

Group 13

Team Members

Maeve Tra (ntra@purdue.edu)

Ishika Vachali (ivachali@purdue.edu)

Arpit Dhawan(dhawan9@purdue.edu)

Tri Tran (tran197@purdue.edu)

Project Title: Car Rental

Project Description: Our project will be based on the car rental business model.

Database Description: We have created our database in MYSQL. The version of MYSQL we are using is MYSQL 8.0. We used MYSQL workbench as an interface to work in the database.

The database is named rental_car.

This database has 6 tables.

Following is the description of the tables.

We have included Candidate keys, Primary keys and Foreign keys in the descriptions.

Table 1: car_types

Description: This table holds car types data. Some of the examples of car types in our database are Sedan, SUV, Minivan etc.

Keys:

Candidate Keys: The only candidate key we have for this table is car_type_id and we have this key as the primary key of this table. It is an auto increment integer field.

Primary Key: The car_type_id is the primary key of the table. This will be the foreign key in the cars table.

Foreign keys: There are no foreign keys in this table.

Indexes: We have the primary key as indexed and also an additional index on car_type field. Since car_type is not the primary key of the table, the database will create a non clustered index on car_type. The reason we choose to put an index on the car_type column is because we have a search criterion on our application page where the users will be able to search a car by car_type.

Table Definition:

car_types

car_type_id int primary key not null auto_increment,

car_type varchar(200) not null,

index(car_type)

TABLE 2: cars

Description: This table holds the data for each car we have in our database. This table will be used to identify the car by its unique_id. It holds columns like color,make,model,year and vin number of the car. There is a field called available_yes_no, this will be tagged as 1 if the car is available for rent else it is tagged as zero.

Keys:

Candidate keys: We have two candidate keys in this table. The first one is the Car_id and the second one is the VIN number field. For our database we are using car_id as the primary key of this table.

Primary key: The car_id is the primary key of the table. This will be the foreign key in the reservation table.

Foreign Keys: The car_type_id is the foreign key in this table, which comes from the car_types table. The current_location_id is another foreign key in this table and it comes from the locations table. This field will hold the current_location_id of the car and if the car is in use this field will just hold null implying that the car is in use.

Indexes: We have one additional index on this table other than the one on the primary key. It is on column rental_price_per_hour. The reason we choose to put an index on this column is because we have a search criterion on our application page where the users will be able to search a car by the price of the car.

Table Definition:

cars

Car_id int Primary key not null Auto_increment,

Car_type_id int not null,

Make varchar(200),

Model varchar(200),

Make_Year int,

Color varchar(50),

VIN varchar(200) not null,

Current_location_id int ,

Available_for_rent tinyint not null,

Rental_price_per_hour decimal(19,4) not null,

FOREIGN KEY (car_type_id) REFERENCES car_types(car_type_id),

FOREIGN KEY (current_location_id) REFERENCES locations(location_id),

index(rental_price_per_hour)

TABLE 3: Customers

Description: This table holds the customer information. Each user who wants to use the application needs to first register on the web page to be able to use the website. Each user will have to provide basic information like name contact information etc and then create a username and password for their account. All this information is then stored in the customer table.

Keys

Candidate keys: We have two candidate keys for this table. One is the customer_login_username which will be unique and the other one will be the customer_id which is also unique.

Primary key: We have chosen Customer_id as the primary key of this table. It will also be referenced in the reservations table as a foreign key.

Foreign keys: We have a foreign key in this table, it is called the customer_address_id and it links the customers table with customer_address table.

Indexes: We have created an index on customer_login_username field as the username will be used to access the website.

Table definition:

Customer_id int Primary key not null auto_increment,

Customer_First_name varchar(200) not null,

Customer_last_name varchar(200) not null,

Customer_Phone_Number varchar(30),

Customer_email varchar(200) not null,

Customer_login_username varchar(200) not null,

Customer_login_password varchar(200) not null ,

Insurance_yes_no tinyint,

Customer_address_id int not null

**FOREIGN KEY (customer_address_id) REFERENCES
customer_address(customer_address_id),**

index(customer_login_username)

TABLE 4: Customer_address

Description: This table holds the customer address information.

