## Rajan Gautam 19BCP101

Div. II, CE 19 SOT, PDPU

# Pandit Deendayal Energy University School of Technology

**Design Pattern and Thinking (20CP210P)** 

B. Tech - Computer Science & Engineering (Sem-IV)

### **Table of Contents**

.ab 4 Assignment: Singleton Design Pattern for Car	.1
AIM: To write a Java Program to implement Singleton Design Pattern for Car	.1
CODE:	.1
Rajan_Singleton.java	.1
Rajan_TestCar.java	.1
SCREENSHOT OF IDE:	.2
SCREENSHOT OF OUTPUT:	3

#### Lab 4 Assignment: Singleton Design Pattern for Car

AIM: To write a Java Program to implement Singleton Design Pattern for Car

#### CODE:

#### Rajan\_Singleton.java

```
1. package Singleton 19BCP101 Rajan;
2.
3. public class Rajan Singleton {
        public static Rajan_Singleton rajanInstance = new Rajan_Singleton();
5.
        public int i;
        private Rajan Singleton()
7.
        System.out.println("Car Created...");
8.
9.
10.
        public static Rajan Singleton getInstance()
11.
12.
13.
             return rajanInstance;
14.
        }
15.
        public int getI()
16.
17.
18.
             return i;
19.
        }
20.
21.
        public static void setRajanInstance (Rajan Singleton rajanInstance)
22.
23.
             Rajan Singleton.rajanInstance = rajanInstance;
24.
        }
25.
        public void setI(int i)
26.
27.
28.
             this.i = i;
29.
30.
```

#### Rajan\_TestCar.java

```
1. package Singleton_19BCP101_Rajan;
2.
3. public class Rajan_TestCar {
4.    public static void main (String args[])
5.    {
6.         Rajan_Singleton rajan_car1 = Rajan_Singleton.getInstance();
7.         Rajan_Singleton rajan_car2 = Rajan_Singleton.getInstance();
```

```
Rajan Singleton rajan_car3 = Rajan_Singleton.getInstance();
8.
9.
10.
            rajan car1.setI(54);
11.
            rajan car2.setI(82);
                                           // It overrides the value 54.
12.
13.
            System.out.println("Speed: " + rajan car1.getI());
            System.out.println("Speed: " + rajan_car2.getI());
14.
15.
            print("My Bugatti", rajan car1);
16.
17.
            print("My Lamborghini", rajan car2);
18.
            print("My Mercedes", rajan car3);
19.
        }
20.
21.
        static void print (String rajan carName, Rajan Singleton rajan obj)
22.
23.
            System.out.println(String.format("My Car : %s, HashCode : %d",
   rajan carName, rajan obj.hashCode()));
24.
25.
    }
```

#### **SCREENSHOT OF IDE:**

```
Projects - Design Pattern/src/Singleton 19BCP101 Rajan/Rajan Singleton.java - Eclipse IDE
                                                                                                                                                                                                   О
Eile Edit Source Refactor Navigate Search Project Run Wind
Q 🔡 🐉
😩 Package Explorer 🛭 🕒 💲 🖇 🗀 🔲 🛽 Rajan_Singleton.java 🖂 📝 Rajan_TestCar.ji
                                   1 package Singleton_19BCP101_Rajan;
                                   3 public class Rajan Singleton {
      Builder_Car

Builder_Car

Factory_Phone_OS

Prototype_Car

Singleton_19BCP101_Rajan

1 Rajan_Singleton.java

2 Rajan_TestCar.java
                                           public static Rajan_Singleton rajanInstance = new Rajan_Singleton();
                                                                                                                                                                                    i: int
c Rajan_Singleton()
gettnstance(): Rajan_Singl
gett(): int
settRajanInstance(Rajan_Si
sett(int): void
                                           public int i;
                                           private Rajan_Singleton()
  SGUI in Java (1)
GUI in Java (2)
GUI in Java (3)
GUI in Java (3)
Java Lab
Java OOP
                                                System.out.println("Car Created...");
                                            public static Rajan_Singleton getInstance()
                                                return rajanInstance;
                                            public int getI()
                                                 return i;
                                           public static void setRajanInstance(Rajan_Singleton rajanInstance)
                                                 Rajan_Singleton.rajanInstance = rajanInstance;
                                           public void setI(int i)
                                                 this.i = i;
                                                                                                          Writable Smart Insert 31:2:545
```

Design Pattern and Thinking

```
o
Q : pr | 8
# Package Explorer ⋈ □ □ ② Rajan_Singleton_java ② Rajan_restor.gere □ □ 1 package Singleton_19BCP101_Rajan;
                                                                                                                                                                                                                                                                                                                                                                                                  □ BE Outline ⊠
                                                                                                                                                                                                                                                                                                                                                                                                          Bisingleton_19BCP101_Rajan

O, Rajan_TestCar

of main(String[]): void

A* print(String, Rajan_Singletc
  Calculator.App
Design Pattern

a JRE System Library JavaSE-14

Builder Car

B Builder Car

B Frototyp.Car

B Frototyp.Car

B Rajan_Singleton nagper or Rajan

B Rajan_Singleton rajan

B Rajan_TestCar [{

Rajan_Singleton rajan

Rajan_Singleton rajan
                                                                                                public static void main (String args[])
                                                                                                            Rajan_Singleton rajan_car1 = Rajan_Singleton.getInstance();
Rajan_Singleton rajan_car2 = Rajan_Singleton.getInstance();
Rajan_Singleton rajan_car3 = Rajan_Singleton.getInstance();
                                                                                                                                                                                                    // It overrides the value 54.
                                                                                                            rajan_car2.setI(82);
                                                                                                            System.out.println("Speed: " + rajan_car1.getI());
System.out.println("Speed: " + rajan_car2.getI());
                                                                                                            print("My Bugatti", rajan_car1);
print("My Lamborghini", rajan_car2);
print("My Mercedes", rajan_car3);
                                                                          16
17
                                                                           20
                                                                                                static void print(String rajan_carName, Rajan_Singleton rajan_obj)
                                                                                                            25
26 }
                                                                                                                                                                                                                                                                                                                            26:2:810
```

#### **SCREENSHOT OF OUTPUT:**



GitHub Link: https://github.com/rgautam320/Design-Pattern-Lab/tree/main/Singleton Car