Divyanshu Tyagi
207919
MC20110

OOP LAB

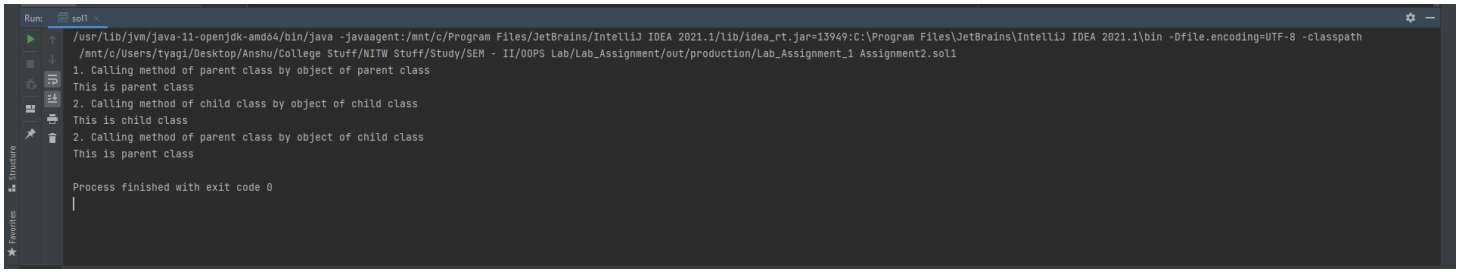
Assignment – 02

Q1. Create a class with a method that prints "This is parent class" and its subclass with another method that prints "This is child class". Now, create an object for each of the class and call

1 - method of parent class by object of parent class
2 - method of child class by object of child class
3 - method of parent class by object of child class
Solu.

```java
/*
 * Copyright (c) 2021.
 * Divyanshu Tyagi
 * NIT Warangal
 * 207919
 */

package Assignment2;
class Parent {
    public void printParent() {
        System.out.println("This is parent class");
    }
}
class Child extends Parent {
    public void printChild() {
        System.out.println("This is child class");
    }
}
public class sol1 {
    public static void main(String[] args) {
        Parent p1 = new Parent();
        Child c1 = new Child();
        System.out.println("1. Calling method of parent class by object of parent class");
        p1.printParent();
        System.out.println("2. Calling method of child class by object of child class");
        c1.printChild();
        System.out.println("2. Calling method of parent class by object of child class");
        c1.printParent();
    }
}
```

OUTPUT:

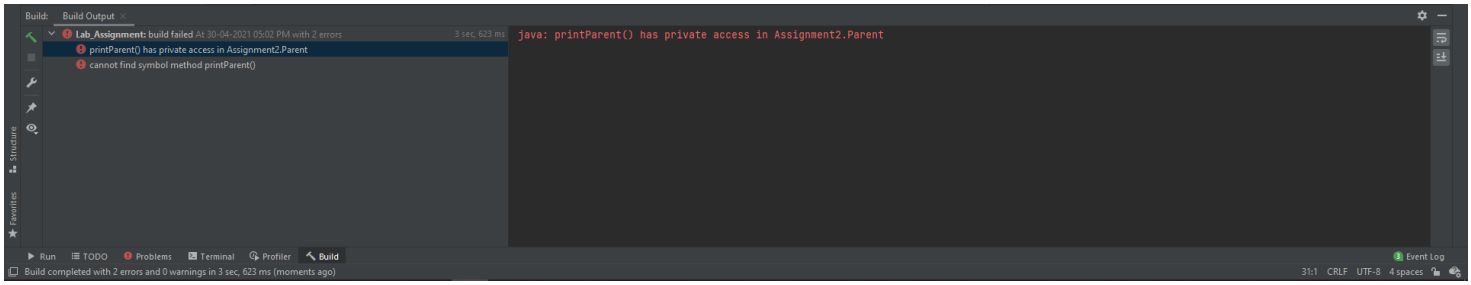


Q2. In the above example, declare the method of the parent class as private and then repeat the first two operations (You will get error in the third).

