Cars24(---RETIRED---)

Grade settings: Maximum grade: 100
Disable external file upload, paste and drop external content: Yes
Based on: Cars24(---RETIRED---)
Run: Yes **Evaluate**: Yes
Automatic grade: Yes

Cars24 is one of the most trustworthy companies. For automobiles that are available in their branches, they have opted to create an application that would collect data. The management has asked you to create software that would return the name of their available vehicle.

Component Specification: CarInfo

Type (Class)	Attributes	Methods
CarInfo	String carId String carName String carType String city	Necessary getters, setters, and four arguments constructors are provided as part of the code skeleton.

Functional Requirement 1: Extract the details and create an object of the CarInfo class.

Type (Class)	Methods	Responsibilities
UserInterface	public static CarInfo **extractDetails**(String carDetails)	This method accepts carDetails as an argument and we extracts details by parsing the data of the Cars from the argument. Set these values to the **CarInfo** object and return this object.

Functional Requirement 2: Check the availability of the car and price based on carName, carType and city.

Type (Class)	Methods	Responsibilities
CarInfo	public String **checkAvailability**()	This method should check the availableCar and price based on the carName , carType and city

		Condition:
		• **carName, carType and city** are case *in-sensitive.* • The **carName** must be either one of ["*Nissan",* *"Ford*"], else return **"Not Available".** • The **city** must be either one of [*"New York", Denver", "Los Angels"*], else return **"Not Available"** • The **carType** must be either one of [*"Sedan","SUV","MUV"*], else return **"Not Available".** • *The string "Not Available" is case-sensitive*

Note: The class and methods should be declared as public, and all the attributes should be declared as private.

Find the availableCar and carPrice based on the below table

carName	carType	availableCar	carPrice
Nissan	Sedan	Kicks	8400.0
	SUV	Magnite	10800.0
	MUV	Terrano	14400.0
Ford	Sedan	Figo	4802.0
	SUV	Eco Sport	9605.0
	MUV	Endeavour	21600.0

For example:

V322:Ford:Sedan:Los Angels

Here the **city** is "Los Angels",**carName** is "Ford" and **carType** is "Sedan"

So, the **availableCar and price is** "*Figo*" and 4802.0

The main method in the UserInterface class is excluded from the evaluation. You are free to write your own code in the main method to invoke the business methods to check its correctness.

Note:

- In the Sample Input / Output provided, the highlighted text in bold corresponds to the input given by the user and the rest of the text represents the output.
- Ensure to follow the object-oriented specifications provided in the question.
- Ensure to provide the names for classes, attributes, and methods as specified in the question.
- Adhere to the code template, if provided.
- **Please do not use System.exit(0) to terminate the program.**

Sample Input/Output 1:

Enter the Car Details

V001:Nissan:SUV:New york

Car Id : V001

Car Name: Nissan

Car Type : suv

City : New york

Available car and price is: Magnite and $10800.0

Sample Input/Output 2:

Enter the Car Details

V002:Ford:MUV:Denver

Car Id : V002

Car Name: Ford

Car Type : MUV

City : Denver

Available car and price is: Endeavour and $21600.0

Sample Input/Output 3:

Enter the Car Details

V052:Honda:MUV:Denver

Invalid Details

[Explanation: carName is invaild]

Sample Input/Output 4:

Enter the Car Details

V037:Ford:Coupe:Denver

Invalid Details

[Explanation: carType is invaild]
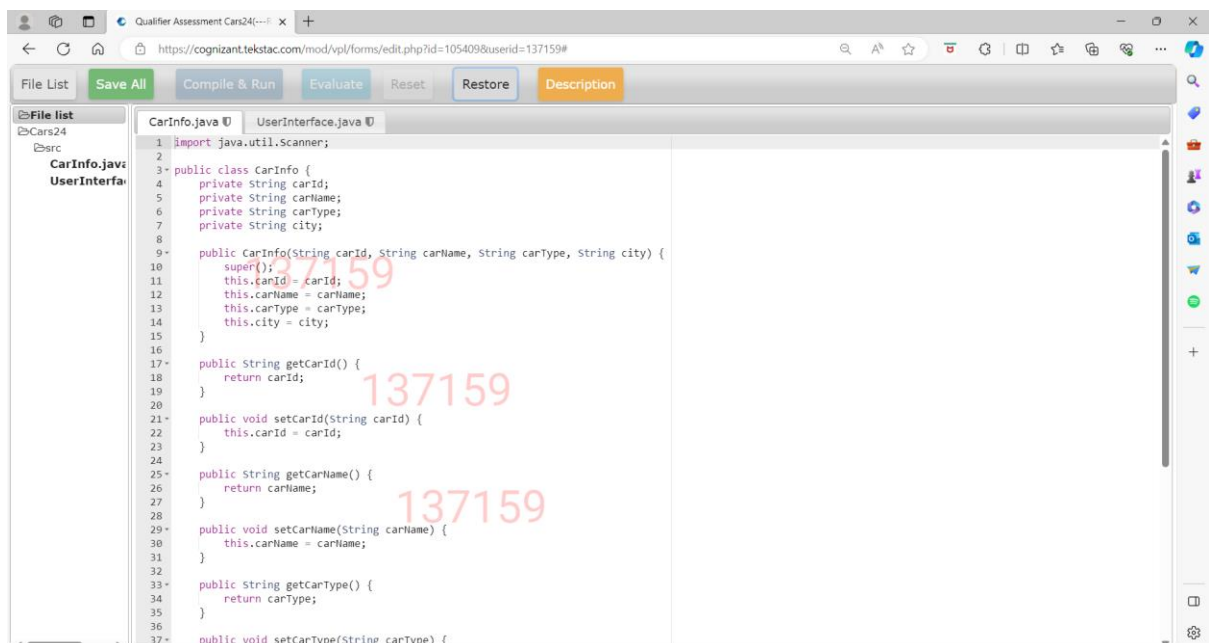
Sample Input/Output 5:

Enter the Car Details

V027:Ford:SUV:Boston

Invalid Details

[Explanation: city is invaild]

File List   Save All   Compile & Run   Evaluate   Reset   Restore   Description

File list
Cars24
  src
    CarInfo.java
    UserInterfac

CarInfo.java 🛈   UserInterface.java 🛈

```java
21    public void setCarId(String carId) {
22        this.carId = carId;
23    }
24
25    public String getCarName() {
26        return carName;
27    }
28
29    public void setCarName(String carName) {
30        this.carName = carName;
31    }
32
33    public String getCarType() {
34        return carType;
35    }
36
37    public void setCarType(String carType) {
38        this.carType = carType;
39    }
40
41    public String getCity() {
42        return city;
43    }
44
45    public void setCity(String city) {
46        this.city = city;
47    }
48
49    public String checkAvailability() {
50
51
52        // Fill the code here
53
54        return null;
55    }
56 }
57
```

File List   Save All   Compile & Run   Evaluate   Reset   Restore   Description

File list
Cars24
  src
    CarInfo.java
    UserInterfac

CarInfo.java 🛈   UserInterface.java 🛈

```java
1  import java.sql.Date;
2  import java.text.SimpleDateFormat;
3  import java.util.Scanner;
4
5  public class UserInterface
6  {
7      public static CarInfo extractDetails(String carDetails)
8      {
9          // Fill the code here
10
11         return null;
12     }
13
14     public static void main(String args[])
15     {
16
17         Scanner sc =new Scanner(System.in);
18         // Fill the code here
19
20     }
21 }
```