Keys

Candidate keys: We have two candidate keys for this table. One is the customer_address_id which will be unique and the other one will be the combination of customer_address1,customer_address2,customer_city,customer_state attributes..

Primary key: We have chosen customer_address_id as the primary key of this table.

Foreign keys: We have no foreign keys in this table.

Indexes: We have no additional indexes on this table other than the one on the primary key.

Table definition:

customer_address_id int Primary key not null auto_increment,
customer_address1 varchar(500) not null,
customer_address2 varchar(200) not null,
customer_city varchar(200),
customer_state varchar(20) not null,
customer_zip varchar(10) not null)

TABLE 5: Locations

Description: This table holds location information. This will be referenced when a user creates reservations in order to specify pickup and dropoff locations.

Keys

Candidate keys: The first candidate key for this table is the location_id. The second candidate key for this table is a combination of Location_Address1, Location_address2, Location_City, and Location_State attributes. For ease of design we have chosen location_id as the primary key for this table.

Primary key: Location_id serves as a primary key for this table. It will also be referenced as a foreign key in Reservations through pickup_location_id and dropoff_location_id.

Foreign keys: No foreign keys

Indexes: We have created an index on location_zip as this field will be frequently used to search the cars. It will be a non clustered index.

Table definition:

Location_id int primary key not null auto_increment ,

Location_description varchar(500),

Location_Address varchar(500) not null,

Location_City varchar(200) not null,

Location_State varchar(30) not null,

Location_Zip varchar(5) not null,

index(location_zip)

TABLE 6: Reservations

Description: This is a table where each reservation will be recorded. The customer will make a reservation on an available car according to his/her selection criterion. This table will hold past and current reservations for each customer. It will have details like the car on which the reservation was made, the location it was picked up from and the drop off location of the car. It will also store the total cost of that reservation.

Keys

Candidate key: The reservation_id is the candidate key for this table. Another candidate key for this table is the combination of Customer_id, Car_id and Pickupdatetime attributes. For the ease of database design we have chosen the reservation_id as the primary key of this table.

Primary key: Reservation_id will be the primary key of this table. It is an auto increment integer type field.

Foreign keys: Car_id from the cars table will be a foreign key to this table. The customer_id is a foreign key in this table from the customers table. Also the pickup_location_id and dropoff_locations_id are the foreign keys from the locations table.

Indexes: We do not have any other additional indexes on this table.

Table definition:

Reservation_id int Primary key not null auto_increment,

Customer_id int not null,

Car_id int not null,

Pickupdatetime datetime,

DropoffDatetime datetime,

Pickup_location_id int,

Dropoff_Location_id int,

Rental_Cost decimal(19,4),

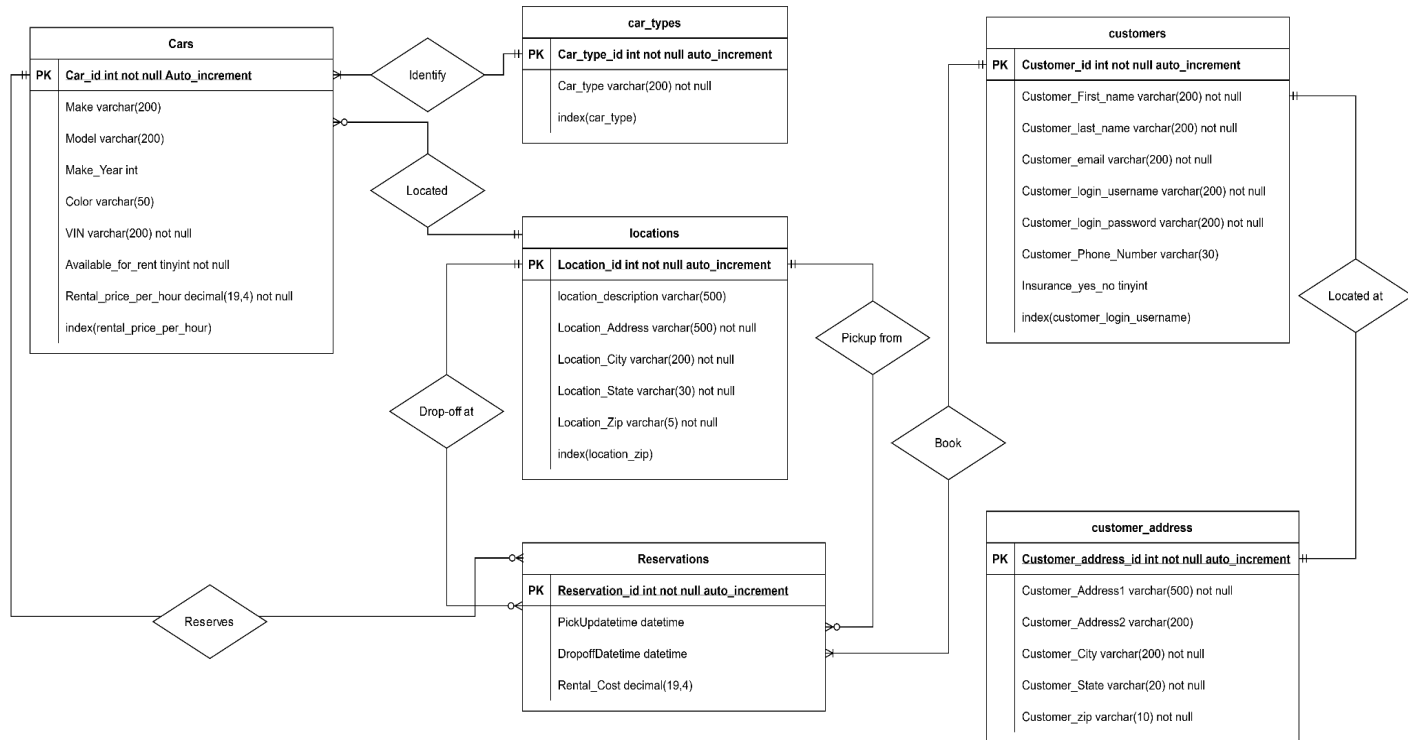
FOREIGN KEY (customer_id) REFERENCES customers(customer_id),