Solu.

```java
/*
 * Copyright (c) 2021.
 * Divyanshu Tyagi
 * NIT Warangal
 * 207919
 */

package Assignment2;
class Parent {
    private void printParent() {
        System.out.println("This is parent class");
    }
}
class Child extends Assignment2.Parent {
    public void printChild() {
        System.out.println("This is child class");
    }
}
public class sol2 {
    public static void main(String[] args) {
        Parent p1 = new Parent();
        Child c1 = new Child();
        System.out.println("1. Calling method of parent class by object of parent class");
        p1.printParent();
        System.out.println("2. Calling method of child class by object of child class");
        c1.printChild();
        System.out.println("2. Calling method of parent class by object of child class");
        c1.printParent();
    }
}
```

OUTPUT:



Q3. Create a class named 'Member' having the following members: Data members

1 - Name
2 - Age
3 - Phone number
4 - Address
5 - Salary

It also has a method named 'printSalary' which prints the salary of the members. Two classes 'Employee' and 'Manager' inherits the 'Member' class. The 'Employee' and 'Manager' classes have data members 'specialization' and 'department' respectively. Now, assign name, age, phone number, address and salary to an employee and a manager by making an

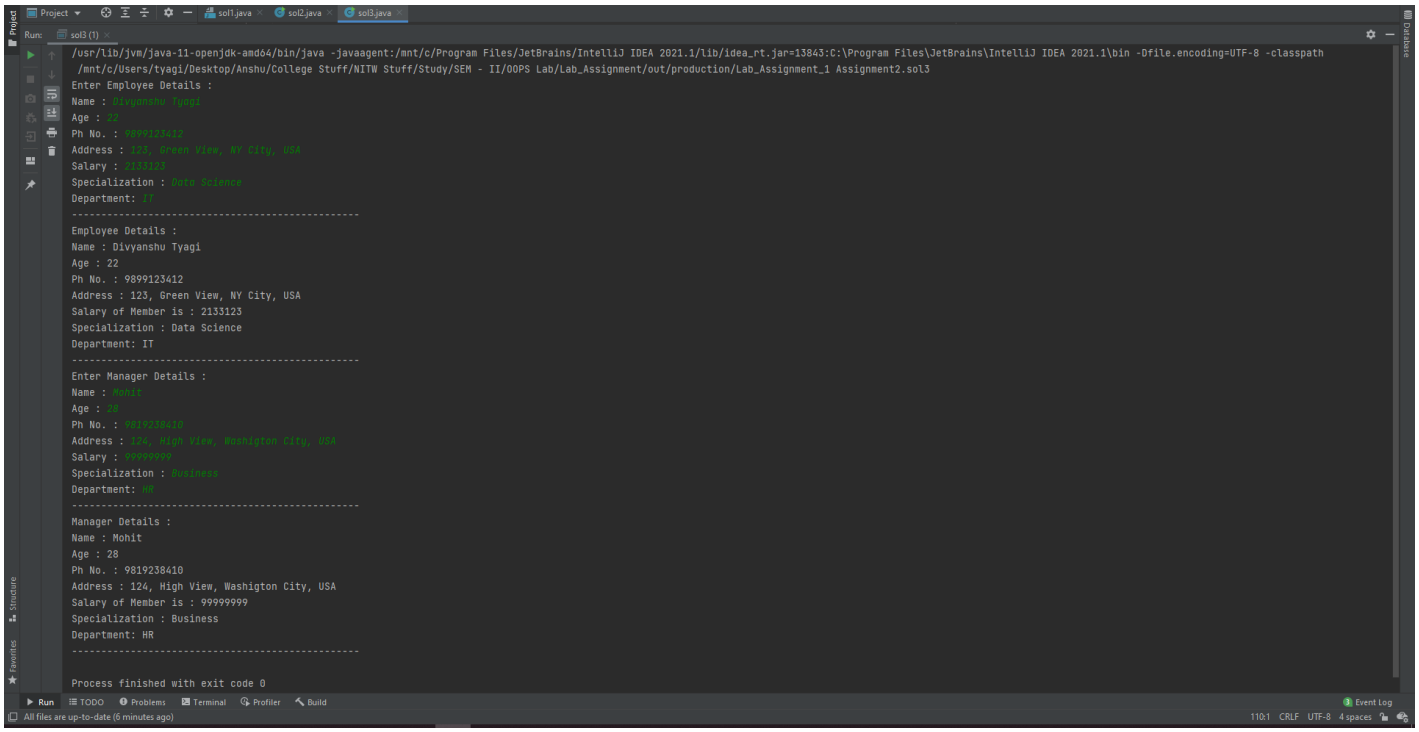
object of both of these classes and print the same.

