// Project Code :

// Code of Car Rental System Management :

import java.util.Scanner;

// Class representing a Car

class Car {

private String model;

private String registrationNumber;

private boolean isAvailable;

private double dailyRent; // Daily rent for the car

public Car(String model, String registrationNumber, double dailyRent) {

this.model = model;

this.registrationNumber = registrationNumber;

this.isAvailable = true; // Car is available by default

this.dailyRent = dailyRent; // Set daily rent

}

public String getModel() {

return model;

}

public String getRegistrationNumber() {

return registrationNumber;

}

public boolean isAvailable() {

return isAvailable;

}

public double getDailyRent() {

return dailyRent; // Return daily rent

}

public void rent() {

isAvailable = false;

}

public void returnCar() {

isAvailable = true;

}

}

// Class representing a Customer

class Customer {

private String name;

private String contact;

public Customer(String name, String contact) {

this.name = name;

this.contact = contact;

}

public String getName() {

return name;

}

public String getContact() {

return contact;

}

}

// Class to manage the car rental service

class CarRentalService {

private Car[] cars;

private int carCount;

public CarRentalService(int size) {

cars = new Car[size];

carCount = 0; // Initialize car count

}

public void addCar(Car car) {

if (carCount < cars.length) {

cars[carCount] = car;

carCount++;

System.out.println("Car added: " + car.getModel());

} else {

System.out.println("Cannot add more cars, maximum capacity reached.");

}

}

public void rentCar(String registrationNumber, Customer customer) {

for (int i = 0; i < carCount; i++) {

if (cars[i].getRegistrationNumber().equals(registrationNumber) && cars[i].isAvailable()) {

cars[i].rent();

System.out.println("Car rented successfully to " + customer.getName());

System.out.println("Daily Rent: $" + cars[i].getDailyRent());

return;

}

}

System.out.println("Car not available or does not exist.");

}

public void returnCar(String registrationNumber) {

for (int i = 0; i < carCount; i++) {

if (cars[i].getRegistrationNumber().equals(registrationNumber)) {

cars[i].returnCar();

System.out.println("Car returned successfully.");

return;

}

}

System.out.println("Car does not exist.");

}

public void displayCars() {

System.out.println("Available Cars:");

for (int i = 0; i < carCount; i++) {

if (cars[i].isAvailable()) {

System.out.println("Model: " + cars[i].getModel() +

", Registration Number: " + cars[i].getRegistrationNumber() +

", Daily Rent: $" + cars[i].getDailyRent());

}

}

}

}

// Main class to run the application

public class Main {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

// Create a CarRentalService with a fixed size of 5

CarRentalService rentalService = new CarRentalService(5);

while (true) {

System.out.println("\n1. Add a Car");

System.out.println("2. Display Available Cars");

System.out.println("3. Rent a Car");

System.out.println("4. Return a Car");

System.out.println("5. Exit");

System.out.print("Choose an option: ");

int choice = scanner.nextInt();

scanner.nextLine(); // Consume newline

switch (choice) {

case 1:

// Adding a new car

System.out.print("Enter Model of the car: ");

String model = scanner.nextLine();

System.out.print("Enter Registration Number of the car: ");

String regNum = scanner.nextLine();

System.out.print("Enter Daily Rent of the car: ");

double dailyRent = scanner.nextDouble();

scanner.nextLine(); // Consume newline

rentalService.addCar(new Car(model, regNum, dailyRent));

break;

case 2:

rentalService.displayCars();

break;

case 3:

System.out.print("Enter Registration Number of the car to rent: ");

String regNumRent = scanner.nextLine();

System.out.print("Enter your name: ");

String customerName = scanner.nextLine();

System.out.print("Enter your contact: ");

String customerContact = scanner.nextLine();

Customer customer = new Customer(customerName, customerContact);

rentalService.rentCar(regNumRent, customer);

break;

case 4:

System.out.print("Enter Registration Number of the car to return: ");

String regNumReturn = scanner.nextLine();

rentalService.returnCar(regNumReturn);

break;

case 5:

System.out.println("Thank you for using the Car Rental Service!");

scanner.close();

return;

default:

System.out.println("Invalid option. Please try again.");

}

}

}

}

