1. Project Overview

I used the cars api from api ninjas that outputs a car based on the parameter that is given in the api call. For example "camry" outputs a json array of a description of car like how many cylinders, engine size, car type. It's a game that quizzes a player by displaying information about a car like its make and other important details and then makes them guess the model. This project solves growing problem of new cars being entered in the world and the need to identify them for example for police or just to help users know more about cars and their specs.

2. Code Breakdown

For each file, explain its role and describe the important methods and logic inside, which includes classes. Use bullet points or a paragraph for each.

`Main.java`

Initializes CarGuessingGame object
Runs the game 3 times
Tracks and displays average cylinders
Reads from saved car data file at the beginning

`API.java`

the method is called with a car model chosen which is added into the endpoint for the api connection.
a get command is used to get data about that specific model from the API which is then returned as a JSONArray to CarGuessingGame.java

`CarGuessingGame.java`

Instance array **models** with a list of 20 possible models for the user to be quizzed on

startGame()
Filters and Displays important information about the car model to the player
Checks if the car model's information is able to be presented by the API
Reads in the JSON array output and filters the important data points.
Reads the user's guess and outputs if the user is correct.
Saves the car model and number of cylinders using the saveCarData method
Returns the number of cylinders to App.java

getRandomModel()
Randomly chooses a model from the instance array **models**

saveCarData()
Takes the car model and number of cylinders from startGame() and writes it in car_history.txt

loadCarData()
Reads line by line the past car history of cars that were played in the Car Guessing Game in car_history.txt

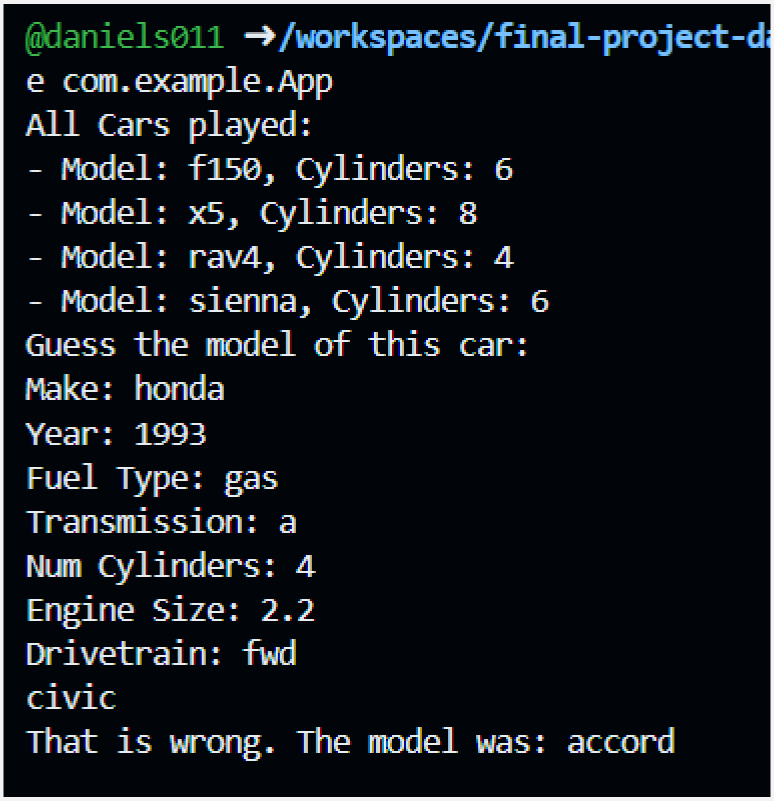
3. Features Implemented

• What optional features did you complete?

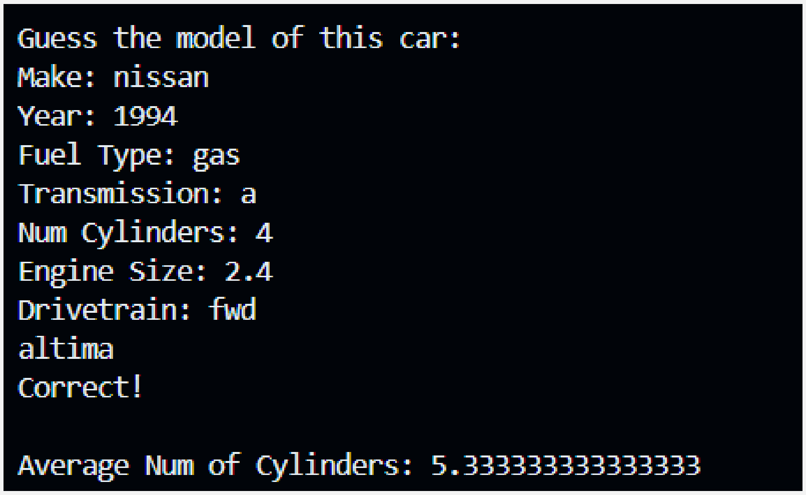
Cars API implemented +88%
Calls the API with a specific model to get information about it.
Basic Statistics on API output +6%
Average number of cylinders is calculated for all cars in a given play instance
Read/Write Past Car Statistics + 2%
Car model and number of cylinders is saved in a file and then shown in the beginning of the game
Filter API Data for Most Important Car Details +2%
The make, year, fuel type, transmission, number of cylinders, engine size,
and drivetrain type are filtered out to get the most relevant details for the player

Total: 98%

4. Screenshots or Outputs



```
@daniels011 ➜/workspaces/final-project-da
e com.example.App
All Cars played:
- Model: f150, Cylinders: 6
- Model: x5, Cylinders: 8
- Model: rav4, Cylinders: 4
- Model: sienna, Cylinders: 6
Guess the model of this car:
Make: honda
Year: 1993
Fuel Type: gas
Transmission: a
Num Cylinders: 4
Engine Size: 2.2
Drivetrain: fwd
civic
That is wrong. The model was: accord
```



```
Guess the model of this car:
Make: nissan
Year: 1994
Fuel Type: gas
Transmission: a
Num Cylinders: 4
Engine Size: 2.4
Drivetrain: fwd
altima
Correct!

Average Num of Cylinders: 5.333333333333333
```

5. What I Learned

- I learned how to read a json output from the cars api.
- I learned to use a REST API and implement it into a functioning game that tests players on cars.
- I learned out to read and write to and from a text file in java.
- I learned how to parse a json array to get the most important information about a car.