Solu.

```java
/*
 * Copyright (c) 2021.
 * Divyanshu Tyagi
 * NIT Warangal
 * 207919
 */

package Assignment2;

import javax.crypto.spec.PSource;
import java.util.Scanner;

/*
Create a class named 'Member' having the following members: Data members
    1 - Name
    2 - Age
    3 - Phone number
    4 - Address
    5 - Salary
It also has a method named 'printSalary' which prints the salary of the members. Two classes
'Employee' and 'Manager' inherits the 'Member' class. The 'Employee' and 'Manager' classes have data
members 'specialization' and 'department' respectively. Now, assign name, age, phone number, address
and salary to an employee and a manager by making an object of both of these classes and print the
same.
*/
class Member {
    public String Name;
    public Integer Age;
    public String PhoneNumber;
    public String Address;
    public Integer Salary;

    public void printSalary() {
        System.out.println("Salary of Member is : " + Salary);
    }
}

class Employee extends Member {
    private String specialization, department;
    public void printEmployeeDetails() {
        System.out.println("-------------------------------------------------");
        System.out.println("Employee Details : ");
        System.out.println("Name : " + this.Name);
        System.out.println("Age : " + this.Age);
        System.out.println("Ph No. : " + this.PhoneNumber);
        System.out.println("Address : " + this.Address);
        this.printSalary();
        System.out.println("Specialization : " + this.specialization);
        System.out.println("Department: " + this.department);
        System.out.println("-------------------------------------------------");
    }

    public void input() {
        Scanner sc = new Scanner(System.in);
        System.out.print("Name : " );
        this.Name = sc.nextLine();
        System.out.print("Age : " );
        this.Age = Integer.parseInt(sc.nextLine());
        System.out.print("Ph No. : " );
        this.PhoneNumber = sc.nextLine();
        System.out.print("Address : ");
        this.Address = sc.nextLine();
```

```
        System.out.print("Salary : ");
        this.Salary = Integer.parseInt(sc.nextLine());
        System.out.print("Specialization : ");
        this.specialization = sc.nextLine();
        System.out.print("Department: ");
        this.department = sc.nextLine();
    }
}
class Manager extends Member {
    private String specialization, department;
    public void input() {
        Scanner sc = new Scanner(System.in);
        System.out.print("Name : " );
        this.Name = sc.nextLine();
        System.out.print("Age : " );
        this.Age = Integer.parseInt(sc.nextLine());
        System.out.print("Ph No. : " );
        this.PhoneNumber = sc.nextLine();
        System.out.print("Address : ");
        this.Address = sc.nextLine();
        System.out.print("Salary : ");
        this.Salary = Integer.parseInt(sc.nextLine());
        System.out.print("Specialization : ");
        this.specialization = sc.nextLine();
        System.out.print("Department: ");
        this.department = sc.nextLine();
    }
    public void printManagerDetails() {
        System.out.println("-------------------------------------------------");
        System.out.println("Manager Details : ");
        System.out.println("Name : " + this.Name);
        System.out.println("Age : " + this.Age);
        System.out.println("Ph No. : " + this.PhoneNumber);
        System.out.println("Address : " + this.Address);
        this.printSalary();
        System.out.println("Specialization : " + this.specialization);
        System.out.println("Department: " + this.department);
        System.out.println("-------------------------------------------------");
    }
}
public class sol3 {


    public static void main(String[] args) {
        System.out.println("Enter Employee Details : ");
        Employee emp = new Employee();
        emp.input();
        emp.printEmployeeDetails();

        System.out.println("Enter Manager Details : ");
        Manager mag = new Manager();
        mag.input();
        mag.printManagerDetails();
    }
}
```

OUTPUT:



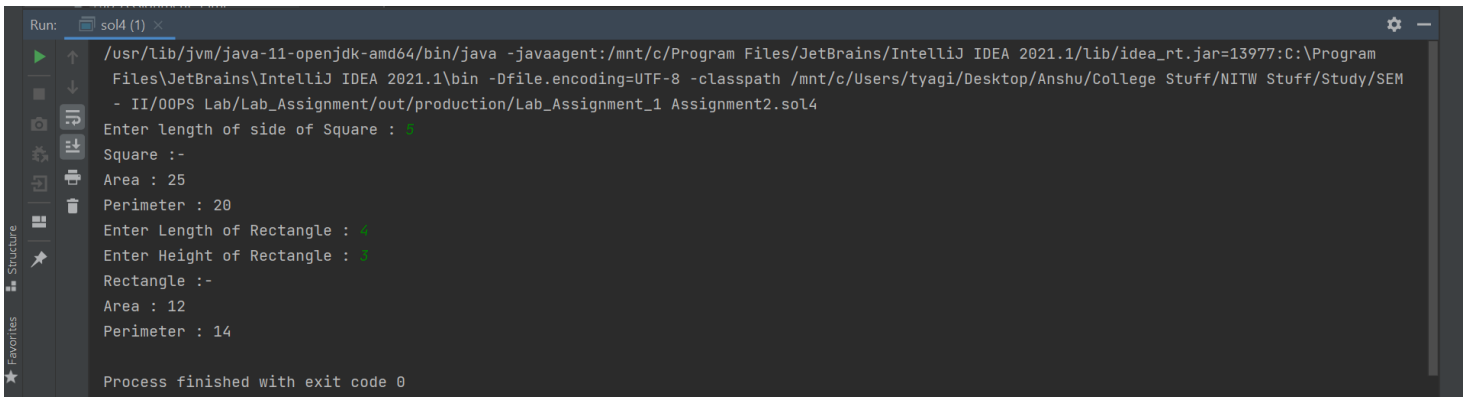
Q4. Create a class named 'Rectangle' with two data members 'length' and 'breadth' and two methods to print the area and perimeter of the rectangle respectively. Its constructor having parameters for length and breadth is used to initialize length and breadth of the rectangle. Let class 'Square' inherit the 'Rectangle' class with its constructor having a parameter for its side (suppose s) calling the constructor of its parent class as 'super (s, s)'. Print the area and perimeter of a rectangle and a square.

