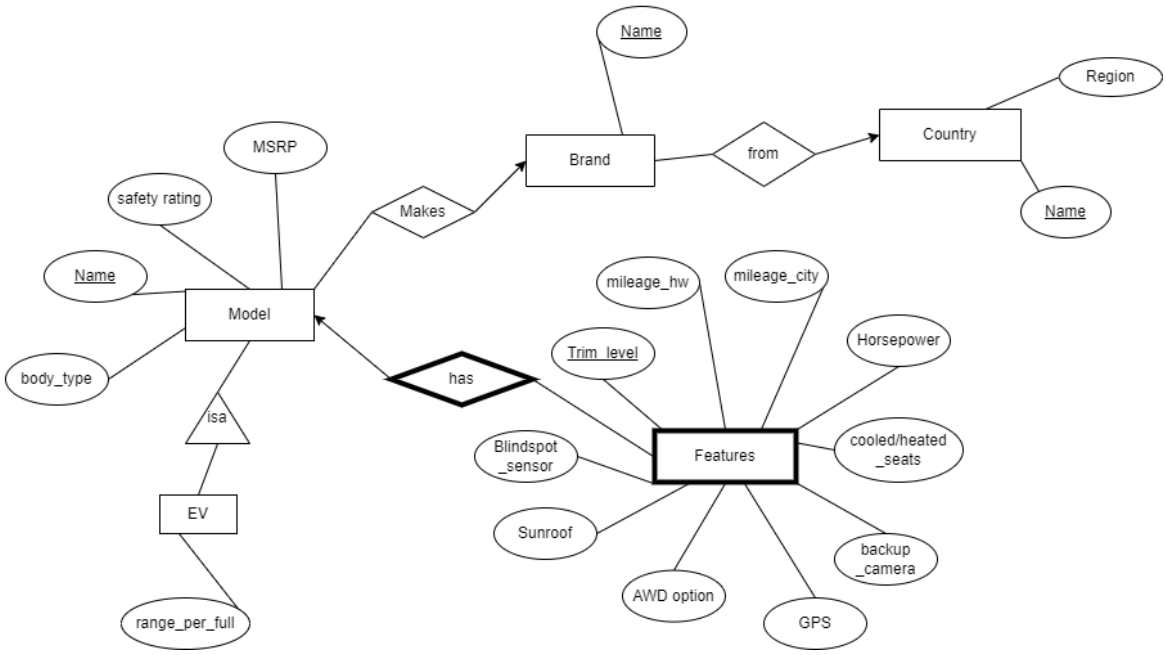
Group n: Nhat Nguyen

EECS 647: Introduction to Database Systems

Course Project Final Report

As the first report for this project, I decided to make a simple library of vehicles with detail information such as MSRP, gas milage, safety rating and more. This project could be a good resource for users that want to look up a detail information of a vehicle or help them find a vehicle with desire specs and features. However, I have modified the ER diagram since the first report that the vehicles will be 2022 model only. This saved me a ton of time to focus on other requirements. Below is the updated ER diagram for my 2022 vehicles library:

2022 Vehicles Library ER Diagram



To implement this to database, I had written a .sql file and the code to create all entity sets and their data. Then I import the .sql file to phpMyAdmin through virtualdesktop. However, I only filled out half of the data for this. I realized I spent too much time to make research and enter the correct data for each attribute. I decided to fill-out the rest of the data with "no"s and create a program to update "no" to random data. Specifically, my random generator program updates the horsepower with random range from 80 to 800, weight is from 2000 to 5000, mileage

is from 14 to 60. For the features, 50% of vehicles has blind spot censor, 75% has back-up camera, 33% has all-wheel-drive option, 50% has GPS, 67% has temperature seats, and 25% has sunroof. I believe correct data is optional for this project. On the other hand, with my random data generator, then I have a reasonably good database to work on.

For the front-end application of the project, I used html, javascript to make an interactive website, css to style, and php to get requests and retrieve the data from the database through mysqli. I have not used any code generator since I just learn to do this all those languages in this semester and I decided not to depend on any cool codes generator yet.

My 2022 Vehicles Library website has 2 main tabs that searching information of a vehicle and searching a vehicle by features. Searching information of a vehicles will display all the information of a vehicle based on brand, model, and trims level that users select. Moreover, it can also search all brand, all model of a selected brand or all trims of a selected brand and model depend on input of the user (If you are going to test this, I recommend selecting Toyota, Honda, Lexus, Acura, Tesla as selecting brands because each of their vehicle has more than one trim level to select, other brands only have one base trim level. I wished I could hire someone to fill-out the database for me back then.). If user select a specific brand and model, it will have a nicer display for the data than just a list. Searching a vehicle by features will display a list of vehicles based on the input of users that body type, safety rating, price range, weight range, minimum horsepower (I do not think of anyone wants to limit the power of a vehicle when look for one.), minimum gas mileages, and available features. This could be potential useful for users who want to look for their dream car or compare between vehicles across the brand and not depend on dealerships. Besides two main tabs, I also implemented two side tabs that EV list and Top 5's in the library. EVlist is the list of every electric vehicle and its specs and features in the library. Top 5's tab has four ranking lists that 5 most expensive vehicles based on their MSRP, 5 best gas saving truck based on their average gas mileage, 5 most powerful sedan based on their ratio of power-to-weight, and 5 countries has the most vehicles in the library. All the lists are based on the data in the database so if, of course, the database got updated, the lists could be updated also.

Here is the link to my project: https://people.eecs.ku.edu/~m926n810/647CourseProject/

Here is the link to random data generator if in case you want to update my library with another random datasets:
https://people.eecs.ku.edu/~m926n810/647CourseProject/randomdatageneratorphpfile/RandomizeSomeData.html