The database for the Car Market application is designed to manage user accounts, vehicle data, saved cars, and user preferences. It provides the backend structure for a platform where users can browse vehicles, save their favorites, and express their preferences by liking or disliking specific cars. The database is structured to ensure data integrity, scalability, and ease of use, leveraging relationships between entities to handle complex interactions.

The **User** table stores essential information about users, such as their username, password, display name, and review level. The username serves as the primary key, uniquely identifying each user. Users can save cars they are interested in or indicate whether they like or dislike a particular vehicle.

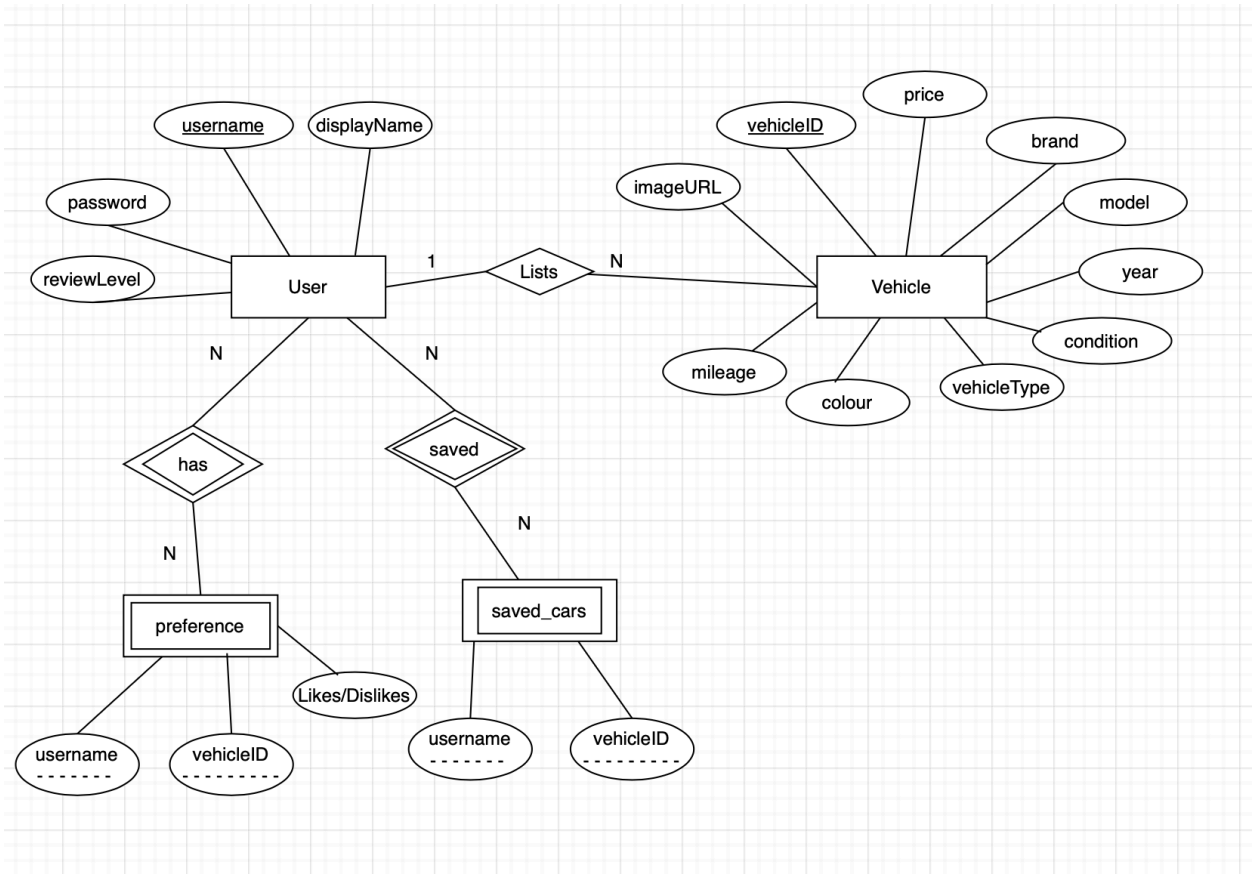
The **Vehicle** table contains information about the cars available on the platform. Each vehicle has attributes like a unique vehicle ID, brand, model, price, year, condition, image URL, mileage, color, and type. The vehicleID is the primary key, ensuring that each car in the database is uniquely identifiable.

To manage the many-to-many relationships between users and vehicles, the database includes two junction tables: **Saved Cars** and **Preference**. The **Saved Cars** table links users and vehicles to track which cars a user has saved. It uses a composite primary key of username and vehicleID, ensuring that each user-vehicle pair is unique. This setup allows multiple users to save the same car and a single user to save multiple cars.

The **Preference** table also connects users and vehicles but adds an additional attribute, likesOrDislikes, which records whether a user likes or dislikes a specific vehicle. Like the **Saved Cars** table, it uses a composite primary key of username and vehicleID to enforce unique user-vehicle relationships. This table allows the application to capture user feedback and preferences, providing valuable insights for personalization or analytics.

The relationships in the database are well-defined. A user can save multiple cars, and a car can be saved by multiple users, reflecting a many-to-many relationship. Similarly, users can like or dislike multiple cars, and cars can receive feedback from multiple users, which is also a many-to-many relationship. The database uses foreign keys and composite keys to maintain these relationships and ensure data consistency.

Overall, the database supports key features of the Car Market application, including user account management, vehicle browsing, saving vehicles, and liking or disliking vehicles. The design is highly normalized, minimizing data redundancy while allowing the system to scale as the user base and inventory grow. This structure provides a robust and efficient platform for managing user interactions with vehicles.


username
displayName
password
reviewLevel
User
1
Lists
N
vehicleID
price
brand
imageURL
model
year
condition
mileage
colour
vehicleType
Vehicle
N
N
has
saved
N
N
preference
saved_cars
Likes/Dislikes
username
vehicleID
username
vehicleID