
Practical - 01

=====

Student Name : Vinni Fengade
Roll No. : 67
Sem & Sec : 4 CSE [B]
Course Name : Object Oriented Programming (CSP256)
Date Compiled : 18-April-2022

=====

Problem Statements:

1. Write a program to demonstrate method overloading.

Write a program to create a class OnlineShoppingPortal. Customers are of two categories, Prime and Regular. "Prime" requires only amount to be paid for checkout and "Regular" customer has to provide amount and promocode. A prime customer gets 30% off if cart value > 1000, else 20% off. A regular customer gets only Rs.10 off if cart value < 500, else 10% off (provided that the code matches "GET10"). Demonstrate function overloading on a function checkout() which will display the original and discounted amount.

Create object in main() to demonstrate function overloading.

Code

File : Project1.java

```
package com.mycompany.project1;

public class Project1 {
    public static void main(String[] args) {
        shopping PrimeCustomer = new shopping();
        PrimeCustomer.checkout(10000);

        PrimeCustomer = new shopping();
        PrimeCustomer.checkout(500);
    }
}
```

```
        shopping NonPrimeCustomer = new shopping();
        NonPrimeCustomer.checkout(1000,"GET10");

        NonPrimeCustomer = new shopping();
        NonPrimeCustomer.checkout(1000,"10GET");
    }
}
```

File :OnlineShoppingClass

```
public class OnlineShoppingClass {
    int discount = 0;
    void checkout(int amount){
        if(amount >1000) discount = (int) (0.3*amount);// 30% discount
        else discount=(int)(0.2*amount);
        System.out.println("Total Amount is Rs."+amount);
        System.out.println("Customer received discount of Rs."+discount);
        System.out.println("Final amount to be paid is Rs."+(amount-discount));
    }
    void checkout(int amount,String Promocode){
        if("GET10".equals(Promocode) ){
            if(amount>=500){
                this.discount = (int) (0.1*amount);
            }else{
                this.discount = 10;
            }
        }else discount=0;
        System.out.println("Total Amount is Rs."+amount);
        System.out.println("Customer received discount of Rs."+discount);
        System.out.println("Thus,Final amount to be paid is Rs."+(amount-
                                                                    discount)+"\n");
    }
}
```

Execution

-----< com.mycompany.practical1q1:Practical1q1 >-----

Building Practical1q1 1.0-SNAPSHOT

-----[jar]-----

exec-maven-plugin:3.0.0:exec (default-cli) @ Practical1q1 ---

Total Amount is Rs.10000

Customer received discount of Rs.3000

Final amount to be paid is Rs.7000

Total Amount is Rs.700

Customer received discount of Rs.140

Final amount to be paid is Rs.560

Total Amount is Rs.1000

Customer received discount of Rs.100

Thus,Final amount to be paid is Rs.900

Total Amount is Rs.1000

Customer received discount of Rs.0

Thus,Final amount to be paid is Rs.1000

BUILD SUCCESS

Total time: 0.766 s

Finished at: 2022-05-07T19:43:16+05:30

2. Create a class complex and provide functionalities to add and multiply 2 complex numbers.

Class Complex will have real and imaginary as the data members. Write appropriate main() to demonstrate the functionalities.

Code

File :Practical1Q2.java

```
package com.mycompany.practical1q2;

public class Practical1Q2 {
    public static void main(String[] args) {
        Complex num1 = new Complex(2, 3);
        Complex num2 = new Complex(3, 4);
        Complex num3 = new Complex();
        num3.add(num1, num2);
        Complex num4 = new Complex();
        num4.mul(num1, num2);
    }
}
```

File :Complex.java

```
package com.mycompany.practical1q2;

public class Complex {
    int Img;
    int real;
    int ar,ai;
    int mr,mi;

    public Complex() {
    }

    public Complex(int real,int Img){
        this.Img = Img;
        this.real = real;
    }
}
```

```

void add(Complex a,Complex b){
    ar = a.real+b.real;
    ai = a.Img + b.Img;
    System.out.println("Addition of "+a.real +"+"+ a.Img +"i and "+ b.real
                        +"+"+ b.Img +"i  is "+this.ar+"+"+this.ai+"i");
}

void mul(Complex a,Complex b){
    mr =(a.real)*(b.real) - (a.Img)*(b.Img);
    mi =(a.Img)*(b.real) + (a.real)*(b.Img);
    System.out.println("Multiplication of "+a.real +"+"+ a.Img +"i and "+
                        b.real +"+"+ b.Img +"i  is "+this.mr+"+"+this.mi+"i");
}
}

```

Execution

```

-----< com.mycompany.practical1q2:Practical1Q2 >-----
Building Practical1Q2 1.0-SNAPSHOT
-----[ jar ]-----

--- exec-maven-plugin:3.0.0:exec (default-cli) @ Practical1Q2 ---
Addition of 4+3i and 3+4i  is 7+7i
Multiplication of 4+3i and 3+4i  is 0+25i

-----

BUILD SUCCESS

-----

Total time:  0.976 s
Finished at: 2022-05-07T19:00:38+05:30
-----

```