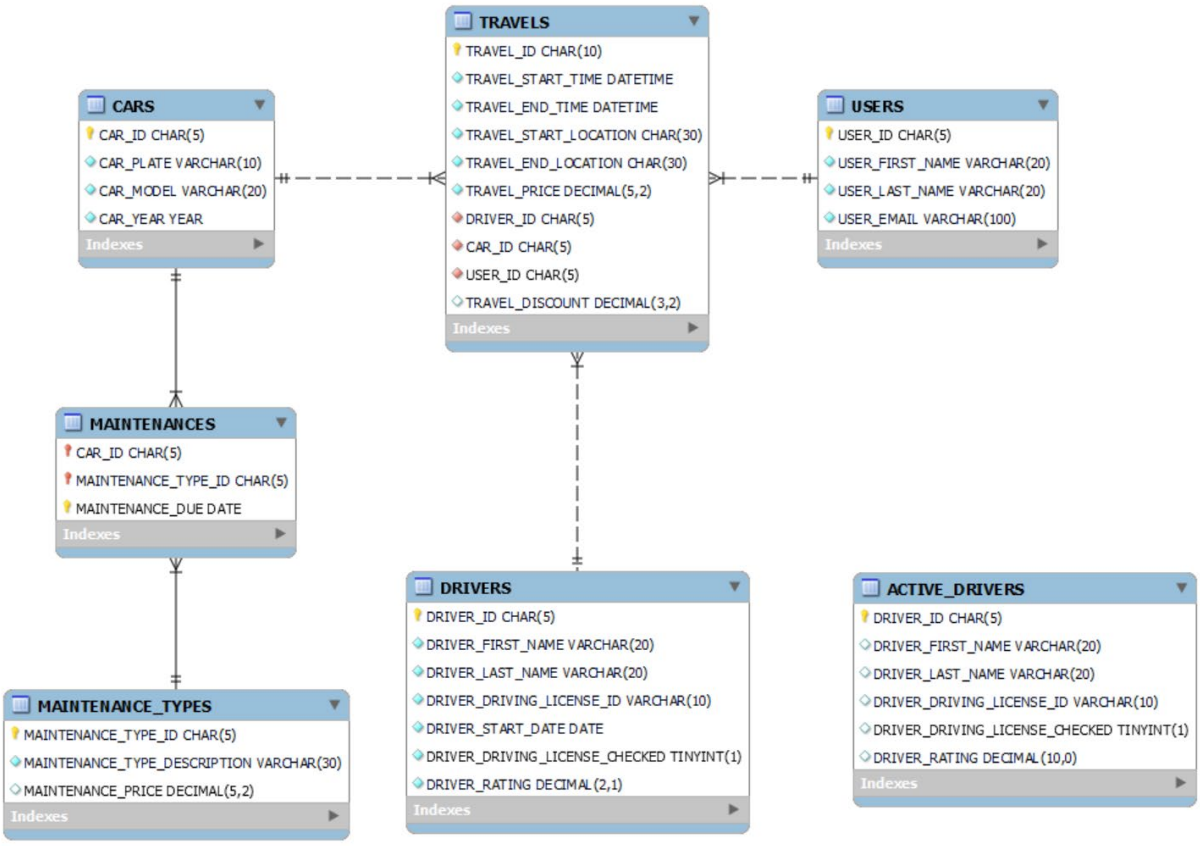
ClassExercise06

Summary

InstantRide is the new ride sharing application in the city and it has just started its operations. With the help of the **InstantRide** mobile application, the users request a ride with their location. Drivers and cars are assigned to the request; and then the driver picks up the user to ride to their requested location. Information for the users, drivers and cars is stored in the database as well as the travel transactions.

Database Schema



In the USERS table, information for the users is stored with their first name, last name and email.

In the DRIVERS table, all the drivers who use **InstantRide** are stored with their name, driving license number and check and rating information.

In the CARS table, all the cars in the **InstantRide** system are kept with the license plate, model, and year.

Finally, the transactions of the rides are stored in the TRAVELS table. For each travel, start and end time with location are stored. In addition, the involved driver, car, and user are listed for each drive. Price and discount information are also available in the database.

You, as the database administrator, are assigned to collect and manage transactional data of the **InstantRide** operations. Your main task is to create SQL scripts to help other teams to retrieve the requested data.

Instructions

Use file *ClassExercise06-SetupDB.sql* to create the *InstantRide* database in your RDS and populate the tables with starter data. You should not change the setup file, but you may want to read it for a sense of the data in the database.

Using the structure of the *InstantRide06* database shown above, create SQL commands to answer the problems in SQL Lab *ClassExercise06.sql*. TEST your solution before submitting them to the autograder!

NOTE: The problems in this set build on one another. The setup file for this set has already applied the changes made in any previous set.

In these tasks, you will edit the SQL commands until they run successfully against the *InstantRide* database. Once the tasks run successfully, submit the file to the autograder. Make sure you keep your code in the locations indicated and DO NOT change the name of the file!