**Introduction**

In today's fast-paced world, the need for convenient and flexible transportation options is more pronounced than ever. Whether for business trips, vacations, or daily commuting, car rentals have become an essential part of modern urban living. Recognizing this growing demand, Rent-a-Ride car rental service was established to provide customers with a wide variety of vehicles tailored to their specific needs and preferences.

**Objective**

The primary objective of Rent-a-Ride is to offer a seamless car rental experience that prioritizes customer satisfaction and affordability. By leveraging cutting-edge technology and maintaining a diverse fleet of vehicles, Rent-a-Ride aims to become a go-to solution for travelers and locals alike.

**DATABASE SCHEMA DESIGN**

**1. Cars Collection**

This collection will store information about the cars available for rental.

**Fields:**

* car\_id: String (unique identifier for the car)
* make: String (car make, e.g., Ford, Honda)
* model: String (car model, e.g., F-150, CR-V)
* year: Number (manufacturing year)
* color: String (car color)
* rental\_price\_per\_day: Number (daily rental price)
* availability: Boolean (whether the car is available for rental)

**2. Customers Collection**

This collection will hold information about the customers.

**Fields:**

* customer\_id: String (unique identifier for the customer)
* first\_name: String
* last\_name: String
* email: String
* phone\_number: String
* address: String
* date\_of\_birth: Date

**3. Rentals Collection**

This collection will store information about rental transactions.

**Fields:**

* rental\_id: String (unique identifier for the rental)
* customer\_id: Reference to Customer
* car\_id: Reference to Car
* rental\_date: Date (start date of the rental)
* return\_date: Date (end date of the rental)
* total\_price: Number (total rental fee)

**JSON Format**

**Cars:**

{

"car\_id": "00001",

"make": "Ford",

"model": "F-150",

"year": 2023,

"color": "Red",

"rental\_price\_per\_day": 81,

"availability": true

}

**Customers:**

{

"customer\_id": "C001",

"first\_name": "John",

"last\_name": "Doe",

"email": "johndoe@example.com",

"phone\_number": "123-456-7890",

"address": "123 Main St",

"date\_of\_birth": "1990-01-01"

}

Rentals:

{

"rental\_id": "R001",

"customer\_id": "C001",

"car\_id": "00001",

"rental\_date": "2023-08-01",

"return\_date": "2023-08-07",

"total\_price": 567

}

Example of inserting data

db.cars.insertMany([

{

"car\_id": "00001",

"make": "Toyota",

"model": "Cruze",

"year": 2021,

"color": "Red",

"rental\_price\_per\_day": 47,

"availability": true

},

{

"car\_id": "00002",

"make": "Honda",

"model": "Focus",

"year": 2020,

"color": "Red",

"rental\_price\_per\_day": 40,

"availability": true

},

{

"car\_id": "00003",

"make": "Ford",

"model": "X5",

"year": 2018,

"color": "Blue",

"rental\_price\_per\_day": 77,

"availability": false

}

]);

For customers:

db.customers.insertMany([

{

"customer\_id": "00001",

"first\_name": "Garrett",

"last\_name": "Carter",

"email": "phillipsaustin@douglas-bartlett.info",

"phone\_number": "040-918-6153",

"date\_of\_birth": "1993-04-15",

"license\_number": "54-J530",

"address": "6028 Ann Ridge Apt. 258 Port Cynthiafurt, IL 22503",

"rented\_car\_id": null

},

{

"customer\_id": "00002",

"first\_name": "Ruth",

"last\_name": "Martin",

"email": "heather08@phillips.com",

"phone\_number": "(850)885-7328",

"date\_of\_birth": "1959-09-09",

"license\_number": "8N SM522",

"address": "6003 Peterson Hills Suite 995 West Zachary, CT 11133",

"rented\_car\_id": null

}

]);

Rentals

db.rentals.insertMany([

{

"rental\_id": "00001",

"customer\_id": "00009",

"car\_id": "00017",

"rental\_start\_date": "2023-01-16",

"rental\_end\_date": "2023-02-12",

"total\_price": 1674

},

{

"rental\_id": "00002",

"customer\_id": "00095",

"car\_id": "00038",

"rental\_start\_date": "2022-06-22",

"rental\_end\_date": "2022-07-01",

"total\_price": 900

},

{

"rental\_id": "00018",

"customer\_id": "00004",

"car\_id": "00012",

"rental\_start\_date": "2023-05-07",

"rental\_end\_date": "2023-05-08",

"total\_price": 85

}

]);

Querying Data :

**Find all cars of a specific make (e.g., "BMW"):**

db.cars.find({ "make": "BMW" });

 **Find all cars released in a specific year or range of years (e.g., from 2015 to 2020):**

db.cars.find({ "year": { "$gte": 2015, "$lte": 2020 } });

 **Find all customers who have rented a car of a specific make (e.g., "BMW"):**

 db.customers.aggregate([

{ "$lookup": {

"from": "rentals",

"localField": "customer\_id",

"foreignField": "customer\_id",

"as": "rental\_info"

}},

{ "$unwind": "$rental\_info" },

{ "$lookup": {

"from": "cars",

"localField": "rental\_info.car\_id",

"foreignField": "car\_id",

"as": "car\_info"

}},

{ "$unwind": "$car\_info" },

{ "$match": { "car\_info.make": "BMW" } }

]);

 **Find the total rental fees collected from a specific customer (e.g., customer\_id "00009"):**

db.rentals.aggregate([

{ "$match": { "customer\_id": "00009" } },

{ "$group": { "\_id": null, "total\_fees": { "$sum": "$total\_price" } } }

]);

 **Find the most popular car(s) based on the number of rentals:**

db.rentals.aggregate([

{ "$group": { "\_id": "$car\_id", "count": { "$sum": 1 } } },

{ "$sort": { "count": -1 } },

{ "$limit": 1 }

]);

### Calculate the average rental fee per make (brand)

javascript

db.rentals.aggregate([

{

"$lookup": {

"from": "cars",

"localField": "car\_id",

"foreignField": "car\_id",

"as": "car\_info"

}

},

{

"$unwind": "$car\_info"

},

{

"$group": {

"\_id": "$car\_info.make",

"average\_rental\_fee": { "$avg": "$total\_price" }

}

}

]);

### b. Calculate the total rental fees collected per month

javascript

db.rentals.aggregate([

{

"$group": {

"\_id": {

"month": { "$month": "$rental\_start\_date" },

"year": { "$year": "$rental\_start\_date" }

},

"total\_rental\_fees": { "$sum": "$total\_price" }

}

}

]);

### c. Find the top 3 most popular car makes among customers

javascript

db.rentals.aggregate([

{

"$lookup": {

"from": "cars",

"localField": "car\_id",

"foreignField": "car\_id",

"as": "car\_info"

}

},

{

"$unwind": "$car\_info"

},

{

"$group": {

"\_id": "$car\_info.make",

"rental\_count": { "$sum": 1 }

}

},

{

"$sort": { "rental\_count": -1 }

},

{

"$limit": 3

}

]);

Data Update (CRUD):

### a. Update the rental price for all cars of a specific make

db.cars.updateMany(

{ "make": "Honda" },

{ "$set": { "rental\_price\_per\_day": 2000 } }

);

### b. Update the contact information (phone number and address) for a specific customer

db.customers.updateOne(

{ "customer\_id": "00002" },

{

"$set": {

"phone\_number": 6456545467,

"address": “9 Shawford cresecent”

}

}

);