

JAVA Lab 2

Sunag P(1BM19IS162)

28-04-2021

1).Write a class named Car that has the following data members:

- model. The model field is an int that holds the car's year.
- make. The make field references a String object that holds the make of the car.
- speed. The speed field is an int that holds the car's current speed.

The class should have the following constructor and other methods.

- The constructor should accept the car's year model, make and speed as arguments.
- Accessor methods should get the values stored in an object's year, M
- Accessor methods should get the values stored in an object's year, Model, make, and speed fields.
- Accelerate method should add 5 to the speed field each time it is called.
- Brake method should subtract 5 from the speed field each time it is called.

Demonstrate the class in a program that creates a Car object, and then calls the accelerate method five times. After each call to the accelerate method, get the current speed of the car and display it. Call the brake method five times.

After each call to the brake method, get the current speed of the car and display it.

Code :

```
// import jdk.javadoc.internal.tool.Main;
import java.util.Scanner;
public class car {
    int model;
    String method;
    double speed;

    car(int x,String y,double z){
        model = x;
        method = y;
        speed = z;
    }

    car(){
        model = 2010;
        method = "sedan";
        speed = 50;
    }

    void input(){
        Scanner sc = new Scanner(System.in);
        System.out.println("model of car :");
        model = sc.nextInt();
        System.out.println("Method of car :");
        method = sc.next();
        System.out.println("speed of car in kmph :");
        speed = sc.nextDouble();
    }

    void accelerate(){
        speed += 5;
        System.out.println("Accelerated by 5 kmph");
    }

    void br(){
```

```

        speed -= 5;
        System.out.println("decelerated by 5 kmph");
    }

```

```

    void disp(){
        System.out.println("model of car :"+model);
        System.out.println("Make of car :"+method);
        System.out.println("speed of car :"+speed+" kmph");
    }
    void dispspeed(){
        // System.out.println("model of car :"+model);
        // System.out.println("Method of car :"+method);
        System.out.println("current speed of car :"+speed+" kmph");
    }
}

```

```

class carmain{
    public static void main(String[] args){
        car c = new car(2020,"suv",90.0);
        car c1 = new car();
    }
}

```

```

        c.disp();
        for(int i = 0;i<5;i++)
        {
            c.accelerate();
            c.dispspeed();
        }
        System.out.println("break method is called.....");
        for(int i = 0;i<5;i++)
        {
            c.br();
            c.dispspeed();
        }
        c1.input();
        c1.disp();
    }
}

```

```

int ch = 1;
while(ch!=0){
    System.out.println("1.accelerate\n2.break\n3.display\n0.Exit");
    Scanner sc = new Scanner(System.in);
    ch = sc.nextInt();
    switch(ch){
        case 0: break;
        case 1: c1.accelerate();c1.dispspeed();break;
        case 2 :c1.br();c1.dispspeed();break;
        case 3: c1.disp();
    }
}
}
}

```

Output :

```

model of car :2020
Make of car :suv
speed of car :90.0 kmph

accelerate method is called.....

Accelerated by 5 kmph
current speed of car :95.0 kmph
Accelerated by 5 kmph
current speed of car :100.0 kmph
Accelerated by 5 kmph
current speed of car :105.0 kmph
Accelerated by 5 kmph
current speed of car :110.0 kmph
Accelerated by 5 kmph
current speed of car :115.0 kmph

break method is called.....

decelerated by 5 kmph
current speed of car :110.0 kmph
decelerated by 5 kmph
current speed of car :105.0 kmph
decelerated by 5 kmph
current speed of car :100.0 kmph
decelerated by 5 kmph
current speed of car :95.0 kmph
decelerated by 5 kmph
current speed of car :90.0 kmph
model of car :

```

2) WRITE A PROGRAM TO DEMONSTRATE OVERLOADED METHODS

Code:

```

public class methodOverloading {
    void area(int x){
        System.out.println("area of square is "+x*x + " sq units");
    }
    void area(int x,int y){
        System.out.println("area of rectangle is "+x*y + " sq units");
    }
}

class overMain{
    public static void main(String[] args){
        methodOverloading o = new methodOverloading();
        methodOverloading o1 = new methodOverloading();
        o.area(5);
        o1.area(4,5);
    }
}

```

Output:

```
s\launcher.bat' 'C:\Program Files\Java\jdk-16\bin\j
ile.encoding=UTF-8' '-cp' 'C:\Users\Asus\AppData\Ro
ca8ec5d0b9\redhat.java\jdt_ws\GIT_d4ab13ff\bin' 'ov
area of square is 25 sq units
area of rectangle is 20 sq units
PS C:\Users\Asus\Desktop\GIT> █
```

3) WRITE A PROGRAM TO DEMONSTRATE OVERLOADED CONSTRUCTORS

Code:

```
public class constructorOverloading {
    int x,y;
    constructorOverloading(){
        x = 1;
        y = 1;
    }
    constructorOverloading(int l){
        x = y = l;
    }
    constructorOverloading(int l,int w){
        x = l;
        y = w;
    }
    void area(){
        System.out.println("area is "+x*y + " sq units");
    }
}

class conMain{
    public static void main(String[] args){
        constructorOverloading o = new constructorOverloading();
        constructorOverloading o1 = new constructorOverloading(5);
        constructorOverloading o2 = new constructorOverloading(5,4);
        o.area();
        o1.area();
        o2.area();
    }
}
```

```
}
```

Output:

```
PS C:\Users\Asus\Desktop\GIT> & 'c:\Users\Asus\.vscode\extensions\launcher.bat' 'C:\Program Files\Java\jdk-16\bin\java.exe'
file.encoding=UTF-8' '-cp' 'C:\Users\Asus\AppData\Roaming\Coc
ca8ec5d0b9\redhat.java\jdt_ws\GIT_d4ab13ff\bin' 'conMain'
area is 1 sq units
area is 25 sq units
area is 20 sq units
PS C:\Users\Asus\Desktop\GIT> █
```

- 4) Program to demonstrate passing of objects to a method and returning of objects from a method.

Code:

```
public class objParaReturn {
    int x , y;
    objParaReturn(int a,int b){
        x = a;
        y = b;
    }
    objParaReturn(){
        x = 0;
        y = 0;
    }
    objParaReturn isequal(objParaReturn o){
        if (o.x == x && o.y==y)
            return this;
        else return o;
    }
    void disp(){
        System.out.println("x = "+x+" y = "+y);
    }
}

class demo1{
    public static void main(String[] args) {
```

```
        objParaReturn o1 = new objParaReturn(1, 2) ;
        objParaReturn o2 = new objParaReturn(1, 2) ;
        objParaReturn o3 = new objParaReturn() ;
```

```
        o3 = o1.isequal(o2);
```

```
o3.disp();
```

```
objParaReturn o4 = new objParaReturn(1, 5) ;  
objParaReturn o5 = new objParaReturn() ;  
o5 = o1.isequal(o4);  
o5.disp();  
}}
```

Output:

```
PS C:\Users\Asus\Desktop\GIT> & 'c:\Users\Asus\Asus\launcher.bat' 'C:\Program Files\Java\jdk-16\bin\java.exe' '-file.encoding=UTF-8' '-cp' 'C:\Users\Asus\AppData\Local\Temp\ca8ec5d0b9\redhat.java\jdt_ws\GIT_d4ab13ff\bin' x = 1 y = 2  
x = 1 y = 5  
PS C:\Users\Asus\Desktop\GIT> █
```

5) Read input from keyboard and display the same for name age and salary of a person

Code:

```
import java.util.Scanner;  
  
public class basicInput {  
    public static void main(String[] args) {  
        Scanner sc = new Scanner(System.in);  
        System.out.println("Enter name :");  
        String name = sc.next();  
        System.out.println("Enter age :");  
        int age = sc.nextInt();
```

```
        System.out.println("Enter salary :");
```

```
        double sal = sc.nextDouble();
```

```
        System.out.println("name :"+name);  
        System.out.println("age :"+age);  
        System.out.println("salary :"+sal);
```

```
        sc.close();
```

```
}  
}
```

Output:

```
PS C:\Users\Asus\Desktop\GIT> java -cp C:\Program Files\Java\jdk-10\bin\java.exe  
-Dfile.encoding=UTF-8' '-cp' 'C:\Users\Asus\AppData\Roaming\  
ca8ec5d0b9\redhat.java\jdt_ws\GIT_d4ab13ff\bin' 'basicInp  
Enter name :  
sun  
Enter age :  
18  
Enter salary :  
52000  
name :sun  
age :18  
salary :52000.0  
PS C:\Users\Asus\Desktop\GIT>
```