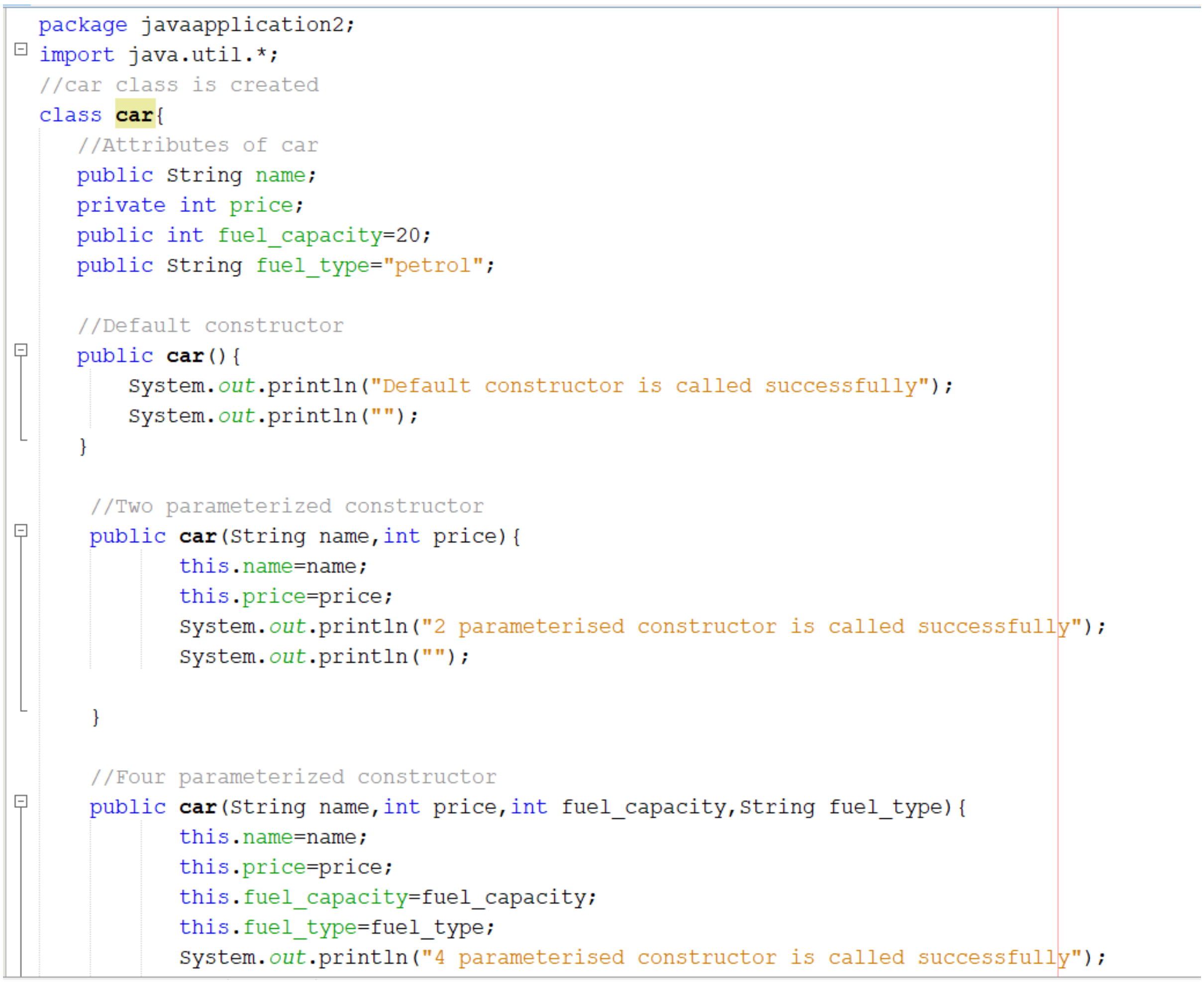
KARTHIK V
CIT
HUAWEI INTERNSHIP
ASSIGNMENT

