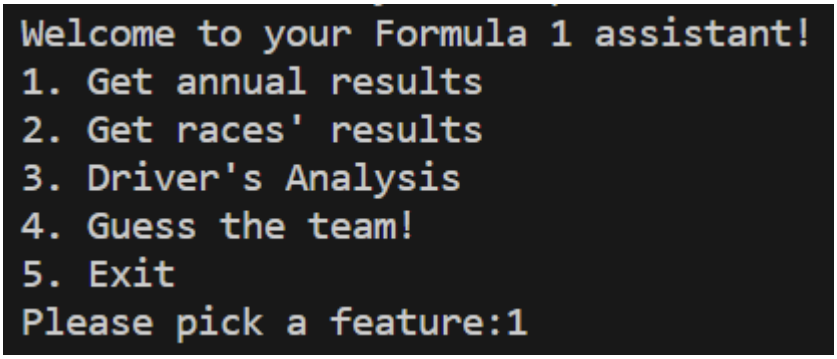
- Link: https://github.com/pontiff701/F1-data-assistant
- Packages: `requests`, `BeautifulSoup`, `time`, `json`, `random`, `numpy`

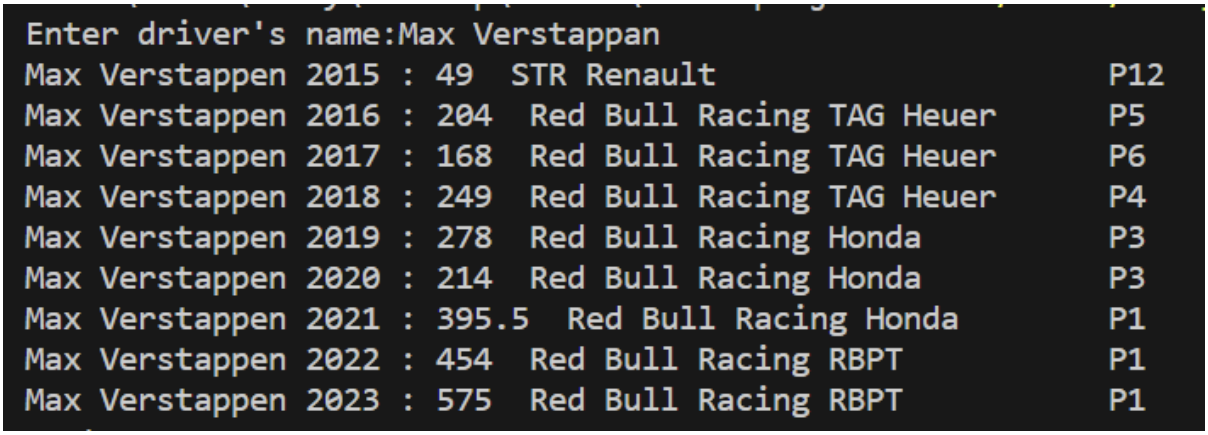
Data Sources

- All data are come from the Formula 1 official website: https://www.formula1.com/en.html
    - All race name are come from: https://www.formula1.com/en/results.html/[input_your_year_here]/races.html
    - All race results are come from: https://www.formula1.com/en/results.html/[input_your_year_here]/races/[input_the_race_name_here]/race-result.html
    - All annual driver information are come from: https://www.formula1.com/en/results.html/[input_your_year_here]/drivers.html
    - All annual team information are come from: https://www.formula1.com/en/results.html/[input_your_year_here]/team.html
- Each website we called in the program will be stored as a json cache file: `cache_drivers.json`, `cache_results.json`, and `cache_teams.json`. The driver's information will be stored in a LiteSQL file `sqlData.txt` from `driverInfo.py`.
- All results will be stored as a txt file.
- The project scripts access Formula 1 data primarily through web scraping techniques, using Python libraries such as `requests` and `BeautifulSoup`.
    - **requests**: Utilized for making HTTP requests to the official Formula 1 website, fetching web pages that contain data about drivers, teams, and race results.
    - **BeautifulSoup**: Employed to parse and extract the required data from the retrieved HTML content. It facilitates efficient navigation of the HTML structure, enabling specific data extraction.
- The data encompasses driver and team statistics, race schedules, results, and standings. Each script targets specific data types:
    - `driverInfo.py` - Focuses on individual driver statistics, using cahce file `cache_drivers.json`.
    - `team.py` - Extracts team-related data, using cahce file "cache_team.json".
    - `raceInfo.py` - Gathers detailed information on races, using cahce file "cache_results.json".
    - `personalAnalysis.py` - Analyze each driver's performance, using cache file from `sqlData.txt`
- Till the end of 2023, I have the data of all drivers and races from 1950 to 2023 seasons, and the data of all teams from 1958 to 2023 sesons.
    - The driver data include their name, team, and total points got in each season.
    - The team results includes the total points of each team got in each season or each peorid we selected.

