

```

# CarRentalModule.py

from datetime import datetime, timedelta

class CarRental:
    def __init__(self, total_cars):
        self.total_cars = total_cars
        self.available_cars = total_cars
        self.rented_cars = 0
        self.inventory = {}

    def display_available_cars(self):
        print(f"Available Cars: {self.available_cars}")

    def rent_hourly(self, num_cars):
        return self._rent_car(num_cars, "hourly")

    def rent_daily(self, num_cars):
        return self._rent_car(num_cars, "daily")

    def rent_weekly(self, num_cars):
        return self._rent_car(num_cars, "weekly")

    def _rent_car(self, num_cars, rental_mode):
        if 0 < num_cars <= self.available_cars:
            self.available_cars -= num_cars
            current_time = datetime.now()
            self.inventory[num_cars] = {"rental_time": current_time, "rental_mode": rental_mode}
            self.rented_cars += num_cars
            return f"{num_cars} cars rented {rental_mode} at {current_time}."
        else:
            return "Invalid input or not enough cars available for rent."

    def return_cars(self, num_cars):
        if num_cars in self.inventory:
            rental_info = self.inventory.pop(num_cars)
            rented_time = rental_info["rental_time"]
            rental_mode = rental_info["rental_mode"]
            rented_duration = datetime.now() - rented_time
            self.available_cars += num_cars

            if rental_mode == "hourly":
                bill = rented_duration.seconds * 400 * num_cars // 3600 # Rs.400 per hour
            elif rental_mode == "daily":
                bill = rented_duration.days * 3000 * num_cars # Rs. 3000 per day
            elif rental_mode == "weekly":
                bill = rented_duration.days * 15000 * num_cars // 7 # Rs.6000 per week

            self.rented_cars -= num_cars
            return f"{num_cars} cars returned.\nRental period: {rented_duration}.\nBill: Rs.{bill}."
        else:
            return "Invalid input or car not rented."

class Customer:
    def __init__(self, name):
        self.name = name

    def request_cars(self, car_rental, num_cars, rental_mode):
        if rental_mode == "hourly":
            return car_rental.rent_hourly(num_cars)
        elif rental_mode == "daily":
            return car_rental.rent_daily(num_cars)
        elif rental_mode == "weekly":
            return car_rental.rent_weekly(num_cars)
        else:
            return "Invalid rental mode."

    def return_cars(self, car_rental, num_cars):
        return car_rental.return_cars(num_cars)

```