FOREIGN KEY (car_id) REFERENCES cars(car_id),

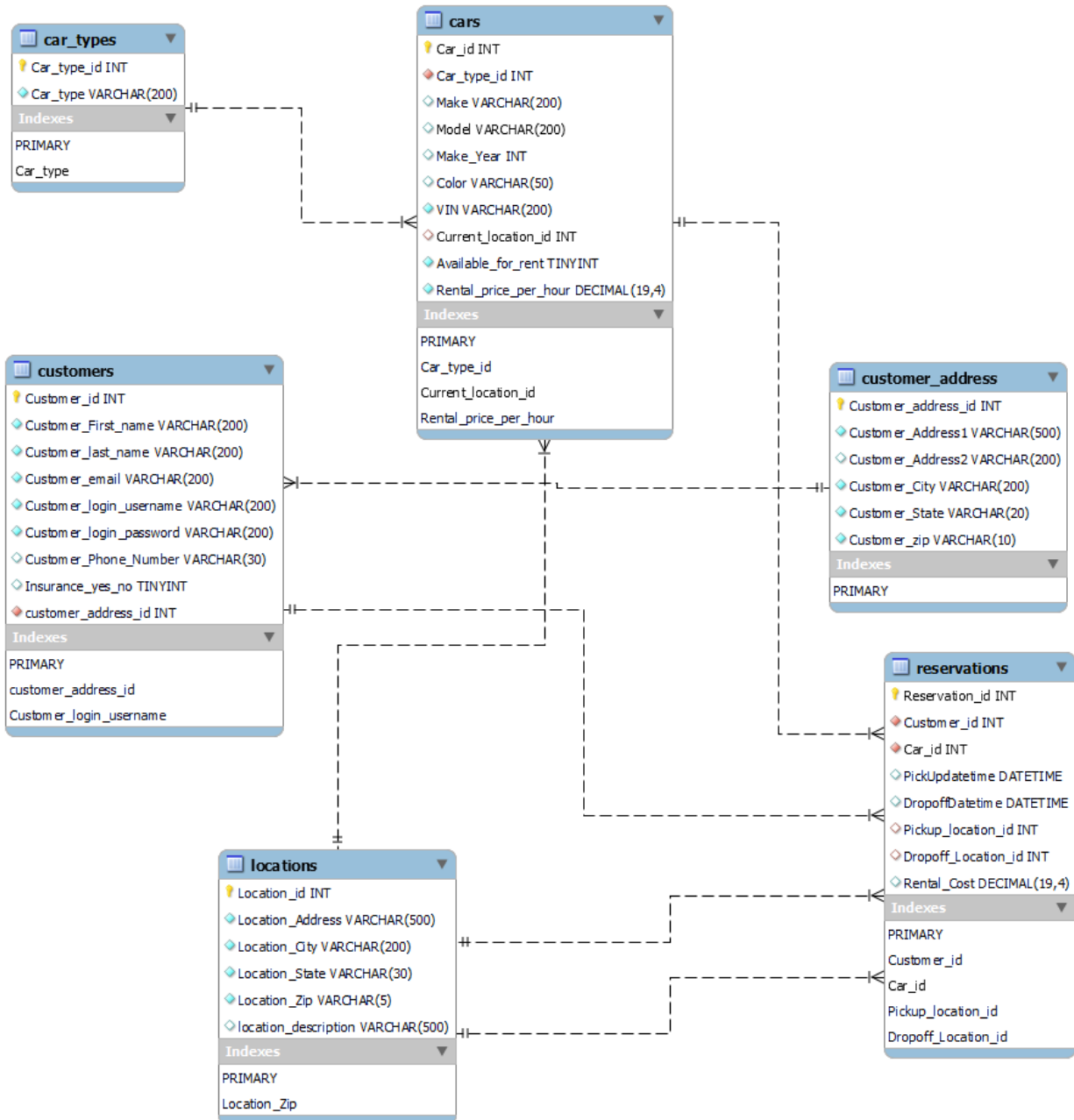
FOREIGN KEY (pickup_location_id) REFERENCES locations(location_id),

FOREIGN KEY (dropoff_location_id) REFERENCES locations(location_id)

ERD



Model Diagram from the database



RELATIONAL SCHEMA

1. car_types(car_type_id, car_type)

2. cars(car_id, car_type_id , make, model ,make_year ,color,VIN,Description ,Available_for_rent,Rental_price_per_hour,current_location_id)

FOREIGN KEY (car_type_id) REFERENCES car_types(car_type_id)

FOREIGN KEY (current_location_id) REFERENCES locations(location_id)

Another candidate key in the cars table other than car_id is the VIN number field.

3. customers(customer_id,Customer_First_name ,Customer_last_name, Customer_email , Customer_login_username , Customer_login_password , Customer_Phone_Number ,Insurance_yes_no,customer_address_id)

FOREIGN KEY (customer_address_id) REFERENCES customer_address(customer_address_id)

Another candidate key in the customer table other than customer_id is the customer_login_username field.

4. customer_address: (customer_address_id, customer_address1,customer_address2,customer_city,customer_state,customer_zip)

Another candidate key in this table other than customer_address_id is the combination of customer_address1,customer_address2,customer_city,customer_state attributes.

5. locations(location_id ,location_description,location_Address ,location_City ,location_state ,location_Zip)

Another candidate key in this table other than location_id is the combination of location_Address ,location_City ,location_state attributes.

6. reservations(Reservation_id ,Customer_id, car_id ,PickUpdatetime , DropoffDatetime, Pickup_location_id ,Dropoff_Location_id ,Rental_Cost)

Another candidate key for this table other than the reservation_id is the combination of Customer_id ,Car_id and PickUpdatetime keys.

FOREIGN KEY (customer_id) REFERENCES customers(customer_id)

FOREIGN KEY (car_id) REFERENCES cars(car_id)

FOREIGN KEY (pickup_location_id) REFERENCES locations(location_id)

FOREIGN KEY (dropoff_location_id) REFERENCES locations(location_id)