Solu.

```java
/*
 * Copyright (c) 2021.
 * Divyanshu Tyagi
 * NIT Warangal
 * 207919
 */

package Assignment2;
/*
Create a class named 'Rectangle' with two data members 'length' and 'breadth' and two methods to
print the area and perimeter of the rectangle respectively. Its constructor having parameters for
length and breadth is used to initialize length and breadth of the rectangle. Let class 'Square'
inherit the 'Rectangle' class with its constructor having a parameter for its side (suppose s)
calling the constructor of its parent class as 'super(s,s)'. Print the area and perimeter of a
rectangle and a square.
*/

import java.util.Scanner;

class Rectangle {
    Rectangle(int length, int breadth) {
        this.length = length;
        this.breadth = breadth;
```

```
    }

    private int length, breadth;
    public void printArea() {
        System.out.println("Area : " + length*breadth);
    }
    public void printPerimeter() {
        System.out.println("Perimeter : " + 2*(length+breadth));
    }
}
class Square extends Rectangle {
    Square(int side) {
        super(side,side);
    }
}
public class sol4 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter length of side of Square : ");
        int side = sc.nextInt();

        Square sq = new Square(side);
        System.out.println("Square :- ");
        sq.printArea();
        sq.printPerimeter();

        System.out.print("Enter Length of Rectangle : ");
        int length = sc.nextInt();
        System.out.print("Enter Height of Rectangle : ");
        int height = sc.nextInt();

        Rectangle rec = new Rectangle(length,height);
        System.out.println("Rectangle :- ");
        rec.printArea();
        rec.printPerimeter();
    }
}
```

OUTPUT :