    - The race results includes all driver's standings, lap time, and the points they obtained in each race.

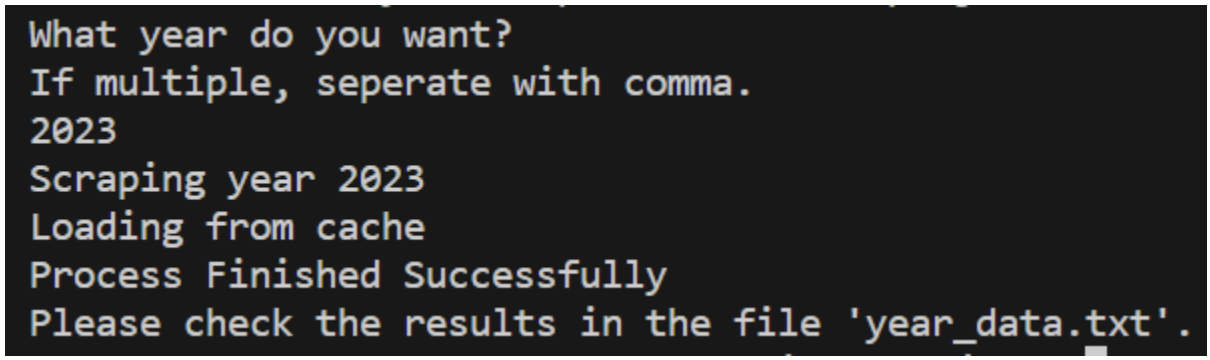
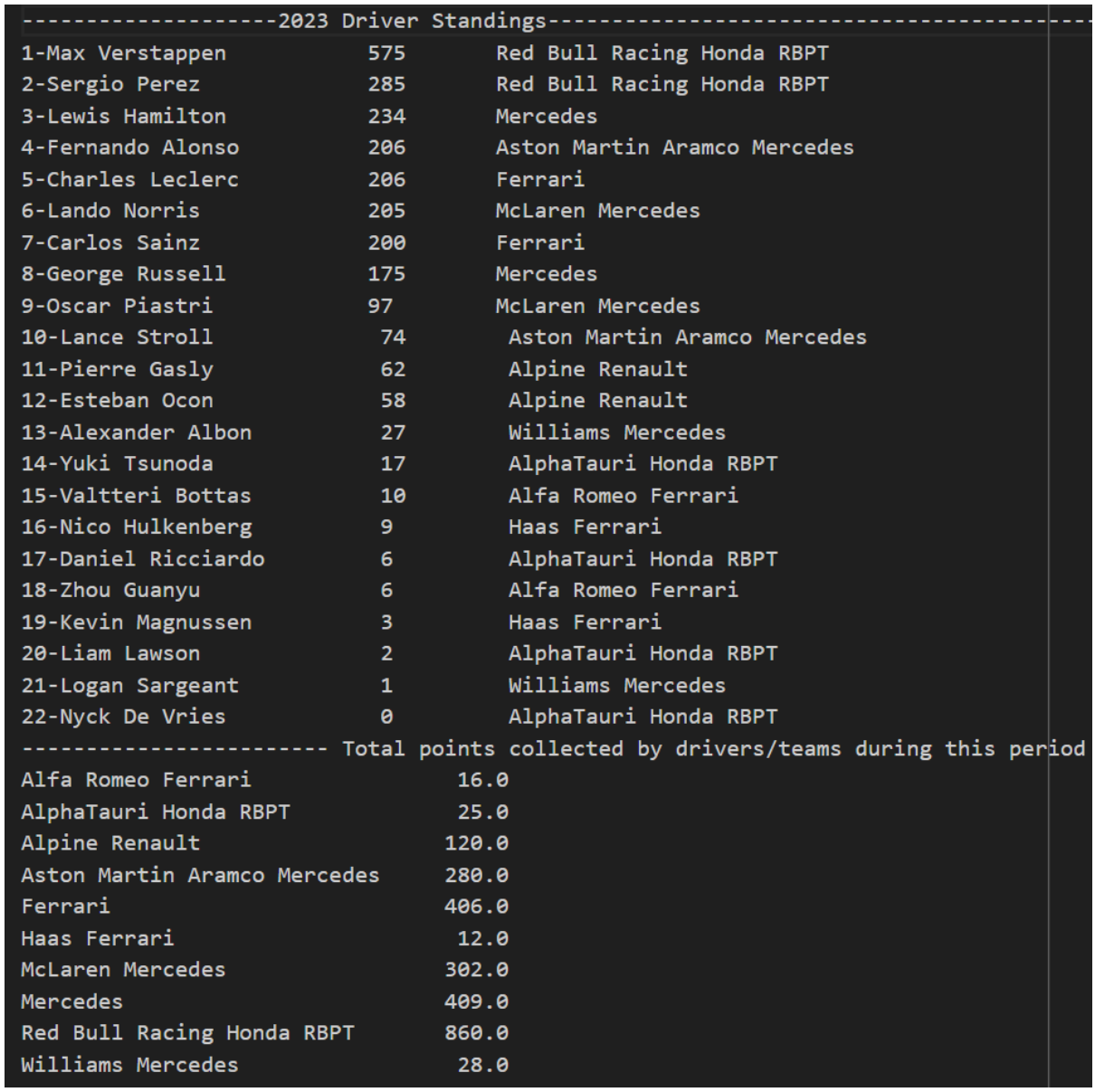
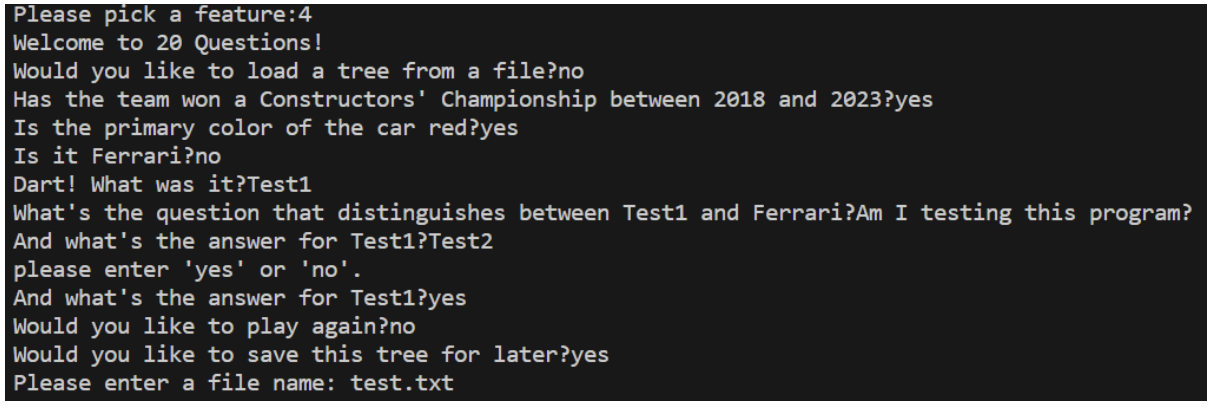
Data Structure

- In this program, the data is organized into structured formats suitable for analysis and read. Python data structures such as dictionaries and lists are used to store and manipulate the data. The project structures data in a way that facilitates easy access and analysis of specific aspects like driver performance and team standings.
- All race results & team performace will be summarized and output into a txt file. All driver's performance data will be analyzed and stored into a txt file.
- some screenshots of the current features:
    - 

    - 


```
What year do you want?
If multiple, seperate with comma.
2023
Scraping year 2023
Loading from cache
These are the races of 2023 select the race you want by typing number of it.
0 -
1 - bahrain
2 - saudi-arabia
3 - australia
4 - azerbaijan
5 - miami
6 - italy
7 - monaco
8 - spain
9 - canada
10 - austria
11 - great-britain
12 - hungary
13 - belgium
14 - netherlands
15 - italy
16 - singapore
17 - japan
18 - qatar
19 - united-states
20 - mexico
21 - brazil
22 - las-vegas
23 - abu-dhabi
Enter: 0
1141/bahrain
```

```
-------------------- 2023 1141/bahrain Race Info ----------------------------------
Pos  Name              Team                          Laps Time          Points
1    Max Verstappen    Red Bull Racing Honda RBPT      57 1:33:56.736       25
2    Sergio Perez      Red Bull Racing Honda RBPT      57 +11.987s          18
3    Fernando Alonso   Aston Martin Aramco Mercedes    57 +38.637s          15
4    Carlos Sainz      Ferrari                         57 +48.052s          12
5    Lewis Hamilton    Mercedes                        57 +50.977s          10
6    Lance Stroll      Aston Martin Aramco Mercedes    57 +54.502s           8
7    George Russell    Mercedes                        57 +55.873s           6
8    Valtteri Bottas   Alfa Romeo Ferrari              57 +72.647s           4
9    Pierre Gasly      Alpine Renault                  57 +73.753s           2
10   Alexander Albon   Williams Mercedes               57 +89.774s           1
11   Yuki Tsunoda      AlphaTauri Honda RBPT           57 +90.870s           0
12   Logan Sargeant    Williams Mercedes               56 +1 lap             0
13   Kevin Magnussen   Haas Ferrari                    56 +1 lap             0
14   Nyck De Vries     AlphaTauri Honda RBPT           56 +1 lap             0
15   Nico Hulkenberg   Haas Ferrari                    56 +1 lap             0
16   Zhou Guanyu       Alfa Romeo Ferrari              56 +1 lap             0
17   Lando Norris      McLaren Mercedes                55 +2 laps            0
18   Esteban Ocon      Alpine Renault                  41 DNF                0
19   Charles Leclerc   Ferrari                         39 DNF                0
20   Oscar Piastri     McLaren Mercedes                13 DNF                0
```

- 
```
What year do you want?
If multiple, seperate with comma.
2023
Scraping year 2023
Loading from cache
Process Finished Successfully
Please check the results in the file 'year_data.txt'.
```
- 
```
--------------------2023 Driver Standings--------------------------------------------
1-Max Verstappen           575       Red Bull Racing Honda RBPT
2-Sergio Perez             285       Red Bull Racing Honda RBPT
3-Lewis Hamilton           234       Mercedes
4-Fernando Alonso          206       Aston Martin Aramco Mercedes
5-Charles Leclerc          206       Ferrari
6-Lando Norris             205       McLaren Mercedes
7-Carlos Sainz             200       Ferrari
8-George Russell           175       Mercedes
9-Oscar Piastri            97        McLaren Mercedes
10-Lance Stroll             74        Aston Martin Aramco Mercedes
11-Pierre Gasly             62        Alpine Renault
12-Esteban Ocon             58        Alpine Renault
13-Alexander Albon          27        Williams Mercedes
14-Yuki Tsunoda             17        AlphaTauri Honda RBPT
15-Valtteri Bottas          10        Alfa Romeo Ferrari
16-Nico Hulkenberg          9         Haas Ferrari
17-Daniel Ricciardo         6         AlphaTauri Honda RBPT
18-Zhou Guanyu              6         Alfa Romeo Ferrari
19-Kevin Magnussen          3         Haas Ferrari
20-Liam Lawson              2         AlphaTauri Honda RBPT
21-Logan Sargeant           1         Williams Mercedes
22-Nyck De Vries            0         AlphaTauri Honda RBPT
----------------------- Total points collected by drivers/teams during this period
Alfa Romeo Ferrari               16.0
AlphaTauri Honda RBPT            25.0
Alpine Renault                  120.0
Aston Martin Aramco Mercedes    280.0
Ferrari                         406.0
Haas Ferrari                     12.0
McLaren Mercedes                302.0
Mercedes                        409.0
Red Bull Racing Honda RBPT      860.0
Williams Mercedes                28.0
```
- 
```
Please pick a feature:4
Welcome to 20 Questions!
Would you like to load a tree from a file?no
Has the team won a Constructors' Championship between 2018 and 2023?yes
Is the primary color of the car red?yes
Is it Ferrari?no
Dart! What was it?Test1
What's the question that distinguishes between Test1 and Ferrari?Am I testing this program?
And what's the answer for Test1?Test2
please enter 'yes' or 'no'.
And what's the answer for Test1?yes
Would you like to play again?no
Would you like to save this tree for later?yes
Please enter a file name: test.txt
```

Interaction and Presentation Plans

- The user will be asked to select the features at the first, including get the driver's data, get the race's result, get the team's data, and play the 20 questions game.
    - For the firt three functions, the users will be asked to enter the year, select the race, or enter the driver's name to get the information they need.
    - For the game, the users will be asked if they want to load a existed tree. Then, during the game, they will only need to enter yes or no for the questions. If the program don't know the answer, user will be asked to enter the correct answer and question to match this answer. The game can also study the user's answer and save it into a txt file for the next play.
- The program will mainly using command line prompts for the interactive options and presentation.

Demo Video

- Link: https://youtu.be/_bLmWmwvFMU