

Getters, Setters & Method Overloading in Java Example Tasks

Task1:

Make a class named Student. Declare a field named name of type String in Student class. Make it Private field. Define two methods getName() and setName() in class Student. The return_type of getName is String and Access modifier is public. The return type of setName is Void and Access modifier is public. In main class make an object of Student class. Using method setName(), set any name in name field, and using getName method print the name. Then using object **(don't use setName() method)**, set the name field with any name, and print the name using getName method. Show output. If you get an error then tell its reason explicitly.

Solution:

Here I made two classes in two separate files so as to keep the code simple and keep the main function separated from the remaining code.

Code:

```
public class Student {
    private String name;
    public String getName()
    {
        return name;
    }
    void setName(String name)
    {
        this.name=name;
    }
}

import java.util.Scanner;
public class StudentMain extends Student{
    public static void main(String abc[])
    {
        Student st1 = new Student();
        Scanner input = new Scanner (System.in);
        System.out.println("Enter the name of the student");
        String name = input.nextLine();
        //st1.name="Hania Arif";
        //System.out.println("The name of the student is: "+ st1.getName());
        st1.setName(name);
        System.out.println("The name of the student is: "+ st1.getName());
    }
}
```

The first one is the Student class while the second one is the main class.

- ✓ The program takes the name of the student from the user.
- ✓ Sets the student name by setName method
- ✓ Displays the student name to the user by the getName method.

As, Name of the student is a private member of the Student class so we cannot access it directly via dot operator. So if we use the dot operator for setting the name of the student, it gives an error.

- *Firstly, I tried not to use the setName operator as said in the task.*
- *But as we can't access a private member of class by the dot operator so the program gave an error.*

Output:

Enter the name of the student

Hania Arif

The name of the student is: Hania Arif

Task2:

Make a class Program1 with four fields x=5, y=5, z=0 and result=0 of type double. In this class defines a public method named calculate which calculates

- 1. Sum //No parameter method (sum of x and y only)**
- 2. Division of two numbers //two parameters of type double**
- 3. Multiplication of three numbers //three parameters of type int**
- 4. Subtraction of two numbers //two parameters of type int**
- 5. Square of a number //one parameter of type double**

All methods have return type void except one which calculates square. It should return square of a number so it's return type would be double. All methods have public accessibility.

In main class make an object and call all methods with different arguments to show method overloading.

Solution:

Here I made two classes in two separate files so as to keep the code simple and keep the main function separated from the remaining code.

Code:

```
public class Program1 {  
    double x=5;  
    double y=5;  
    double z=0;  
    double result=0;  
    public void calculate()  
    {  
        double sum = x+y;  
        System.out.println("the sum of x and y is: " + sum);  
    }  
    public void calculate(double a, double b)  
    {
```

```
        double division = a/b;
        System.out.println("the division of the two numbers is: " + division);
    }
    public void calculate(int a, int b, int c)
    {
        int multiply=a*b*c;
        System.out.println("the multiplication of the three numbers is: " +
multiply);
    }
    public void calculate(int a, int b)
    {
        int subtract=a-b;
        System.out.println("The subtraction of two numbers is: "+ subtract);
    }
    public double calculate(double a)
    {
        return (a*a);
    }
}

import java.util.Scanner;
public class program1main extends Program1{
    public static void main(String abc[])
    {int choice =1;
the_while:
        do
        {
            System.out.println("what do you want to do?");
            System.out.println("1-Add\n2-Divide\n3-Multiply\n4-Subtract\n5-Square\n6-
Exit");
            Scanner input = new Scanner(System.in);
            choice = input.nextInt();
            Program1 p1 = new Program1();
            switch(choice)
            {
                case 1:
                {
                    p1.calculate();
                }
                break;
                case 2:
                {
                    p1.calculate(p1.y, p1.z);
                }
                break;
                case 3:
                {
                    System.out.println("Enter three numbers");
                    int a= input.nextInt();
                    int b=input.nextInt();
                    int c=input.nextInt();
                    p1.calculate(a, b, c);
                }
                break;
                case 4:
                {
                    System.out.println("Enter two numbers");
                    int a= input.nextInt();
                    int b=input.nextInt();
                    p1.calculate(a, b);
                }
            }
        }
    }
}
```

```

    }
    break;
    case 5:
    {
        p1.result=p1.calculate(p1.x);
        System.out.println("Square of x is: "+ p1.result);
    }
    break;
    case 6:
    {
        System.out.println("Exited");
        break the_while;
    }
    break;

    default:
    {
        System.out.println("Exited");
        break the_while;
    }
}
}while(choice != 0);
}

```

The first one in the Program1 class while the second one is the main class.

- ✓ The program asks the user that which function he wants to perform.
- ✓ Program performs the calculations on the stored numbers on the number stored in x, y, z or asks the user to enter integers (as required by the functions and given in the task) and then performs the calculations on the numbers and shows the result to the user.
- ✓ *If the user enters an option other than the given option or number 6 the program exits and gets terminated else the program keeps on asking the user that what function he wants to perform next.*

Output:

what do you want to do?

1-Add
2-Divide
3-Multiply
4-Subtract
5-Square
6-Exit

1

the sum of x and y is: 10.0

what do you want to do?

1-Add
2-Divide
3-Multiply
4-Subtract
5-Square
6-Exit

2

the division of the two numbers is: Infinity

```
what do you want to do?
1-Add
2-Divide
3-Multiply
4-Subtract
5-Square
6-Exit
3
Enter three numbers
1
2
3
the multiplication of the three numbers is: 6
what do you want to do?
1-Add
2-Divide
3-Multiply
4-Subtract
5-Square
6-Exit
4
Enter two numbers
5
8
The subtraction of two numbers is: -3
what do you want to do?
1-Add
2-Divide
3-Multiply
4-Subtract
5-Square
6-Exit
5
Square of x is: 25.0
what do you want to do?
1-Add
2-Divide
3-Multiply
4-Subtract
5-Square
6-Exit
9
Exited
```

(In the end as I entered a number other than the options the program terminated by getting out of the loop else it kept asking the user that which function he wants to perform next?)