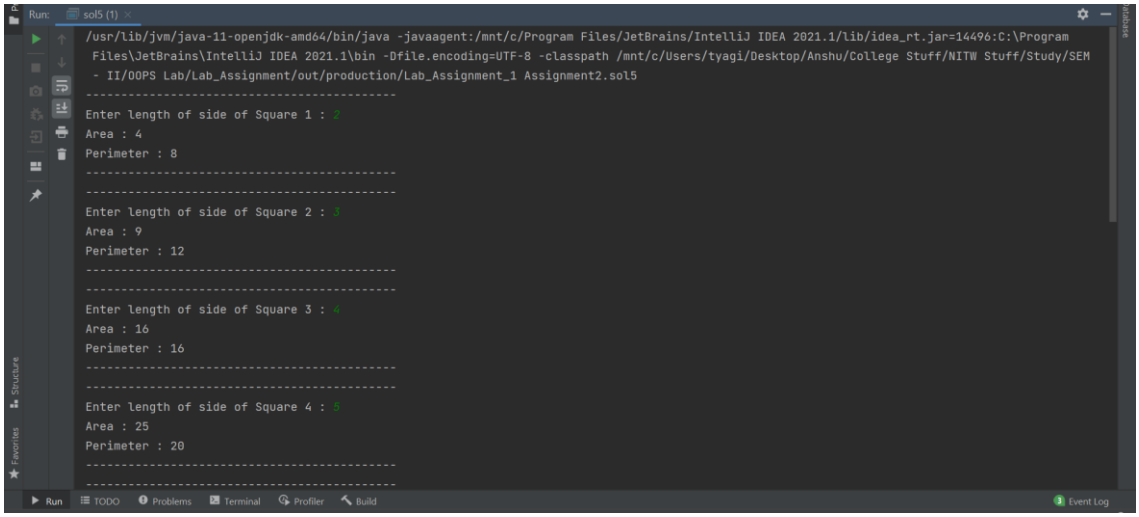


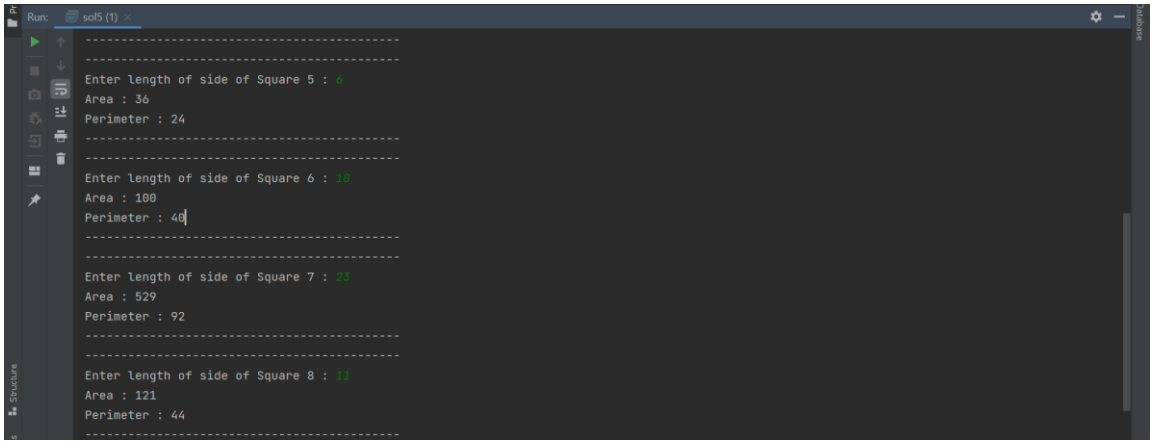
Q5. Now repeat the above example to print the area of 10 squares. Hint-Use array of objects.

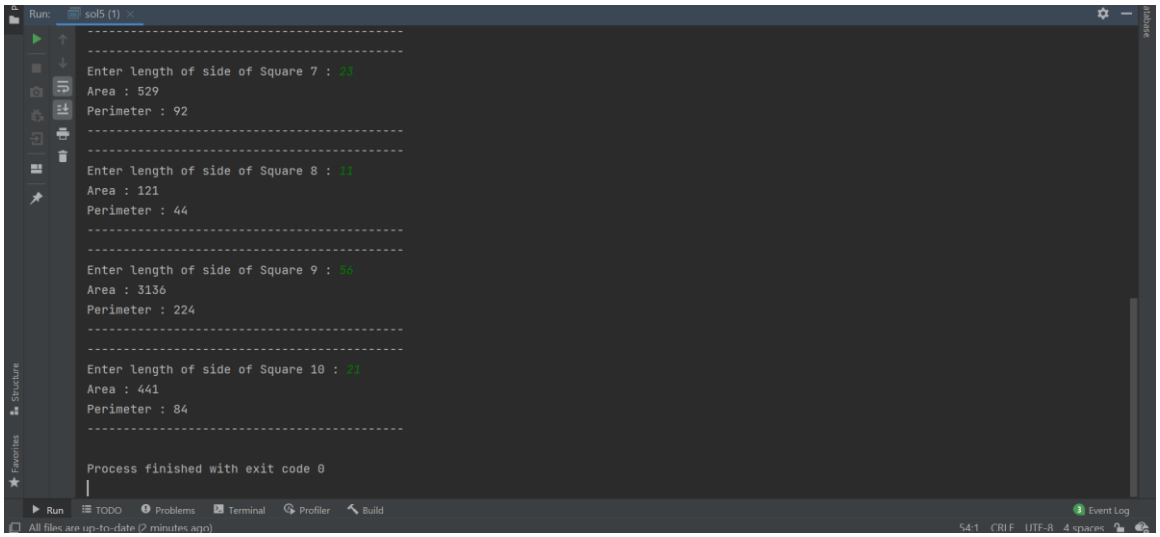
Solu.

```java
/*
 * Copyright (c) 2021.
 * Divyanshu Tyagi
 * NIT Warangal
 * 207919
 */

package Assignment2;

import java.util.Scanner;

class Rectangle {
    Rectangle(int length, int breadth) {
        this.length = length;
        this.breadth = breadth;
    }

    private int length, breadth;
    public void printArea() {
        System.out.println("Area : " + length*breadth);
    }
    public void printPerimeter() {
        System.out.println("Perimeter : " + 2*(length+breadth));
    }
}
class Square extends Rectangle {
    Square(int side) {
        super(side,side);
    }
}
public class sol5 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        Square[] sq = new Square[10];
        for (int i = 0; i < 10; i++) {
            System.out.println("-----------------------------------------");
            System.out.print("Enter length of side of Square " + (i+1) + " : ");
            int side = sc.nextInt();
            sq[i] = new Square(side);
            sq[i].printArea();
            sq[i].printPerimeter();
            System.out.println("-----------------------------------------");
        }
    }
}
```