JAVA CORE

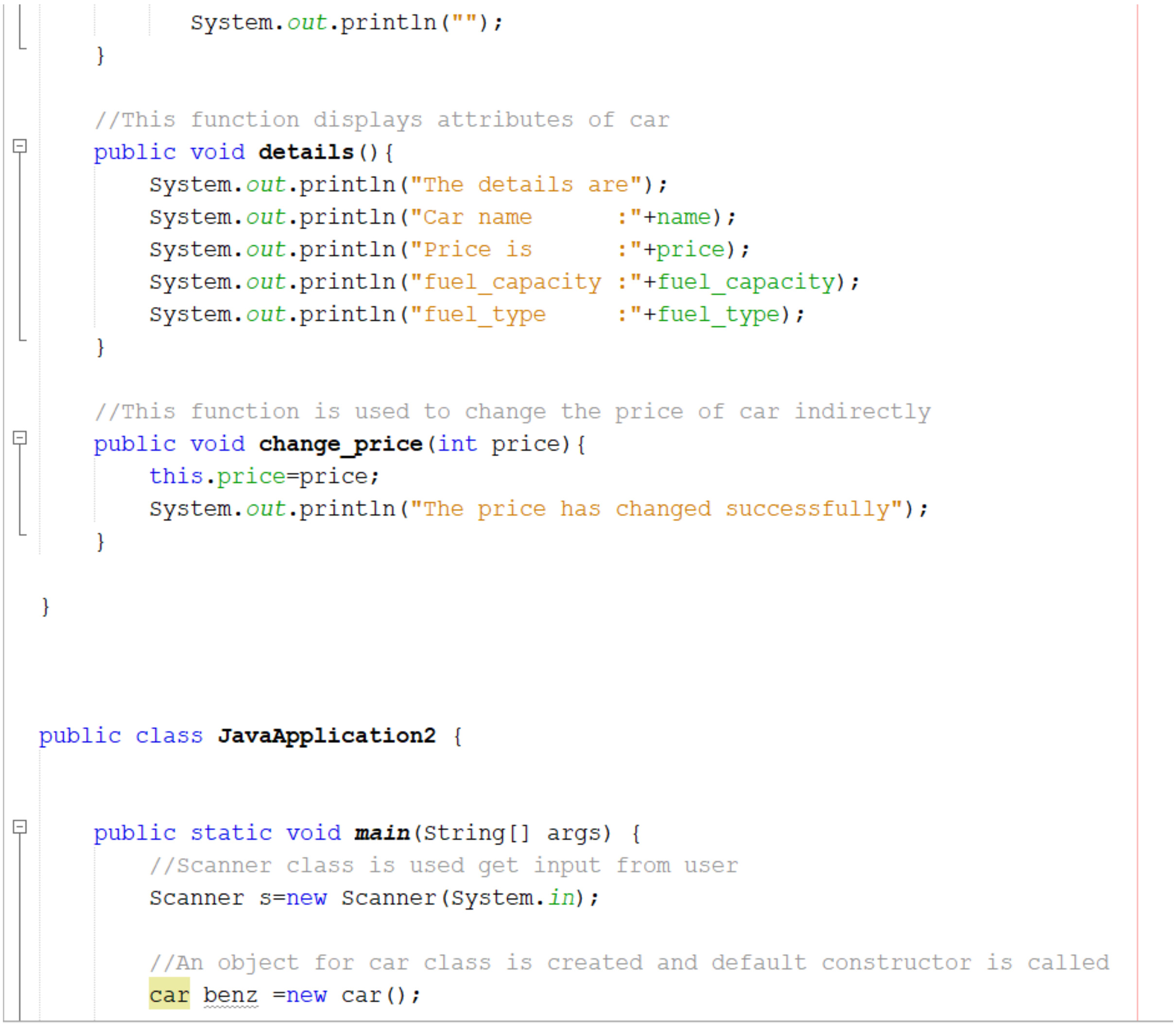
SESSON 7

1.Create an example by yourself to explain the capabilities of Constructor and Methods by creating a class like human or car The example program should have multiple constructor and method with access modifiers

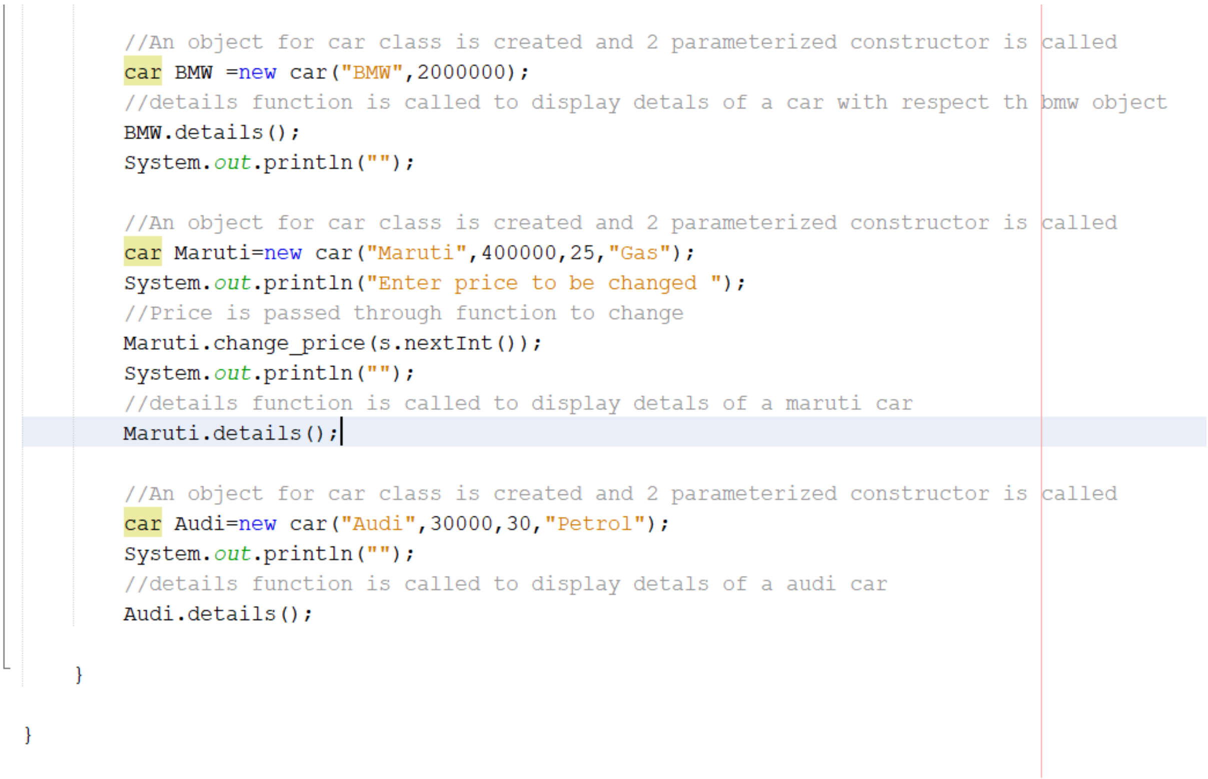
CODE



```
package javaapplication2;
import java.util.*;
//car class is created
class car{
   //Attributes of car
   public String name;
   private int price;
   public int fuel_capacity=20;
   public String fuel_type="petrol";

   //Default constructor
   public car(){
       System.out.println("Default constructor is called successfully");
       System.out.println("");
   }

    //Two parameterized constructor
    public car(String name,int price){
           this.name=name;
           this.price=price;
           System.out.println("2 parameterised constructor is called successfully");
           System.out.println("");

    }

    //Four parameterized constructor
    public car(String name,int price,int fuel_capacity,String fuel_type){
           this.name=name;
           this.price=price;
           this.fuel_capacity=fuel_capacity;
           this.fuel_type=fuel_type;
           System.out.println("4 parameterised constructor is called successfully");
```

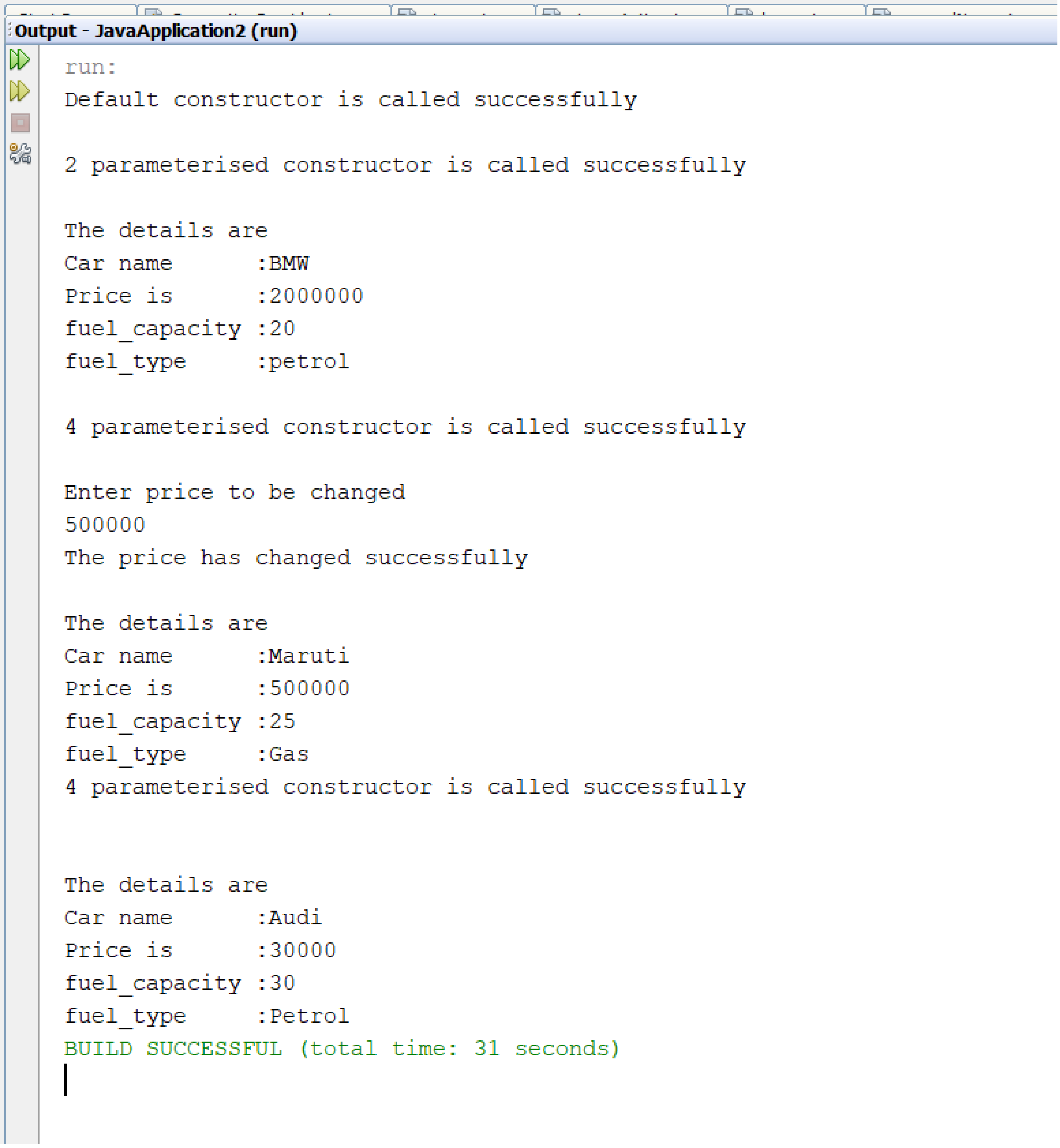


```
            System.out.println("");
    }

    //This function displays attributes of car
    public void details(){
        System.out.println("The details are");
        System.out.println("Car name      :"+name);
        System.out.println("Price is      :"+price);
        System.out.println("fuel_capacity :"+fuel_capacity);
        System.out.println("fuel_type     :"+fuel_type);
    }

    //This function is used to change the price of car indirectly
    public void change_price(int price){
        this.price=price;
        System.out.println("The price has changed successfully");
    }

}



public class JavaApplication2 {


    public static void main(String[] args) {
        //Scanner class is used get input from user
        Scanner s=new Scanner(System.in);

        //An object for car class is created and default constructor is called
        car benz =new car();
```



```
        //An object for car class is created and 2 parameterized constructor is called
        car BMW =new car("BMW",2000000);
        //details function is called to display detals of a car with respect th bmw object
        BMW.details();
        System.out.println("");

        //An object for car class is created and 2 parameterized constructor is called
        car Maruti=new car("Maruti",400000,25,"Gas");
        System.out.println("Enter price to be changed ");
        //Price is passed through function to change
        Maruti.change_price(s.nextInt());
        System.out.println("");
        //details function is called to display detals of a maruti car
        Maruti.details();

        //An object for car class is created and 2 parameterized constructor is called
        car Audi=new car("Audi",30000,30,"Petrol");
        System.out.println("");
        //details function is called to display detals of a audi car
        Audi.details();

    }

}
```

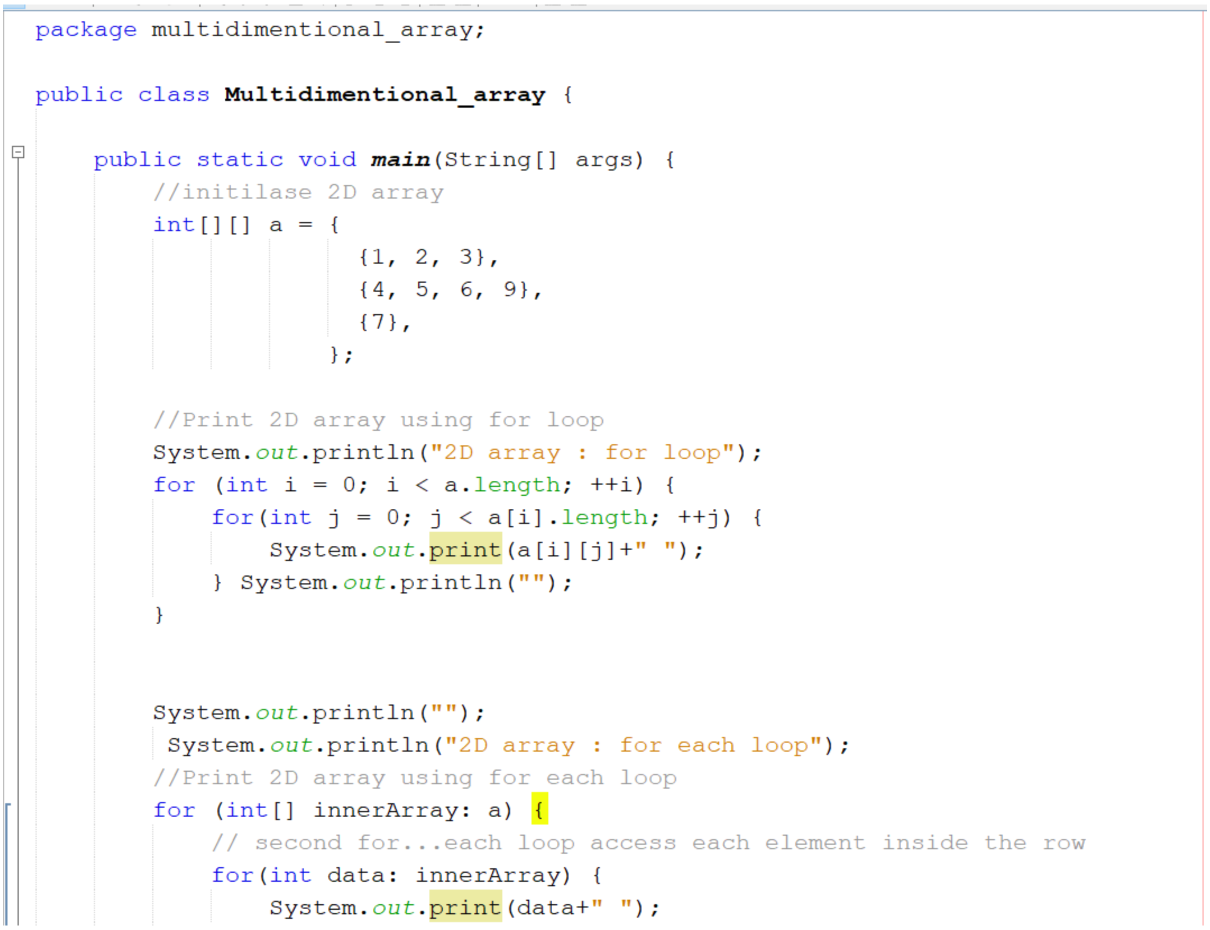
OUTPUT



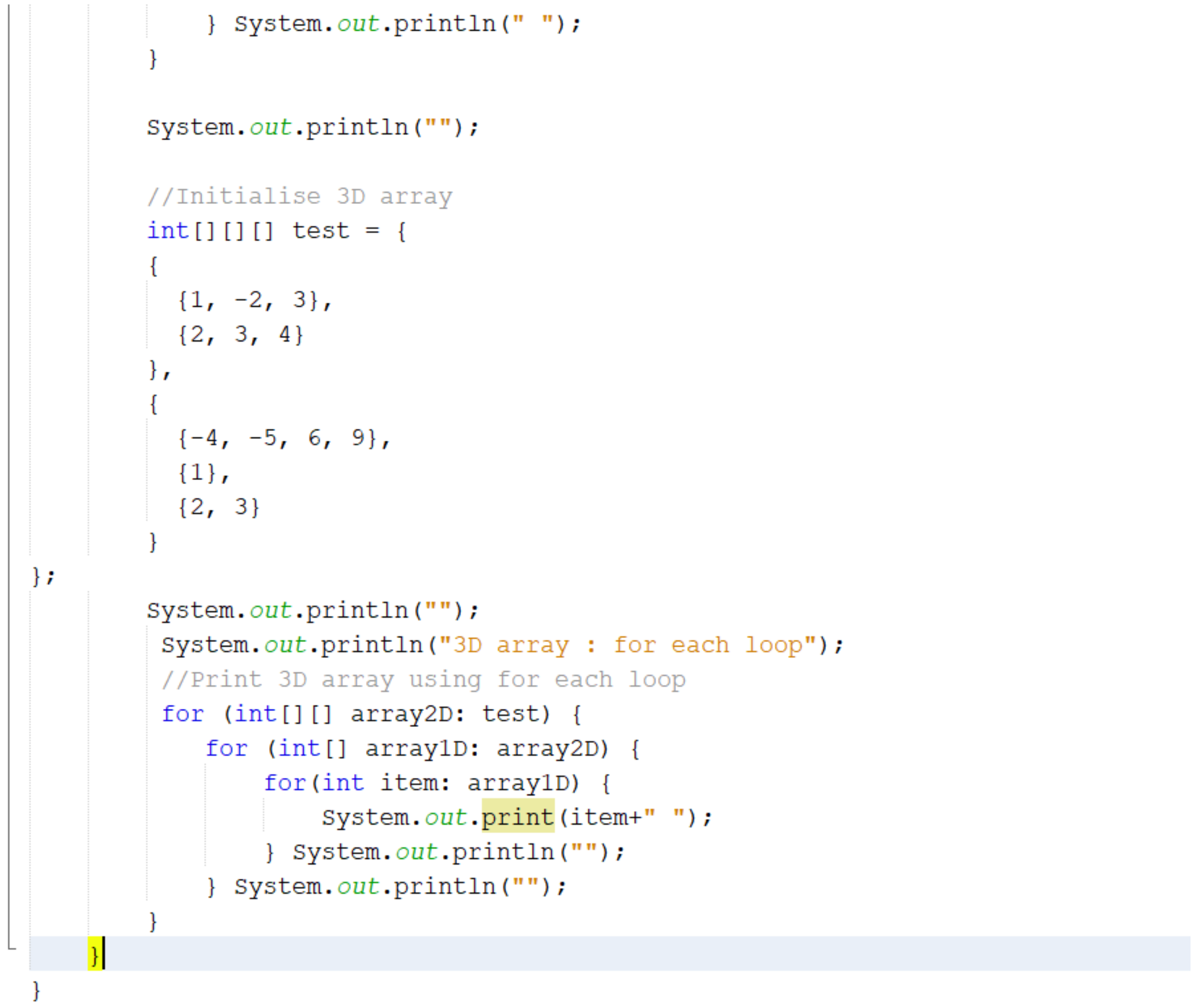
```
run:
Default constructor is called successfully

2 parameterised constructor is called successfully

The details are
Car name      :BMW
Price is      :2000000
fuel_capacity :20
fuel_type     :petrol

4 parameterised constructor is called successfully

Enter price to be changed
500000
The price has changed successfully

The details are
Car name      :Maruti
Price is      :500000
fuel_capacity :25
fuel_type     :Gas
4 parameterised constructor is called successfully


The details are
Car name      :Audi
Price is      :30000
fuel_capacity :30
fuel_type     :Petrol
BUILD SUCCESSFUL (total time: 31 seconds)
```

2. Multidimensional arrays

CODE

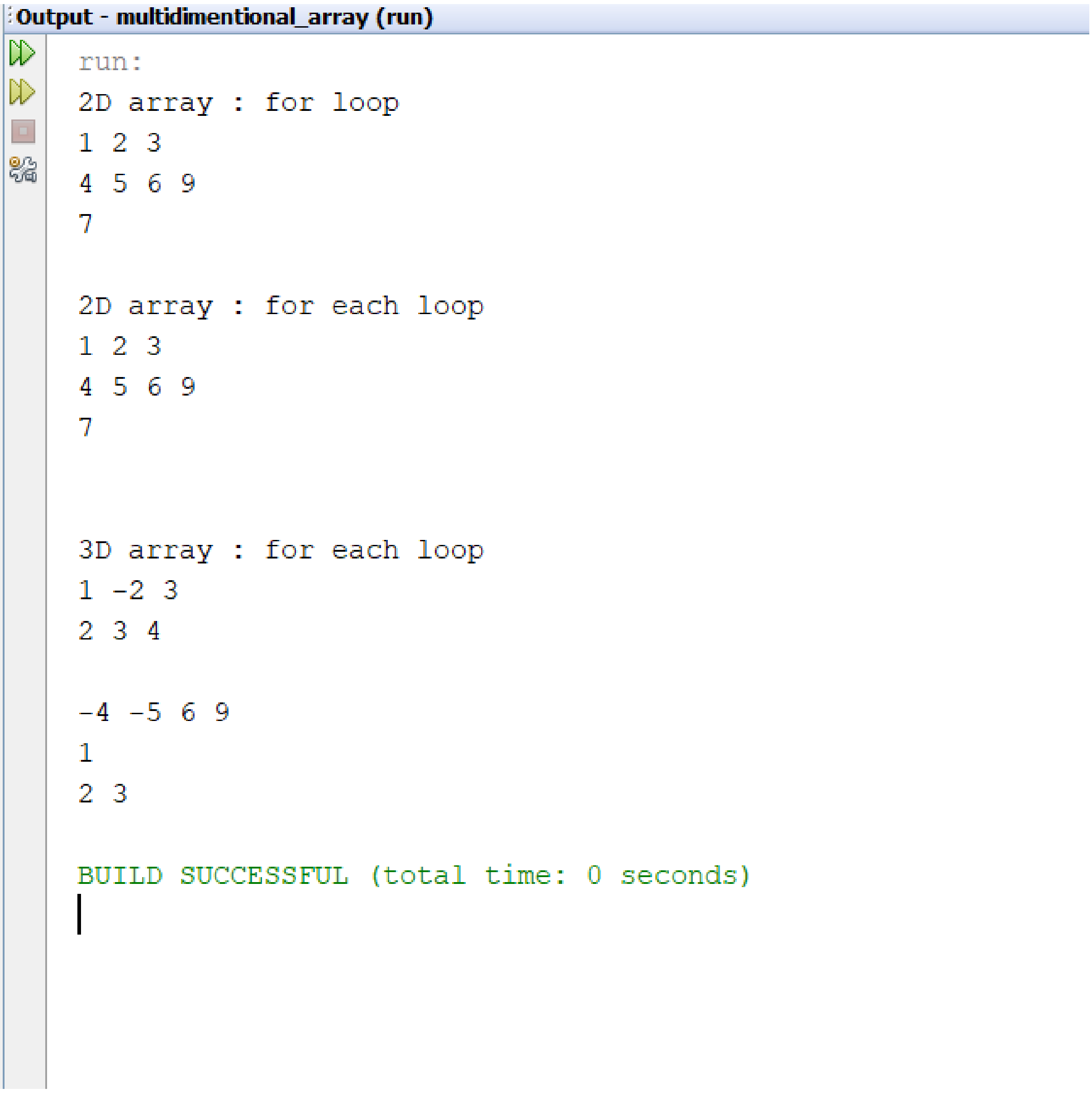


```
package multidimentional_array;

public class Multidimentional_array {

    public static void main(String[] args) {
        //initilase 2D array
        int[][] a = {
                      {1, 2, 3},
                      {4, 5, 6, 9},
                      {7},
                    };

        //Print 2D array using for loop
        System.out.println("2D array : for loop");
        for (int i = 0; i < a.length; ++i) {
            for(int j = 0; j < a[i].length; ++j) {
                System.out.print(a[i][j]+" ");
            } System.out.println("");
        }


        System.out.println("");
         System.out.println("2D array : for each loop");
        //Print 2D array using for each loop
        for (int[] innerArray: a) {
            // second for...each loop access each element inside the row
            for(int data: innerArray) {
                System.out.print(data+" ");
```



```
            } System.out.println(" ");
        }

        System.out.println("");

        //Initialise 3D array
        int[][][] test = {
        {
          {1, -2, 3},
          {2, 3, 4}
        },
        {
          {-4, -5, 6, 9},
          {1},
          {2, 3}
        }
};
        System.out.println("");
         System.out.println("3D array : for each loop");
         //Print 3D array using for each loop
         for (int[][] array2D: test) {
            for (int[] array1D: array2D) {
                for(int item: array1D) {
                    System.out.print(item+" ");
                } System.out.println("");
            } System.out.println("");
        }
    }
}
```

OUTPUT



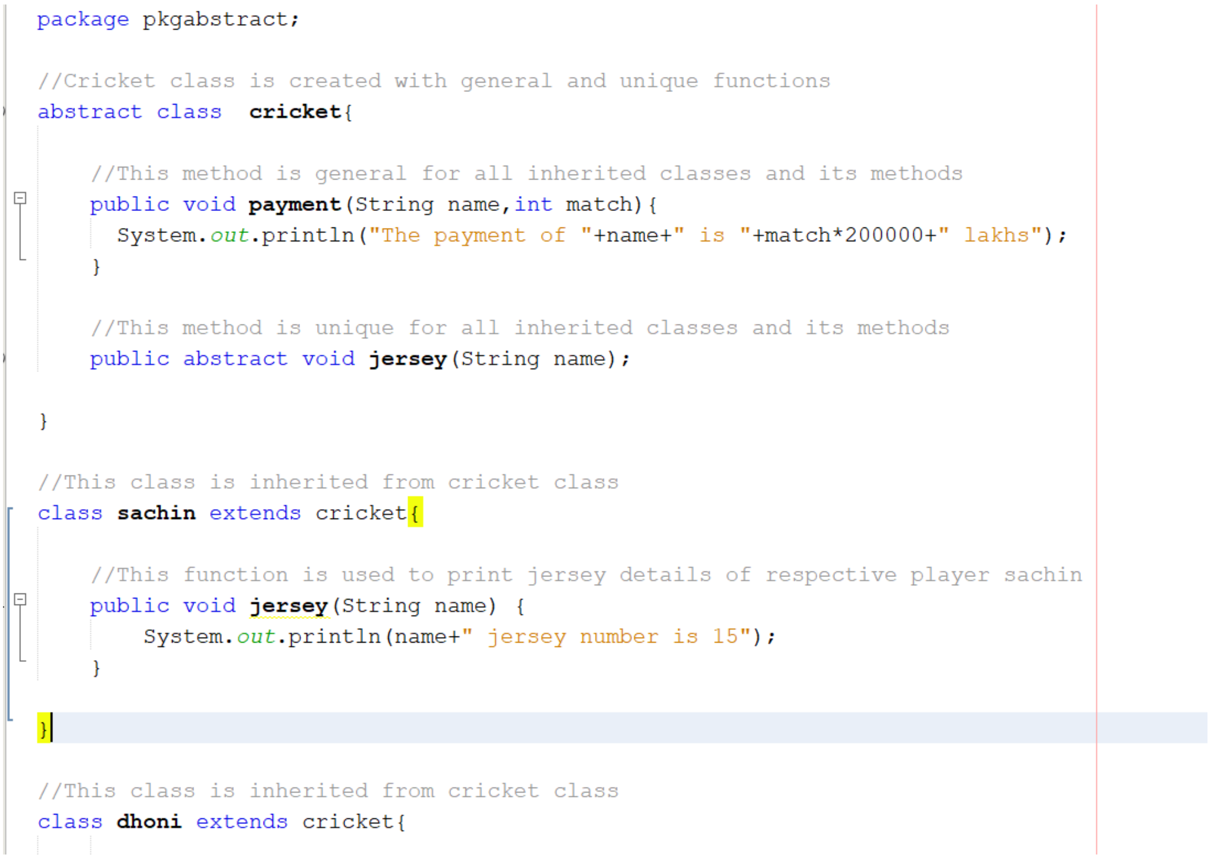
Output - multidimentional_array (run)

```
run:
2D array : for loop
1 2 3
4 5 6 9
7

2D array : for each loop
1 2 3
4 5 6 9
7


3D array : for each loop
1 -2 3
2 3 4

-4 -5 6 9
1
2 3

BUILD SUCCESSFUL (total time: 0 seconds)
```

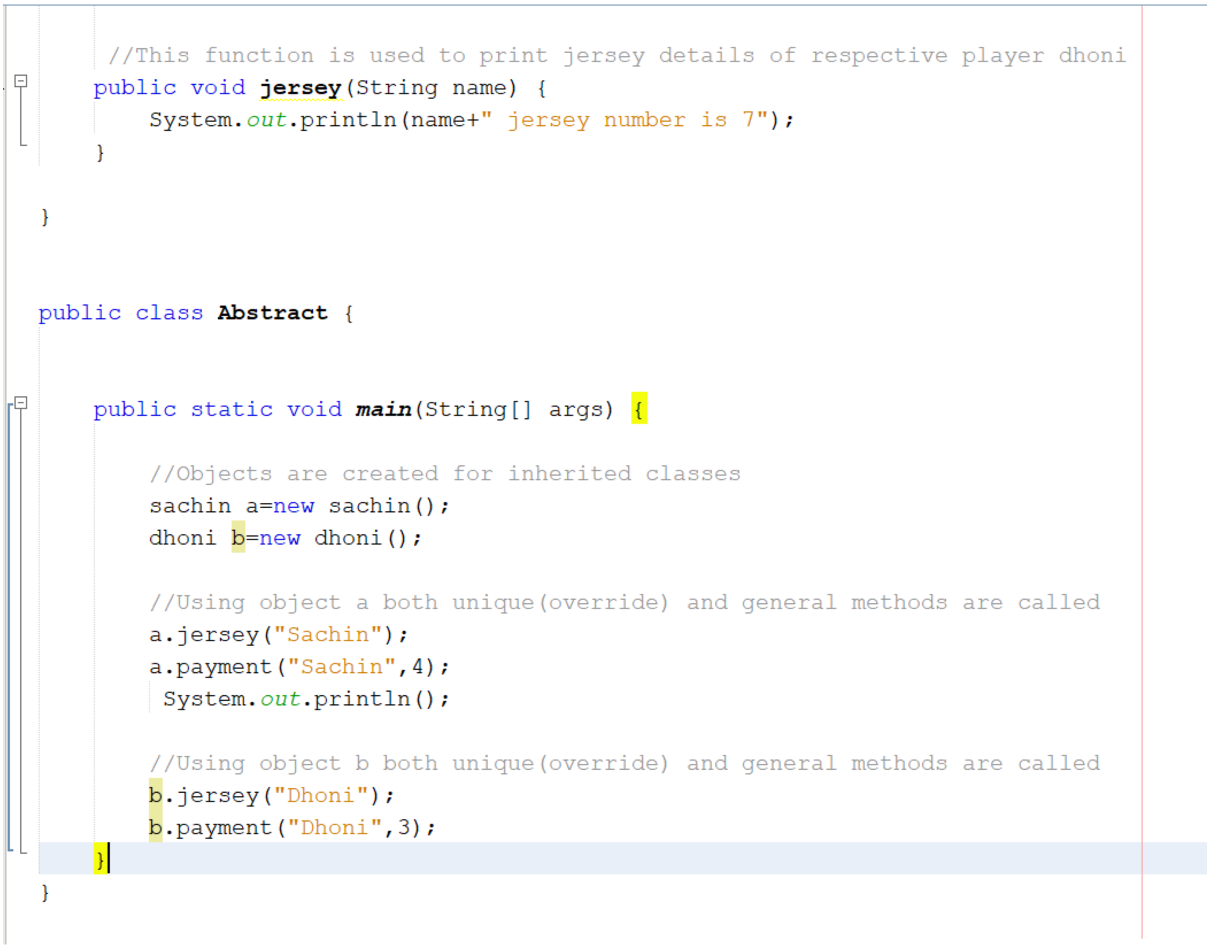
SESSION 8

1. Abstract class

CODE

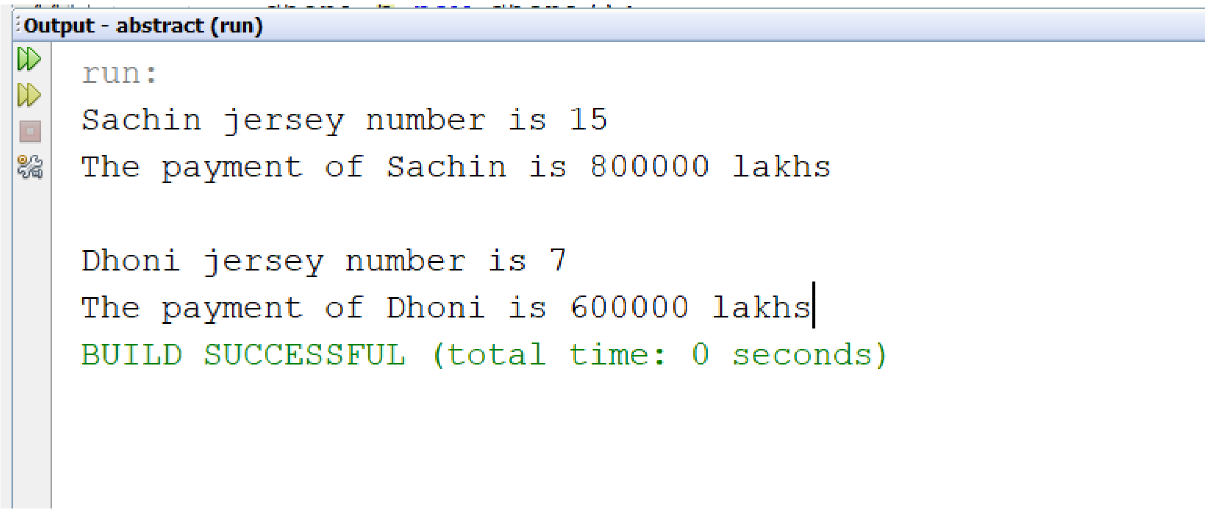


```
package pkgabstract;

//Cricket class is created with general and unique functions
abstract class  cricket{

    //This method is general for all inherited classes and its methods
    public void payment(String name,int match){
      System.out.println("The payment of "+name+" is "+match*200000+" lakhs");
    }

    //This method is unique for all inherited classes and its methods
    public abstract void jersey(String name);

}

//This class is inherited from cricket class
class sachin extends cricket{

    //This function is used to print jersey details of respective player sachin
    public void jersey(String name) {
        System.out.println(name+" jersey number is 15");
    }

}

//This class is inherited from cricket class
class dhoni extends cricket{
```



```
     //This function is used to print jersey details of respective player dhoni
    public void jersey(String name) {
        System.out.println(name+" jersey number is 7");
    }

}


public class Abstract {


    public static void main(String[] args) {

        //Objects are created for inherited classes
        sachin a=new sachin();
        dhoni b=new dhoni();

        //Using object a both unique(override) and general methods are called
        a.jersey("Sachin");
        a.payment("Sachin",4);
         System.out.println();

        //Using object b both unique(override) and general methods are called
        b.jersey("Dhoni");
        b.payment("Dhoni",3);
    }
}
```

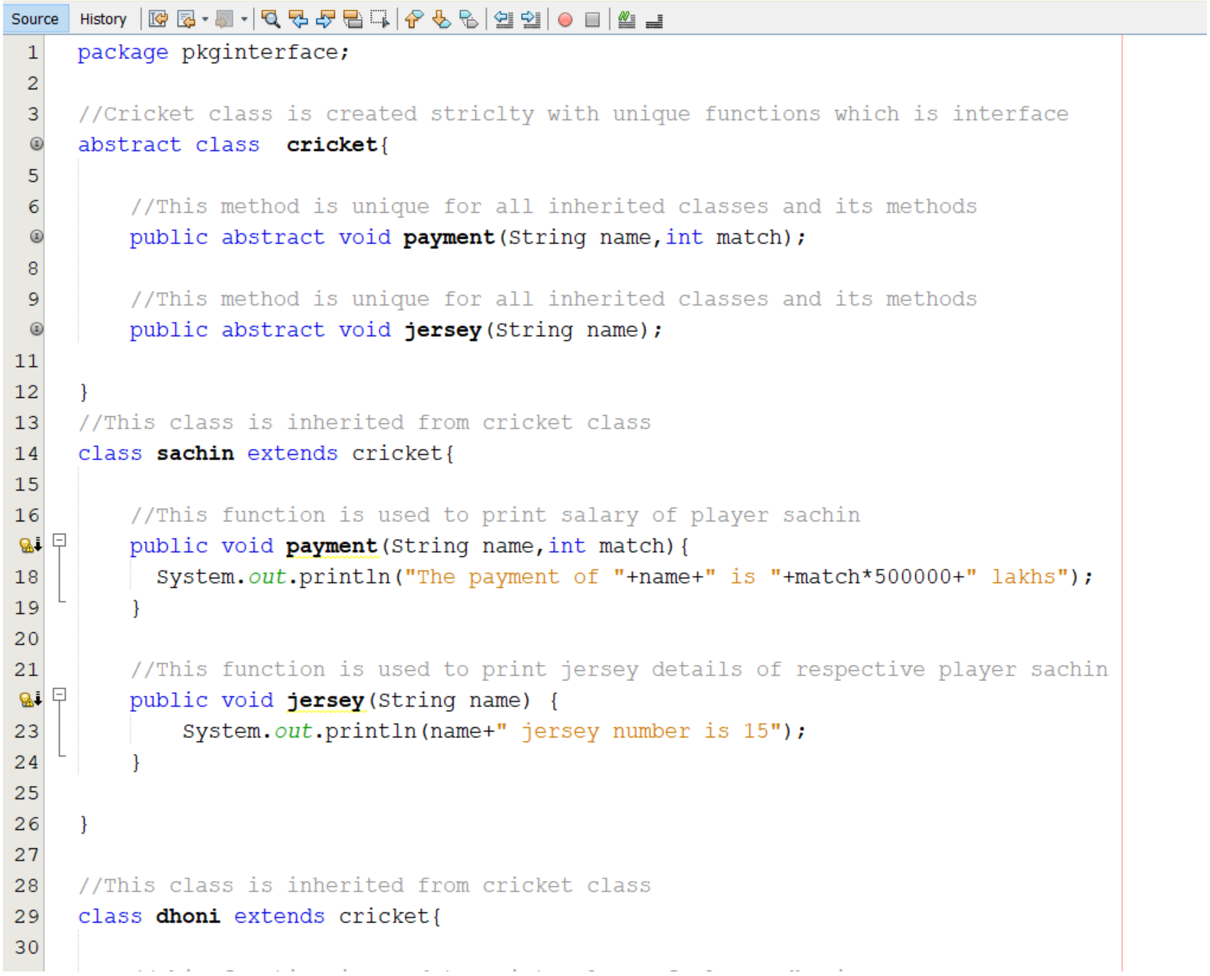
OUTPUT



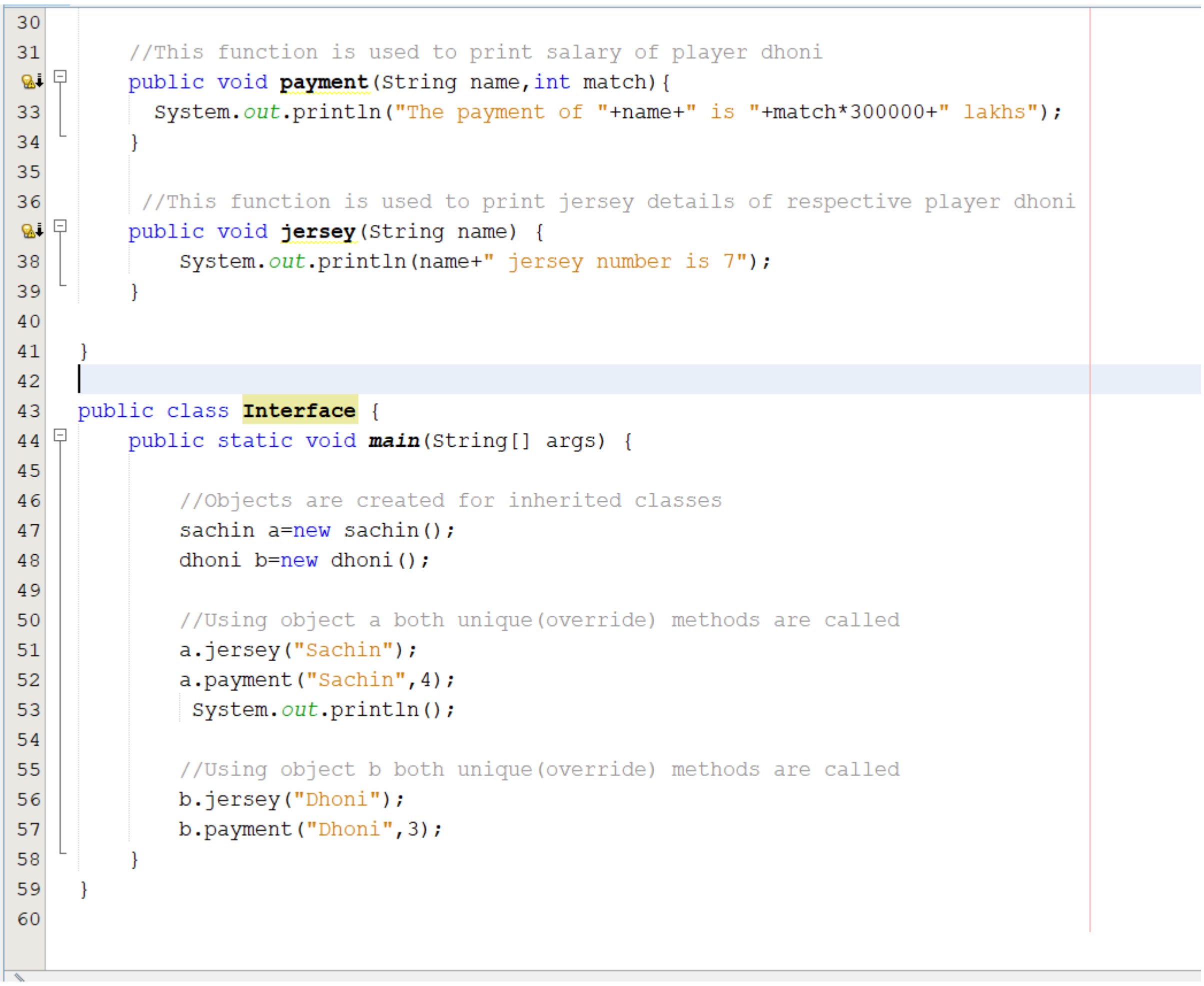
```
run:
Sachin jersey number is 15
The payment of Sachin is 800000 lakhs

Dhoni jersey number is 7
The payment of Dhoni is 600000 lakhs
BUILD SUCCESSFUL (total time: 0 seconds)
```

2. Interface class

CODE

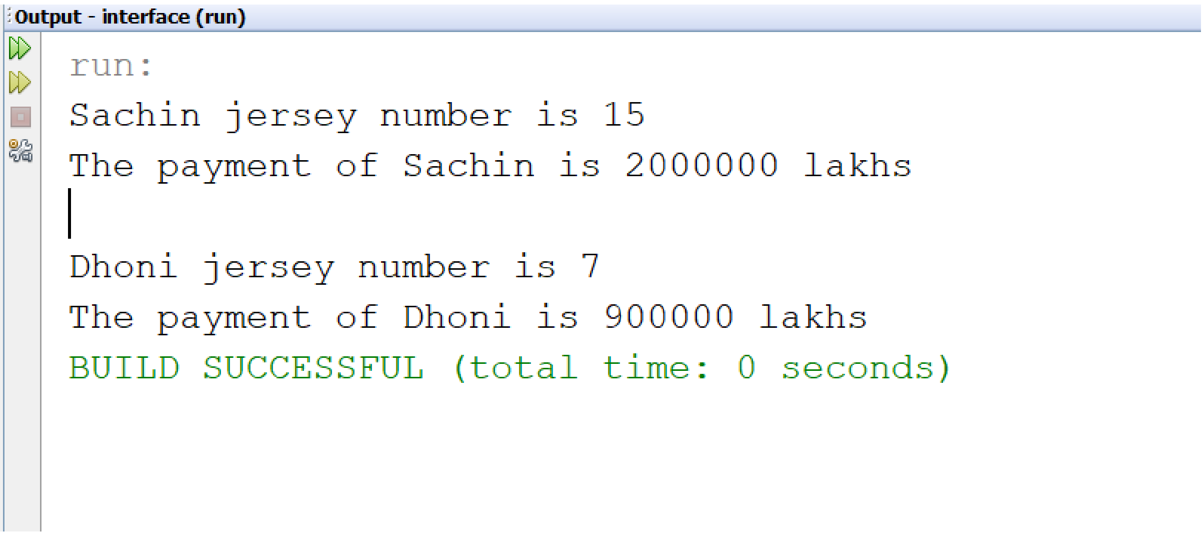


```java
package pkginterface;

//Cricket class is created striclty with unique functions which is interface
abstract class  cricket{

    //This method is unique for all inherited classes and its methods
    public abstract void payment(String name,int match);

    //This method is unique for all inherited classes and its methods
    public abstract void jersey(String name);

}
//This class is inherited from cricket class
class sachin extends cricket{

    //This function is used to print salary of player sachin
    public void payment(String name,int match){
      System.out.println("The payment of "+name+" is "+match*500000+" lakhs");
    }

    //This function is used to print jersey details of respective player sachin
    public void jersey(String name) {
        System.out.println(name+" jersey number is 15");
    }

}

//This class is inherited from cricket class
class dhoni extends cricket{

```



```java

    //This function is used to print salary of player dhoni
    public void payment(String name,int match){
      System.out.println("The payment of "+name+" is "+match*300000+" lakhs");
    }

     //This function is used to print jersey details of respective player dhoni
    public void jersey(String name) {
        System.out.println(name+" jersey number is 7");
    }

}

public class Interface {
    public static void main(String[] args) {

        //Objects are created for inherited classes
        sachin a=new sachin();
        dhoni b=new dhoni();

        //Using object a both unique(override) methods are called
        a.jersey("Sachin");
        a.payment("Sachin",4);
         System.out.println();

        //Using object b both unique(override) methods are called
        b.jersey("Dhoni");
        b.payment("Dhoni",3);
    }
}

```

OUTPUT



```
run:
Sachin jersey number is 15
The payment of Sachin is 2000000 lakhs

Dhoni jersey number is 7
The payment of Dhoni is 900000 lakhs
BUILD SUCCESSFUL (total time: 0 seconds)
```