

# Node Js Assignment – Vehicle Rental Management API

## Description:

You are tasked with developing a Vehicle Rental Management API for a car rental company. The API will allow the company to manage their vehicle fleet, handle customer bookings, and track vehicle availability. Below are the specific requirements for this assignment:

### Vehicle Management:

- Implement endpoints to add, update, and delete vehicles in the rental fleet.
- Each vehicle should have properties like `make`, `model`, `year`, `fuel_type`, `seating_capacity`, and `availability_status`.

### Vehicle Search:

- Create endpoints to search for available vehicles based on criteria like `make`, `model`, `year`, and `seating_capacity`.
- Allow users to filter vehicles based on availability and fuel type.

### Customer Management:

- Implement endpoints to add, update, and delete customer information, including `name`, `email`, and `contact number`.

### Booking Management:

- Create endpoints for customers to make vehicle bookings for a specific date range.
- Ensure that vehicles are marked as unavailable during the booked period.
- Allow customers to view their current and past bookings.

### Vehicle Maintenance:

- Implement an endpoint that allows staff to mark vehicles as "under maintenance" and specify the maintenance period.
- Ensure that maintenance bookings do not overlap with customer bookings.

### Rental Statistics:

- Create an endpoint to provide rental statistics, such as the most popular vehicle, the number of bookings per vehicle, and average rental duration.

### Authentication and Authorization:

- Implement user authentication using JWT.
- Separate access for regular users (customers) and administrators (staff).

- Allow only administrators to perform vehicle management and maintenance operations.

#### Error Handling:

- Implement appropriate error handling and return informative error responses for various scenarios, such as invalid input or unauthorized access.

#### Unit Tests:

- Write comprehensive unit tests to validate the functionality of the API endpoints, especially for complex booking and availability scenarios.

#### Additional Considerations:

- Use Node.js and Express.js for building the API.
- Utilize a database (e.g., MongoDB) to store vehicle and booking information.
- Consider implementing a feature for notifying customers about upcoming bookings and staff about maintenance schedules.