OUTPUT :


Run: sol5 (1)
/usr/lib/jvm/java-11-openjdk-amd64/bin/java -javaagent:/mnt/c/Program Files/JetBrains/IntelliJ IDEA 2021.1/lib/idea_rt.jar=14496:C:\Program Files\JetBrains\IntelliJ IDEA 2021.1\bin -Dfile.encoding=UTF-8 -classpath /mnt/c/Users/tyagi/Desktop/Anshu/College Stuff/NITW Stuff/Study/SEM - II/OOPS Lab/Lab_Assignment/out/production/Lab_Assignment_1 Assignment2.sol5
Enter length of side of Square 1 : 2
Area : 4
Perimeter : 8
Enter length of side of Square 2 : 3
Area : 9
Perimeter : 12
Enter length of side of Square 3 : 4
Area : 16
Perimeter : 16
Enter length of side of Square 4 : 5
Area : 25
Perimeter : 20
Run TODO Problems Terminal Profiler Build Event Log


Run: sol5 (1)
Enter length of side of Square 5 : 6
Area : 36
Perimeter : 24
Enter length of side of Square 6 : 10
Area : 100
Perimeter : 40
Enter length of side of Square 7 : 23
Area : 529
Perimeter : 92
Enter length of side of Square 8 : 11
Area : 121
Perimeter : 44


Run: sol5 (1)
Enter length of side of Square 7 : 23
Area : 529
Perimeter : 92
Enter length of side of Square 8 : 11
Area : 121
Perimeter : 44
Enter length of side of Square 9 : 56
Area : 3136
Perimeter : 224
Enter length of side of Square 10 : 21
Area : 441
Perimeter : 84
Process finished with exit code 0
Run TODO Problems Terminal Profiler Build Event Log
All files are up-to-date (2 minutes ago)

Q6. Write a program in Java to implement a calculator having four functions such addition, multiplication, division, and subtraction, where the four said functions are defined in four different packages. Inputs are user defined and use the concept of inheritance for the division operation.

Solu.

sol6.java

```java
/*
 * Copyright (c) 2021.
 * Divyanshu Tyagi
 * NIT Warangal
 * 207919
 */

package Assignment2.sol6;
import Assignment2.sol6.addition.*;
import Assignment2.sol6.division.divide;
import Assignment2.sol6.multiplication.*;
import Assignment2.sol6.subtraction.*;

import java.util.Scanner;

public class sol6 extends divide {
    public static void main(String[] args) {
        Double num1, num2;
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter 1st number : ");
        num1 = sc.nextDouble();
        System.out.print("Enter 2nd number : ");
        num2 = sc.nextDouble();

        System.out.println("!! Select Operation !!");
        System.out.println("1. Addition");
        System.out.println("2. Subtraction");
        System.out.println("3. Multiplication");
        System.out.println("4. Division");
        int option;
        option = sc.nextInt();

        switch (option) {
            case 1: {
                System.out.println("Addition of both is : " + new add(num1,num2).getResult());
                break;
            }
            case 2: {
                System.out.println("Subtraction of both is : " + new sub(num1,num2).getResult());
                break;
            }
            case 3: {
                System.out.println("Multiplication of both is : " + new mul(num1,num2).getResult());
                break;
            }
            case 4: {
                System.out.println("Division of both is : " + new sol6().getResult(num1,num2));
                break;
            }
            default: {
                System.out.println("Incorrect choice");
                break;
            }
        }
    }
```

```
    @Override
    public Double getResult(Double n1, Double n2) {
        return n1/n2;
    }
}
```

add.java

```
/*
 * Copyright (c) 2021.
 * Divyanshu Tyagi
 * NIT Warangal
 * 207919
 */

package Assignment2.sol6.addition;

public class add {
    private Double result;

    public add(Double num1,Double num2) {
        result = num1 + num2;
    }
    public Double getResult() {
        return result;
    }
}
```

divide.java

```
/*
 * Copyright (c) 2021.
 * Divyanshu Tyagi
 * NIT Warangal
 * 207919
 */

package Assignment2.sol6.division;

public abstract class divide {
    abstract public Double getResult(Double n1, Double n2);
}
```

sub.java

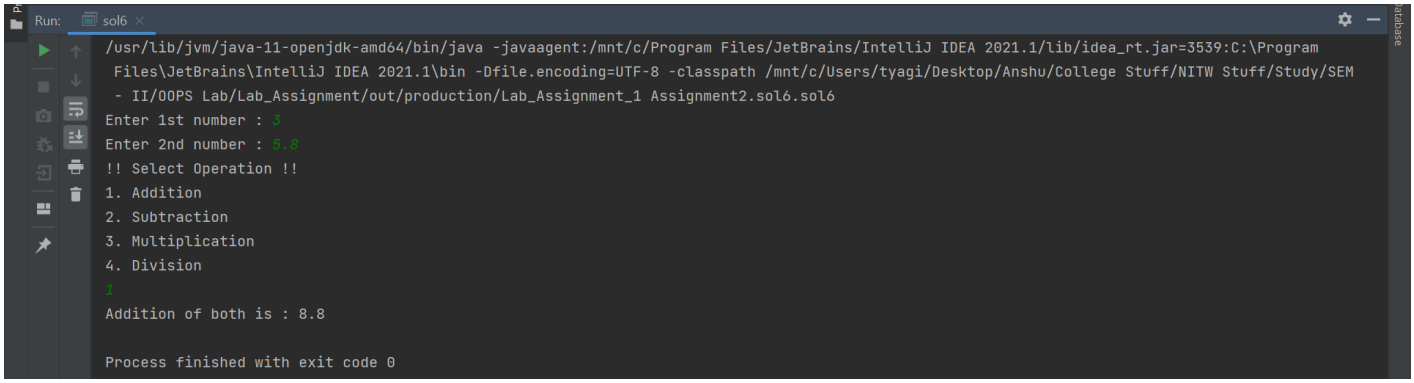
```
/*
 * Copyright (c) 2021.
 * Divyanshu Tyagi
 * NIT Warangal
 * 207919
 */

package Assignment2.sol6.subtraction;

public class sub {
    private  Double result;
    public sub(Double num1, Double num2) {
        result = num1 - num2;
    }

    public Double getResult() {
        return result;
```

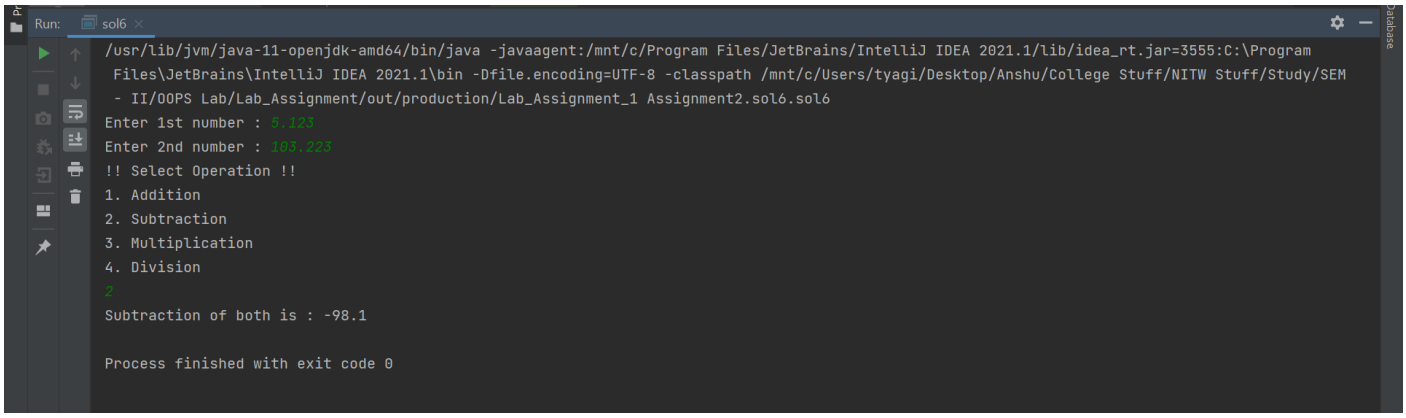
```
    }
}
```

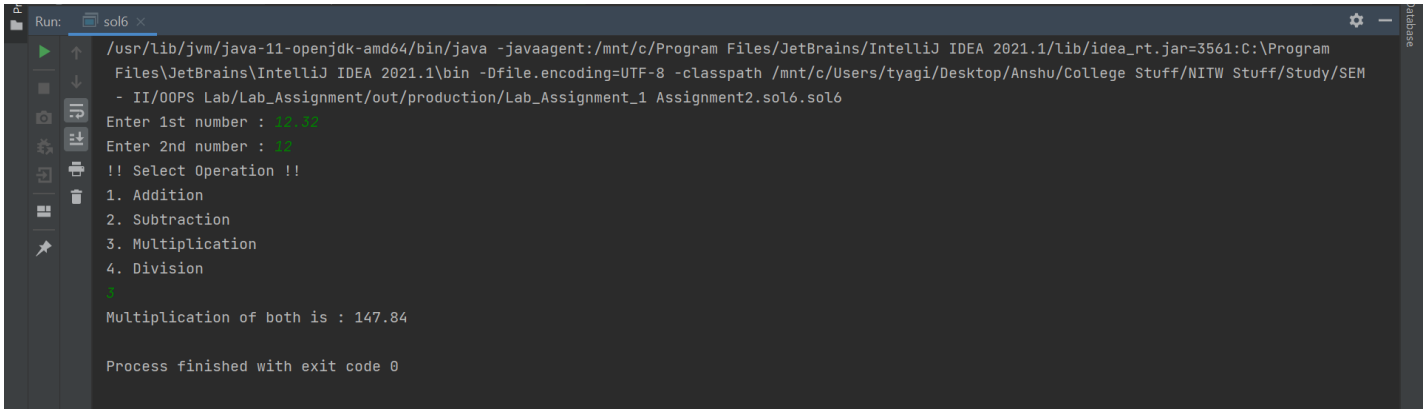
mul.java

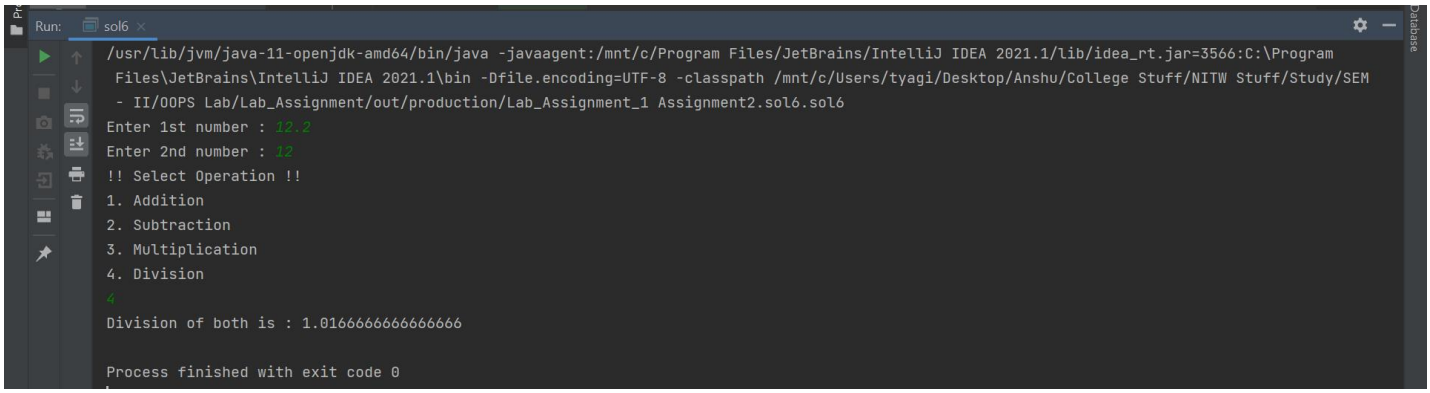
```
/*
 * Copyright (c) 2021.
 * Divyanshu Tyagi
 * NIT Warangal
 * 207919
 */

package Assignment2.sol6.multiplication;

public class mul {
    Double result;
    public mul(Double num1, Double num2) {
        result = num1 * num2;
    }
    public Double getResult() {
        return result;
    }
}
```

OUTPUT:









Q7. It is known that we can create any number of objects for a given class, But from the requirement analysis of a project you come to know that only 5 objects are required for a class. Now, the task is to implement a java program to create a class and if more than 5 objects are created to the class, print some appropriate message.

Solu.

```
/*
 * Copyright (c) 2021.
 * Divyanshu Tyagi
 * NIT Warangal
 * 207919
 */

package Assignment2;
class CounterTest {
    static int count = 0;
    {
        count++;
    }
    CounterTest() {
        if(count > 5) {
            System.out.println("Cannot create more than 5 Objects");
        }else {
            System.out.println("Object number : " + count + " Created Successfully");
        }
    }
}
public class sol7 {
    public static void main(String[] args) {
        CounterTest c1 = new CounterTest();
        CounterTest c2 = new CounterTest();
        CounterTest c3 = new CounterTest();
```

```
        CounterTest c4 = new CounterTest();
        CounterTest c5 = new CounterTest();
        CounterTest c6 = new CounterTest();
        CounterTest c7 = new CounterTest();
    }
}
```

OUTPUT: