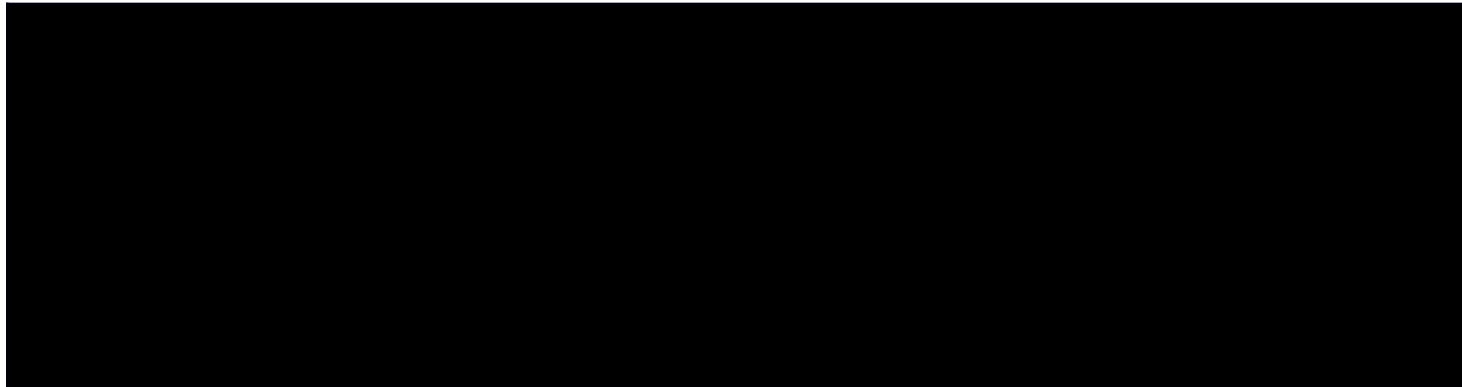




# Silver Oak College of Engineering & Technology

Object Oriented Programming with UML  
(1010043220)  
4<sup>th</sup> SEMESTER



## **DEPARTMENT OF INFORMATION TECHNOLOGY**

### **VISION**

To be recognized for the quality education and research in the field of Information Technology known for its accomplished graduates.

### **MISSION**

1. Continually improve the standard of our graduates by engaging in innovative teaching learning methods with high caliber motivated faculty members keeping in-line with the rapid technological advancements.
2. Promote and support research activities over a wide range of academic interests among students and staff for growth of individual knowledge and continuous learning.
3. Provide an education system that promotes innovation, creativity, entrepreneurial spirit, leadership as well as freedom of thought with emphasis on professionalism and ethical behavior.

### **PROGRAM EDUCATIONAL OBJECTIVES (PEO):**

**PEO1:**To provide fundamental knowledge of science and engineering for an IT professional and to equip them with proficiency of mathematical foundations and algorithmic principles and inculcate competent problem-solving ability.

**PEO2:**To implant ability in creativity & design of IT systems and transmit knowledge and skills to analyze, design, test and implement various software applications.

**PEO3:** To exhibit leadership capability, triggering social and economical commitment and inculcate community services.

**PEO4:**To inculcate professional-social ethics, teamwork in students and acquaint them with requisite technical and managerial skills to attain a successful career.

## **PROGRAM OUTCOMES (POs):**

### **Engineering Graduates will be able to:**

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

- 10. Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
- 11. Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
- 12. Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

**Object Oriented Programming with UML**  
**PRACTICAL BOOK**  
**DEPARTMENT OF INFORMATION TECHNOLOGY**

**PREFACE**

It gives us immense pleasure to present the first edition of Object Oriented Programming with UML Practical Book for the B.E. 2<sup>nd</sup> year students of Silver Oak College of Engineering and Technology.

The theory and laboratory course of Object Oriented Programming with UML , at Silver Oak College of Engineering and Technology, Ahmedabad, is designed in such a manner that students can develop the basic understanding of the subject during theory classes and gain the hands-on practical experience during their laboratory sessions. Java being one of the most prominent languages opens the door to much application development involving navies of Input/output, Networking, the standard utilities, Applets, Frameworks, GUI-based controls and much more.

The Laboratory Manual presented here to you takes you onto learning journey of Java, both at the basic and advanced levels. In this you will be exploring the wide range of topics from object oriented programming concepts of java, data types, variables, arrays, strings, various operators, control statements, classes & inheritance, input and output Mechanism to developing Swing and Applets along with Java.

Lab Manual Revised by: Prof. Kuldeep Jyani, Silver Oak College of Engineering and Technology

Lab Manual Revision No.: SOCET\_1010043220\_LM\_2020\_1

## **INSTRUCTIONS TO STUDENTS**

1. Be prompt in arriving to the laboratory and always come well prepared for the experiment.
2. Students need to maintain a proper decorum in the computer lab. Students must use the equipment with care. Any damage is caused is punishable.
3. Students are instructed to come to lab in formal dresses only.
4. Students are supposed to occupy the systems allotted to them and are not supposed to talk or make noise in the lab.
5. Students are required to carry their observation book and lab records with completed exercises while entering the lab.
6. Lab records need to be submitted every week.
7. Students are not supposed to use pen drives in the lab.
- 8.** The grades for the Object Oriented Programming with UML Practical course work will be awarded based on your performance in the laboratory, regularity, recording of experiments in the Object Oriented Programming with UML practical Final Notebook, lab quiz, regular viva-voce and end-term examination.
- 9.** Find the answers of all the questions mentioned under the section 'Post Practical Question' at the end of each experiment in the Object Oriented Programming with UML Practical Book.

## **TABLE OF CONTENT**

Sr. No	Experiment Title	Page No		Date of Start	Date of Completion	Sign	Marks (out of 10)
		To	From				
1	Write a Program that displays Welcome to Java, This is my first JAVA program.						
2	Write a program to convert rupees to dollar. 70 rupees=1 dollar.						
3	Write a program that calculates percentage marks of the student if marks of 6 subjects are given.						
4	Write a program that solves the following equation and displays the value x and y: 1) $3.4x + 50.2y = 44.5$ 2) $2.1x + .55y = 5.9$ (Assume Cramer's rule to solve equation $ax + by = e$ $x = \frac{ed - bf}{ad - bc}$ $y = \frac{af - ec}{ad - bc}$ )						
5	Write a program that reads a number in meters, converts it to feet, and displays the result.						
6	Write a program to count the number of words that start with capital letters.						
7	Write a program that prompts the user to enter three integers and display the integers in decreasing order.						
8	Write a program that prompts the user to enter a letter and check whether a letter is a vowel or constant.						

9	<p>Write an interactive program to print a string entered in a pyramid form. For instance, the string “stream” has to be displayed as follows:</p> <pre>       S      S t     S t r    S t r e   S t r e a  S t r e a m </pre>						
10	Write a program to find length of string and print second half of the string.						
11	Write a program to find that given number or string is palindrome or not.						
12	Assume a vehicle plate number consists of three uppercase letters followed by four digits. Write a program to generate a plate number.						
13	Write a java program to demonstrate multilevel inheritance						
14	Write a java program to find out the area of circle and area of rectangle using the concept of method overloading.						
15	Write a java program to find out the volume of rectangular box and volume of cub using the concept of constructor overloading.						
16	Design a class which finds out the factorial of a given number using recursion.						
17	Write a java program to demonstrate the use of dynamic method dispatch.						



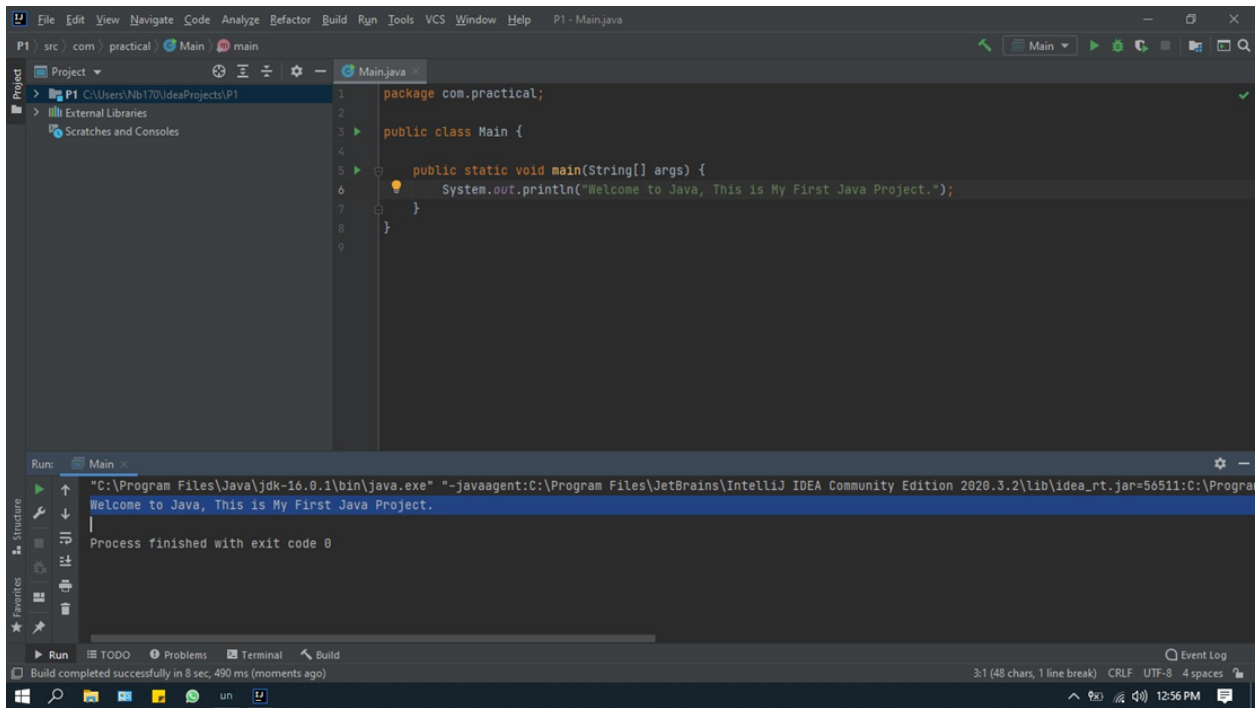
18	Write a java program to demonstrate the concept of interface.						
19	Write a program to show divide by zero error through exception, and also try to catch the exception.						
20	Write a program to create two threads, one thread will print odd numbers and second thread will print even numbers between 1 to 20 numbers.						
21	Create a class called Student. Write a student manager program to manipulate the student information from files by using FileInputStream and FileOutputStream						
22	Refine the student manager program to manipulate the student information from files by using the BufferedReader and BufferedWriter						
23	Draw use case diagram for hotel management system.						
24	Draw class diagram of bank management system.						
25	Prepare a sequence diagram for issuing a book in the library management system.						

## Practical:1

**Write a Program that displays Welcome to Java, this is my first JAVA program.**

```
public class Main {  
  
    public static void main(String[] args) {  
        System.out.println("Welcome to Java, This is My First Java Project.");  
    }  
}
```

## OUTPUT:

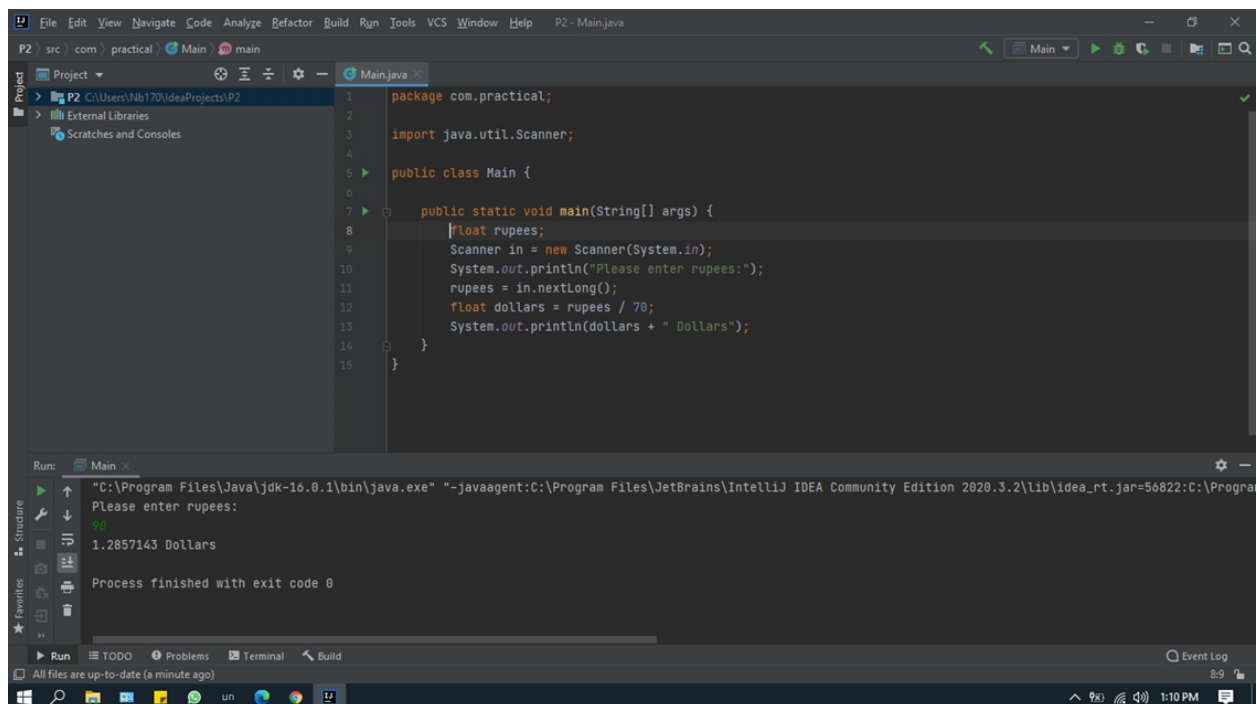


## Practical:2

**Write a program to convert rupees to dollar. 70 rupees=1 dollar.**

```
import java.util.Scanner; public class Main {  
public static void main(String[] args) {  
  
float rupees;  
Scanner in = new Scanner(System.in);  
System.out.println("Please enter rupees:");  
    rupees = in.nextLong();  
float dollars = rupees / 70;  
System.out.println(dollars + " Dollars");  
}  
}
```

**OUTPUT:**



The screenshot displays an IDE window with the following components:

- Editor:** Shows the Java code for converting rupees to dollars. The code is as follows:

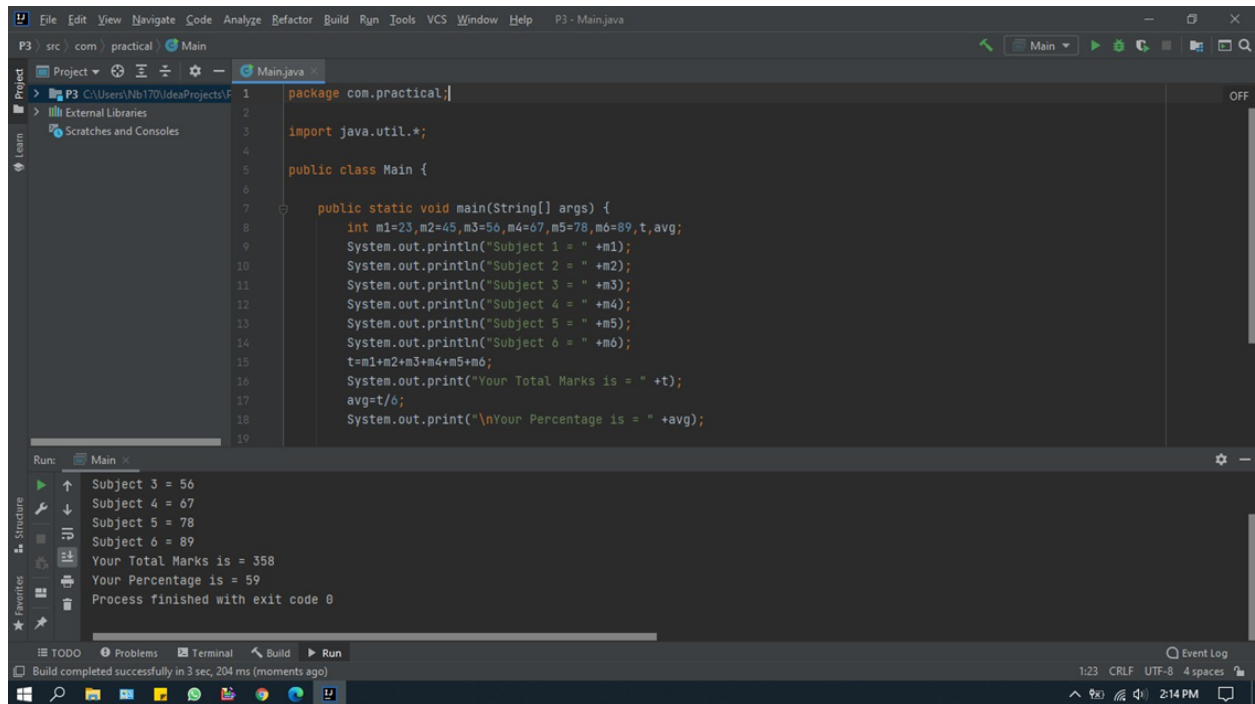
```
1 package com.practical;  
2  
3 import java.util.Scanner;  
4  
5 public class Main {  
6  
7     public static void main(String[] args) {  
8         float rupees;  
9         Scanner in = new Scanner(System.in);  
10        System.out.println("Please enter rupees:");  
11        rupees = in.nextLong();  
12        float dollars = rupees / 70;  
13        System.out.println(dollars + " Dollars");  
14    }  
15 }
```
- Run Console:** Shows the execution output. The prompt "Please enter rupees:" is followed by the input "12857143". The output is "1.2857143 Dollars". The console also shows "Process finished with exit code 0".
- Run Command:** The command executed is "C:\Program Files\Java\jdk-16.0.1\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2020.3.2\lib\idea\_rt.jar=56822:C:\Progra".

### Practical:3

**Write a program that calculates percentage marks of the student if marks of 6 subjects are given.**

```
import java.util.*; public class Main {  
public static void main(String[] args) {  
int m1=23,m2=45,m3=56,m4=67,m5=78,m6=89,t,avg;  
System.out.println("Subject 1 = " +m1);  
System.out.println("Subject 2 = " +m2);  
System.out.println("Subject 3 = " +m3);  
System.out.println("Subject 4 = " +m4);  
System.out.println("Subject 5 = " +m5);  
System.out.println("Subject 6 = " +m6); t=m1+m2+m3+m4+m5+m6;  
System.out.print("Your Total Marks is = " +t);  
avg=t/6;  
System.out.print("\nYour Percentage is = " +avg);  
  
}  
}
```

**OUTPUT:**



The screenshot shows an IDE with a Java file named Main.java. The code defines a class Main with a main method that calculates the total marks and percentage for six subjects. The output window shows the results of the program execution.

```
package com.practical;

import java.util.*;

public class Main {

    public static void main(String[] args) {
        int m1=23,m2=45,m3=56,m4=67,m5=78,m6=89,t,avg;
        System.out.println("Subject 1 = " +m1);
        System.out.println("Subject 2 = " +m2);
        System.out.println("Subject 3 = " +m3);
        System.out.println("Subject 4 = " +m4);
        System.out.println("Subject 5 = " +m5);
        System.out.println("Subject 6 = " +m6);
        t=m1+m2+m3+m4+m5+m6;
        System.out.print("Your Total Marks is = " +t);
        avg=t/6;
        System.out.print("\nYour Percentage is = " +avg);
    }
}
```

Run: Main x

```
Subject 3 = 56
Subject 4 = 67
Subject 5 = 78
Subject 6 = 89
Your Total Marks is = 358
Your Percentage is = 59
Process finished with exit code 0
```

## Practical:4

Write a program that solves the following equation and displays the value x and y: 1)

$$3.4x+50.2y=44.5$$

2)  $2.1x+.55y=5.9$  (Assume Cramer's rule to solve equation  $ax+by=e$   $x=\frac{ed-bf}{ad-bcc}$   $y=\frac{af-ec}{ad-bc}$ )

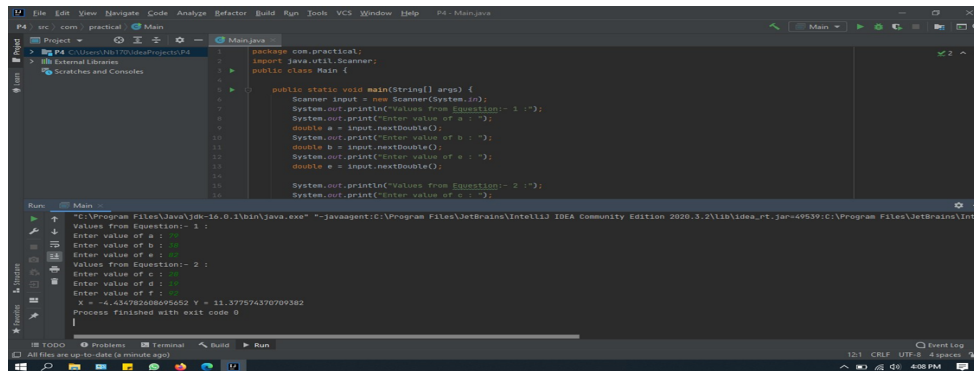
```
import java.util.Scanner; public class Main {
public static void main(String[] args) {
Scanner input = new Scanner(System.in);
System.out.println("Values from Equestion:- 1 :");
System.out.print("Enter value of a : ");
double a = input.nextDouble();
System.out.print("Enter value of b : ");
double b = input.nextDouble();
System.out.print("Enter value of e : ");
double e = input.nextDouble();
System.out.println("Values from Equestion:- 2 :");
System.out.print("Enter value of c : ");
double c = input.nextDouble();
```

```

System.out.print("Enter value of d : ");
double d = input.nextDouble();
System.out.print("Enter value of f : ");
double f = input.nextDouble();
double x = ((e*d)-(b*f))/((a*d)-(b*c));
double y = ((a*f)-(e*c))/((a*d)-(b*c));
System.out.print(" X = "+ x + " Y = " + y);
} }

```

## OUTPUT:



The screenshot shows an IDE with a Java file named Main.java. The code is as follows:

```

package com.practical;
import java.util.Scanner;
public class Main {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.println("Values from Equation:- 2 :");
        System.out.print("Enter value of a : ");
        double a = input.nextDouble();
        System.out.print("Enter value of b : ");
        double b = input.nextDouble();
        System.out.print("Enter value of c : ");
        double c = input.nextDouble();
        System.out.print("Enter value of d : ");
        double d = input.nextDouble();
        System.out.print("Enter value of e : ");
        double e = input.nextDouble();
        System.out.println("Values from Equation:- 2 :");
        System.out.print("X = ");
        double x = ((e*d)-(b*f))/((a*d)-(b*c));
        System.out.print("Y = ");
        double y = ((a*f)-(e*c))/((a*d)-(b*c));
        System.out.print("X = "+ x + " Y = " + y);
    }
}

```

The Run console shows the following output:

```

C:\Program Files\Java\jdk-16.0.1\bin\java.exe -javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2020.3.2\lib\idea_rt.jar=49539:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2020.3.2\bin -Dfile.encoding=UTF-8
Enter value of a : 1
Enter value of b : 2
Enter value of c : 3
Enter value of d : 4
Enter value of e : 5
Values from Equation:- 2 :
X = -4.434782608695652 Y = 11.37757437090909
Process finished with exit code 0

```

## Practical:5

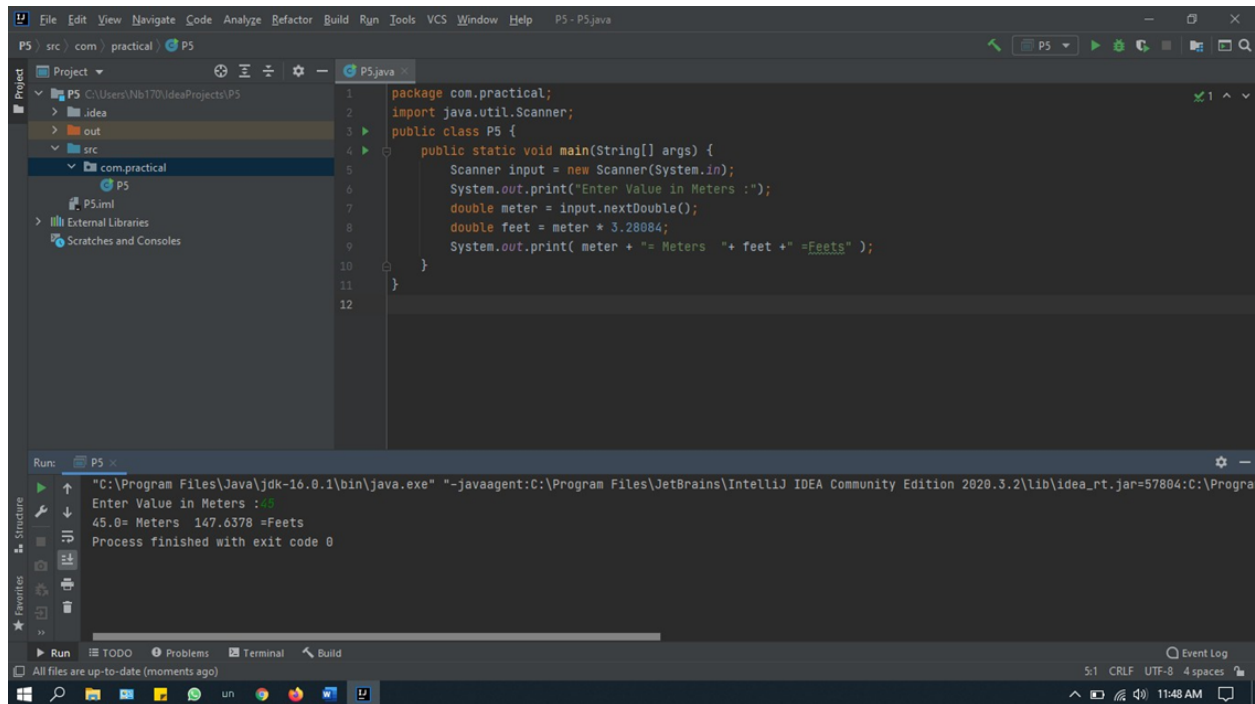
**Write a program that reads a number in meters, converts it to feet, and displays the result.**

```

import java.util.Scanner; public class P5 {
public static void main(String[] args) {
Scanner input = new Scanner(System.in);
System.out.print("Enter Value in Meters :");
double meter = input.nextDouble();
double feet = meter * 3.28084;
System.out.print( meter + "= Meters "+ feet +" =Feets" );
}
}

```

## OUTPUT:



## Practical:6

**Write a program to count the number of words that start with capital letters.**

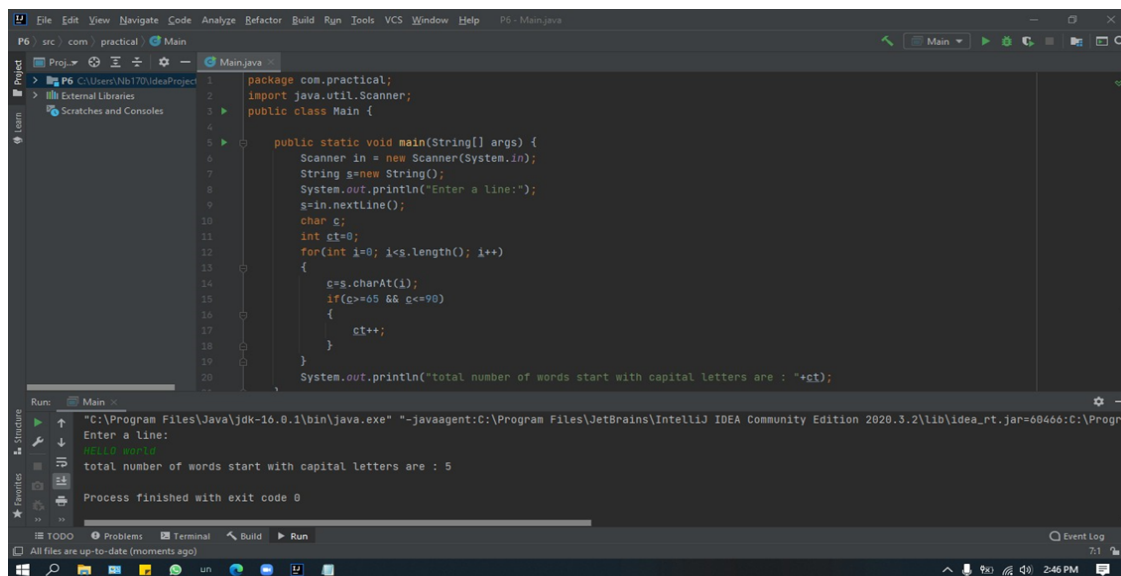
```
import java.util.Scanner; public class Main {  
public static void main(String[] args) {  
Scanner in = new Scanner(System.in);  
String s=new String();  
System.out.println("Enter a line:");  
s=in.nextLine();  
char c; int ct=0;  
for(int i=0; i<s.length(); i++)
```

```

{
c=s.charAt(i);
if(c>=65 && c<=90)
{
ct++;
}
}
System.out.println("total number of words start with capital letters are : "+ct);
}
}

```

**OUTPUT:**



The screenshot shows an IDE window with a Java file named 'Main.java'. The code defines a class 'Main' with a 'main' method. The 'main' method uses a 'Scanner' to read a line of input, iterates through each character, and counts the number of capital letters. The output of the program is displayed in the 'Run' console, showing the input 'hello world' and the output 'total number of words start with capital letters are : 5'.

```

package com.practical;
import java.util.Scanner;
public class Main {

    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);
        String s=new String();
        System.out.println("Enter a line:");
        s=in.nextLine();
        char c;
        int ct=0;
        for(int i=0; i<s.length(); i++)
        {
            c=s.charAt(i);
            if(c>=65 && c<=90)
            {
                ct++;
            }
        }
        System.out.println("total number of words start with capital letters are : "+ct);
    }
}

```

Run: Main

```

C:\Program Files\Java\jdk-10.0.1\bin\java.exe" --javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2020.3.2\lib\idea_rt.jar=60466:C:\Progr
Enter a line:
hello world
total number of words start with capital letters are : 5
Process finished with exit code 0

```

## Practical :7

**Write a program that prompts the user to enter three integers and display the integers in decreasing order.**

```

import java.util.Scanner;

public class Main {

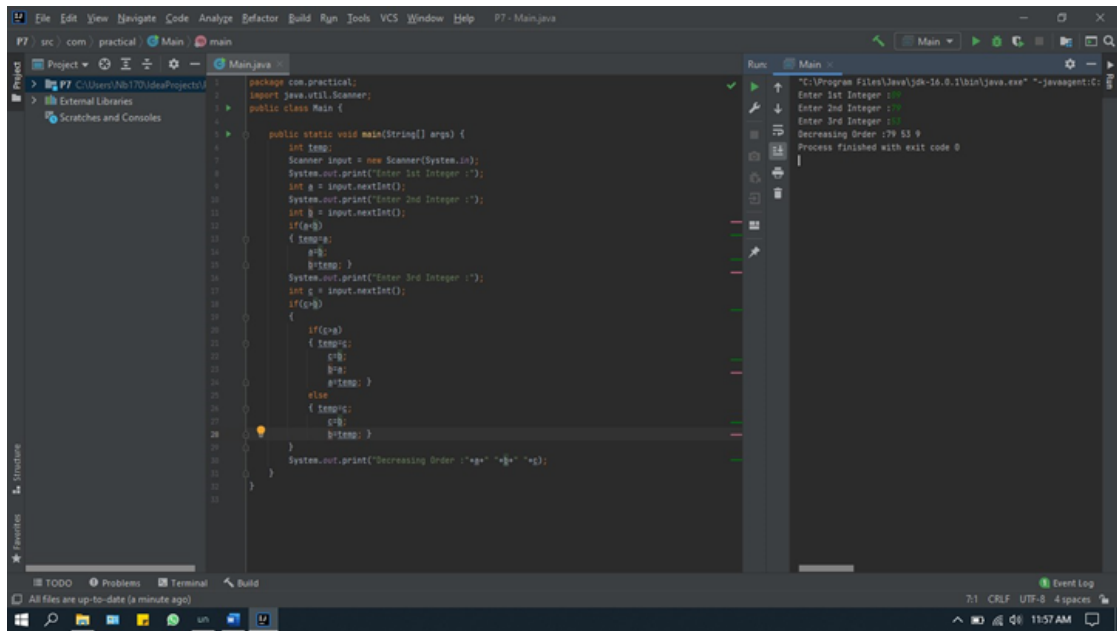
    public static void main(String[] args) {
        int temp;
        Scanner input = new Scanner(System.in);
        System.out.print("Enter 1st Integer :");
    }
}

```



```
int a = input.nextInt();
System.out.print("Enter 2nd Integer :");
int b = input.nextInt();
if(a<b)
{ temp=a;
a=b;
b=temp; }
System.out.print("Enter 3rd Integer :");
int c = input.nextInt();
if(c>b)
{
if(c>a)
{ temp=c;
c=b;
b=a;
a=temp; }
else
{
temp=c;
c=b;
b=temp;
}
}
System.out.print("Decreasing Order :"+a+" "+b+" "+c");
}
}
```

**Output:**

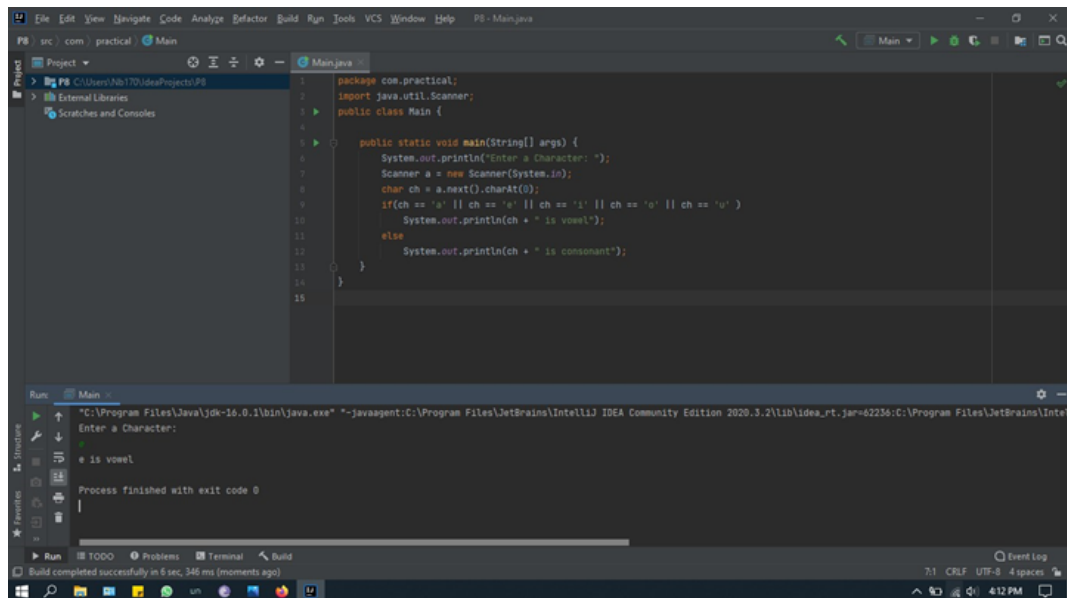


## Practical :8

**Write a program that prompts the user to enter a letter and check whether a letter is a vowel or constant.**

```
import java.util.Scanner;
public class Main {
    public static void main(String[] args) {
        System.out.println("Enter a Character: ");
        Scanner a = new scanner(System.in)
        Char ch = a.next().charAt(0);
        If(ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u')
        System.out.println(ch+"is vowel");
        Else
        System.out.print(ch +"is Consonants");
    }
}
```

Output:

The screenshot shows an IDE window with a project named 'P8'. The main file 'Main.java' contains the following code:

```
1 package com.practical;
2 import java.util.Scanner;
3 public class Main {
4
5     public static void main(String[] args) {
6         System.out.println("Enter a Character: ");
7         Scanner a = new Scanner(System.in);
8         char ch = a.next().charAt(0);
9         if(ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u')
10             System.out.println(ch + " is vowel");
11         else
12             System.out.println(ch + " is consonant");
13     }
14 }
15
```

The Run window at the bottom shows the command executed: `"C:\Program Files\Java\jdk-16.0.1\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2020.3.2\lib\idea_rt.jar=62236:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2020.3.2\bin" -Didea.config.path=C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2020.3.2\conf -Didea.copyright=Copyright (c) 2020 JetBrains s.r.o. -Didea.log=C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2020.3.2\log -Didea.platform.prefix=JDK -Didea.vendor.id=jetbrains -Didea.version=2020.3.2 -jar C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2020.3.2\bin\idea_rt.jar 62236`. The output shows: `Enter a Character: e` followed by `e is vowel`. The process finished with exit code 0. The status bar at the bottom indicates 'Build completed successfully in 6 sec, 348 ms (moments ago)'.

## Practical :9

**Write an interactive program to print a string entered in a pyramid form. For instance, the string “stream” has to be displayed as follows:**

```
S
S t
S t r
S t r e
S t r e a
S t r e a m
```

```
import java.util.Scanner;
public class Main {
public static void main(String[] args) {
char c;
int i,j;
Scanner in= new Scanner(System.in);
String s;//=new String("STRING");
System.out.println("Enter a string:");
s=in.next();
int k,d;
for(i=0;i<s.length();i++)
{
for(k=0;k<s.length()-i;k++) {
System.out.print(" ");
}
for(j=0;j<=i;j++)
{
c=s.charAt(j);
System.out.print(c+" ");
}
System.out.println(" ");
}
}
}
```

Output:

The screenshot shows an IDE with a Java file named `Main.java`. The code implements a method to reverse a string using two nested loops. The first loop prints each character of the string in reverse order, and the second loop prints the original string. The output window shows the execution of the program, displaying the reversed string and the original string.

```
package com.practical;
import java.util.Scanner;
public class Main {

    public static void main(String[] args) {
        char s;
        int i,j;
        Scanner in= new Scanner(System.in);

        String s;//new String("STREAM");
        System.out.println("Enter a string:");
        s=in.next();
        int k=0;
        for(i=0;i<s.length();i++)
        {
            for(k=s.length()-1;k>=i;k--) {
                System.out.print(" ");
            }
            for(j=0;j<=i;j++)
            {
                c=s.charAt(j);
                System.out.print(c+" ");
            }
            System.out.println(" ");
        }
    }
}
```

Run: Main

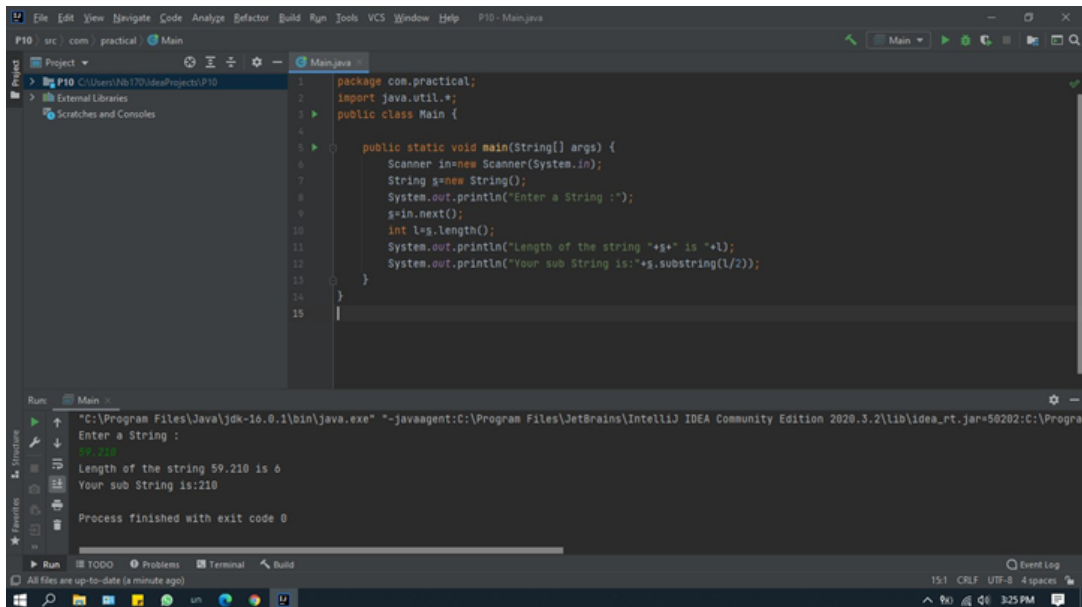
```
"C:\Program Files\Java\jdk-10.0.1\bin\java.exe"
Enter a string:
String
S
t
r
e
a
m
String
Process finished with exit code 0
```

## Practical :10

**Write a program to find length of string and print second half of the string.**

```
import java.util.*;
public class Main {
    public static void main(String[] args) {
        Scanner in=new Scanner(System.in);
        String s=new String();
        System.out.println("Enter a String :");
        s=in.next();
        int l=s.length();
        System.out.println("Length of the string "+s+" is "+l);
        System.out.println("Your sub String is:"+s.substring(l/2));
    }
}
```

Output:

The screenshot shows an IDE window with a Java file named 'Main.java'. The code is as follows:

```
1 package com.practical;
2 import java.util.*;
3 public class Main {
4
5     public static void main(String[] args) {
6         Scanner in=new Scanner(System.in);
7         String s=new String();
8         System.out.println("Enter a String :");
9         s=in.next();
10        int l=s.length();
11        System.out.println("Length of the string "+s+" is "+l);
12        System.out.println("Your sub String is:"+s.substring(l/2));
13    }
14 }
15
```

The 'Run' tab at the bottom shows the execution output:

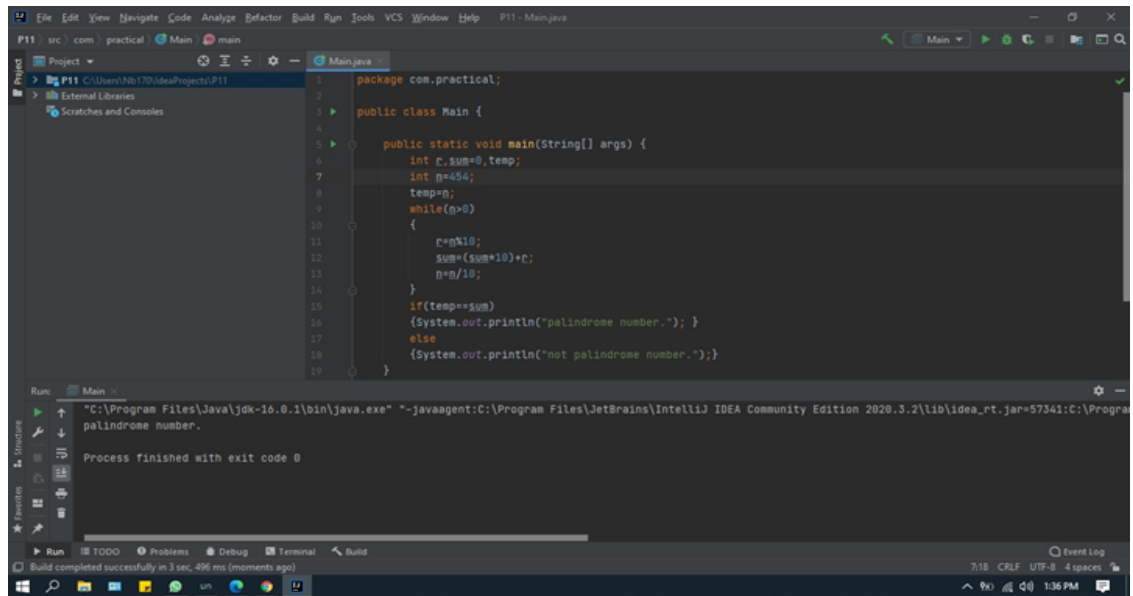
```
"C:\Program Files\Java\jdk-16.0.1\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2020.3.2\lib\idea_rt.jar=50202:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2020.3.2\bin" -Dfile.encoding=UTF-8
Enter a String :
59.210
Length of the string 59.210 is 6
Your sub String is:210
Process finished with exit code 0
```

## Practical :11

**Write a program to find that given number or string is palindrome or not.**

```
public class Main {  
    public static void main(String[] args) {  
        int r,sum=0,temp;  
        int n=454;  
        temp=n;  
        while(n>0)  
        {  
            r=n%10;  
            sum=(sum*10)+r;  
            n=n/10;  
        }  
        if(temp==sum)  
        {  
            System.out.println("palindrome number.");  
        }  
        else  
        {  
            System.out.println("not palindrome number.");  
        }  
    }  
}
```

Output:



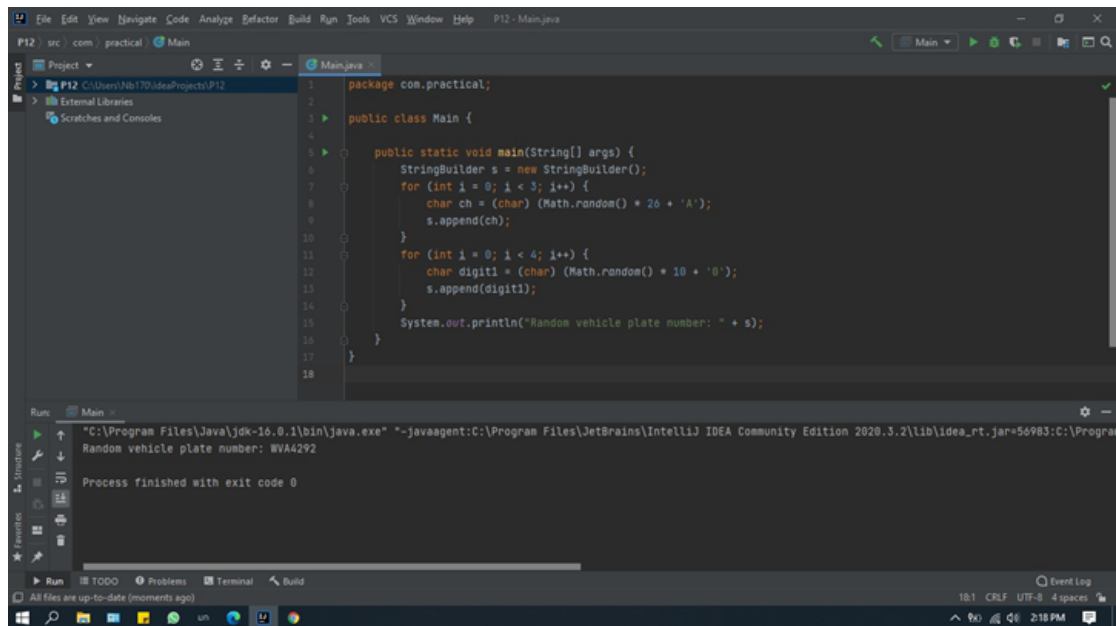


## Practical :12

**Assume a vehicle plate number consists of three uppercase letters followed by four digits.  
Write a program to generate a plate number.**

```
public class Main {  
    public static void main(String[] args) {  
        StringBuilder s = new StringBuilder();  
        for (int i = 0; i < 3; i++) {  
            char ch = (char) (Math.random() * 26 + 'A');  
            s.append(ch);  
        }  
        for (int i = 0; i < 4; i++) {  
            char digit1 = (char) (Math.random() * 10 + '0');  
            s.append(digit1);  
        }  
        System.out.println("Random vehicle plate number: " + s);  
    }  
}
```

Output:



The screenshot displays the IntelliJ IDEA IDE interface. The main editor window shows a Java file named `Main.java` with the following code:

```
1 package com.practical;  
2  
3 public class Main {  
4  
5     public static void main(String[] args) {  
6         StringBuilder s = new StringBuilder();  
7         for (int i = 0; i < 3; i++) {  
8             char ch = (char) (Math.random() * 26 + 'A');  
9             s.append(ch);  
10        }  
11        for (int i = 0; i < 4; i++) {  
12            char digit1 = (char) (Math.random() * 10 + '0');  
13            s.append(digit1);  
14        }  
15        System.out.println("Random vehicle plate number: " + s);  
16    }  
17 }  
18
```

The Run tool window at the bottom shows the execution of the program. The command used is:

```
"C:\Program Files\Java\jdk-10.0.1\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2020.3.2\lib\idea_rt.jar=56983:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2020.3.2\bin" -Didea.config.path=C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2020.3.2\conf -Didea.copyright.profiles=C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2020.3.2\copyright -Didea.home.path=C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2020.3.2\bin -Didea.platform.prefix=JDK -Didea.vendor.id=idea -Didea.version=2020.3.2 -jar C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2020.3.2\bin\idea_rt.jar=56983:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2020.3.2\bin
```

The output of the program is:

```
Random vehicle plate number: WVA4292
```

The process finished with exit code 0. The status bar at the bottom indicates the file encoding is UTF-8 with 4 spaces.

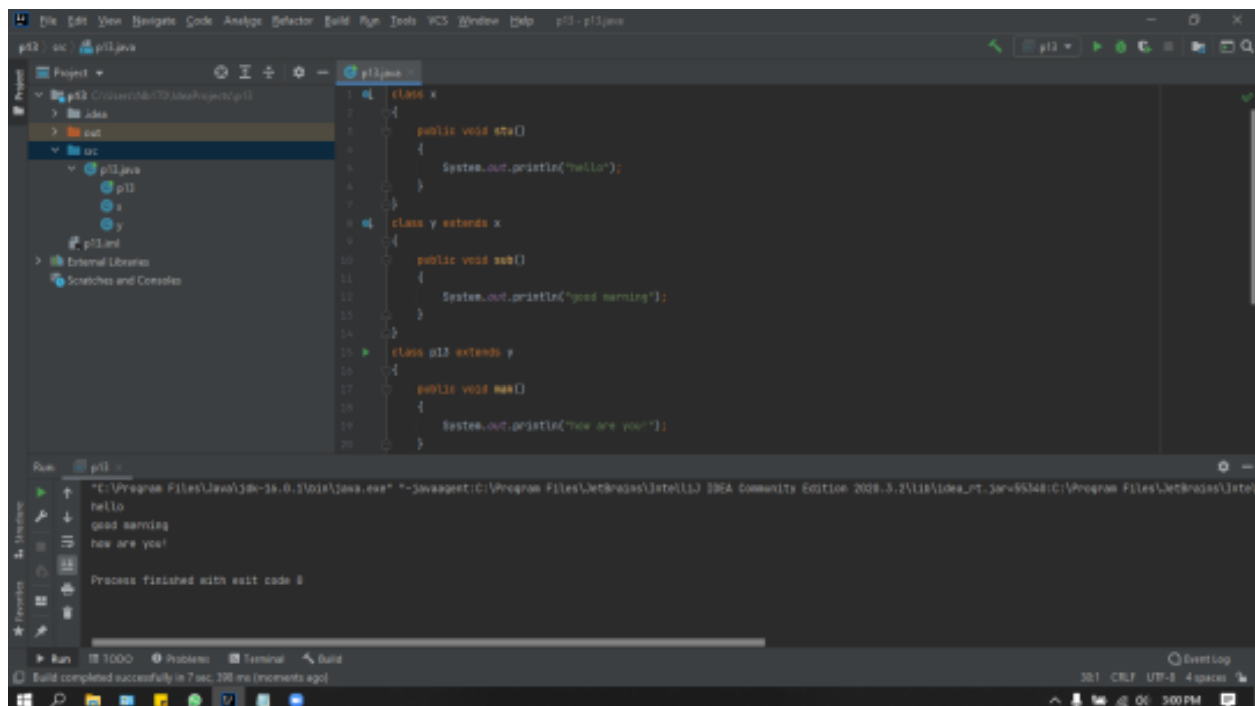
### Practical:13

**Write a java program to demonstrate multilevel inheritance.**

```
class x
{
public void stu()
{
System.out.println("hello");
}
}
class y extends x
{
public void sub()
{
System.out.println("good morning");
}
}
class p13 extends y
{
public void mak()
{
System.out.println("how are you!");
}
public static void main(String args[])

{
p13 dis=new p13();
dis.stu();
dis.sub();
dis.mak();
}
}
```

Output:-



## Practical:14

**Write a java program to find out the area of circle and area of rectangle & Square using the concept of method overloading.**

```

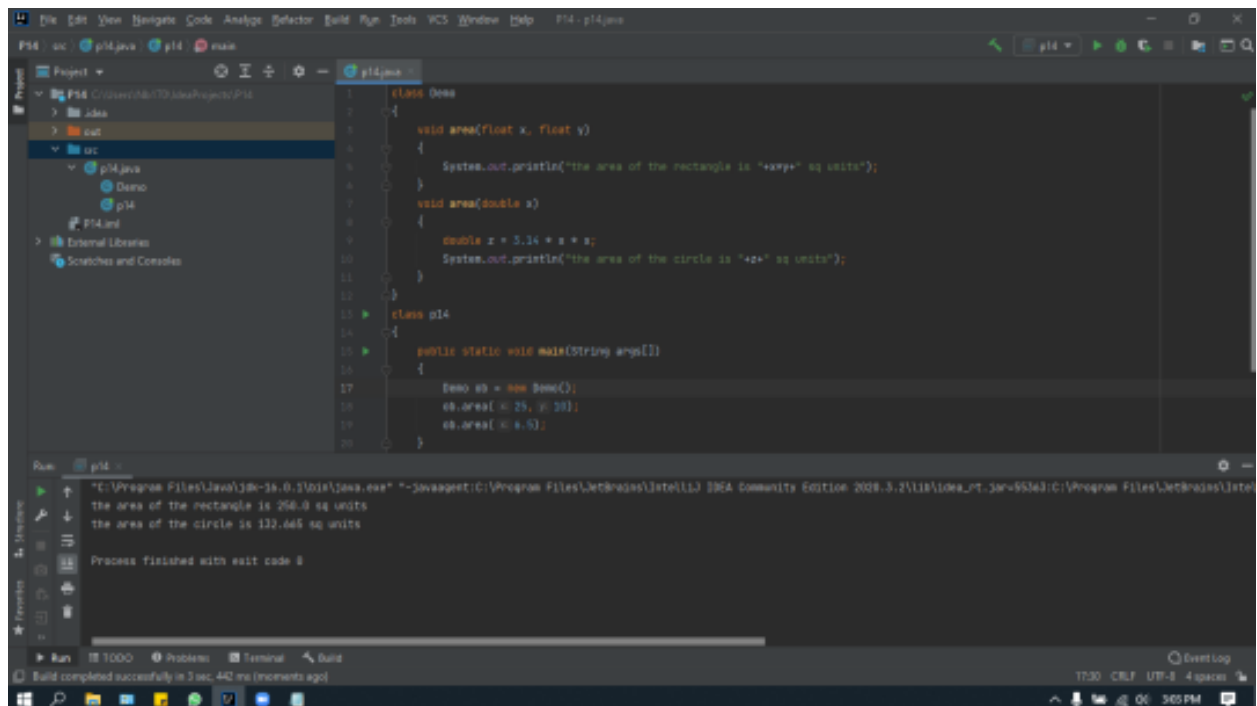
class Demo
{
void area(float x, float y)
{
System.out.println("the area of the rectangle is "+x*y+" sq units");
}
void area(double x)
{
double z = 3.14 * x * x;
System.out.println("the area of the circle is "+z+" sq units");
}
}

```

class p14

```
{  
public static void main(String args[])  
{  
Demo ob = new Demo();  
ob.area(25,10);  
ob.area(6.5);  
}  
}
```

Output:-



The screenshot shows an IDE window with a project named 'P14'. The code editor displays the following Java code:

```
1 class Demo  
2 {  
3     void area(float x, float y)  
4     {  
5         System.out.println("the area of the rectangle is "+x*y+" sq units");  
6     }  
7     void area(double s)  
8     {  
9         double x = 3.14 * s * s;  
10        System.out.println("the area of the circle is "+s+" sq units");  
11    }  
12 }  
13  
14 class p14  
15 {  
16     public static void main(String args[])  
17     {  
18         Demo ob = new Demo();  
19         ob.area(25, 10);  
20         ob.area(6.5);  
21     }  
22 }
```

The Run window shows the output of the program:

```
"C:\Program Files\Java\jdk-16.0.1\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ 2020.3.2\lib\idea_rt.jar=5563:C:\Program Files\JetBrains\IntelliJ 2020.3.2\bin" 5563  
the area of the rectangle is 250.0 sq units  
the area of the circle is 122.665 sq units  
Process finished with exit code 0
```

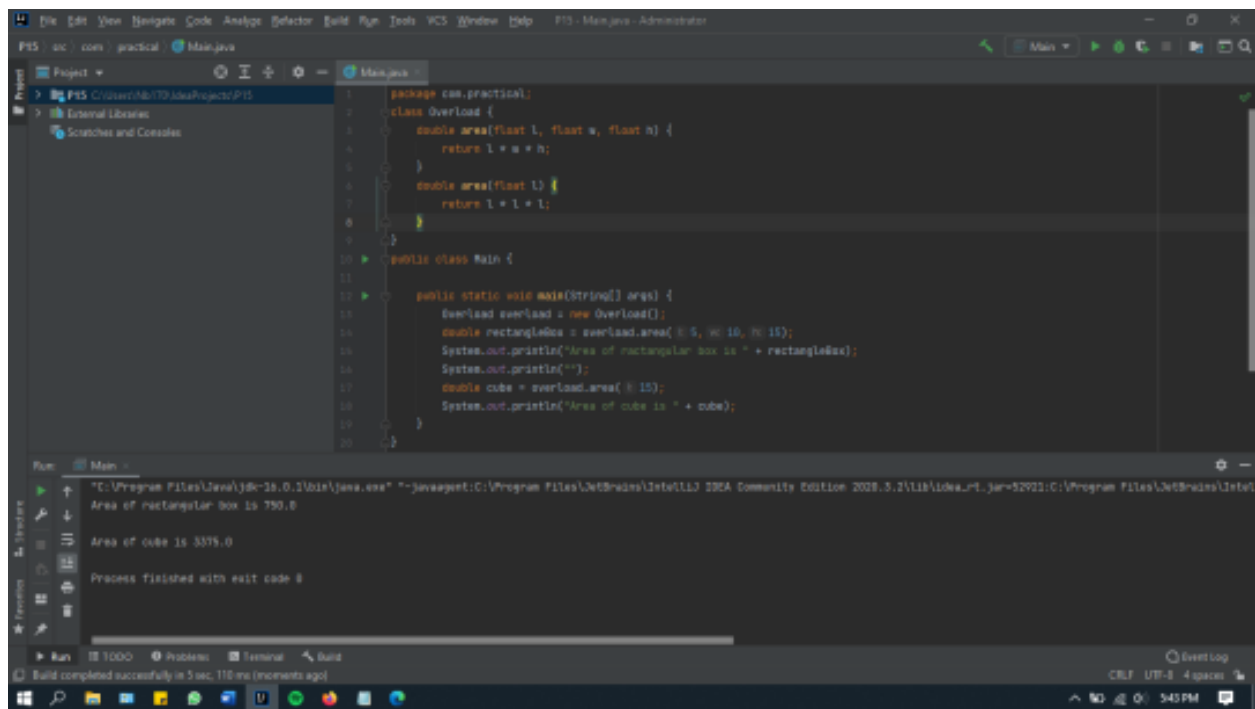
The status bar at the bottom indicates that the build completed successfully in 3 sec, 442 ms (moments ago).

### Practical:15

**Write a java program to find out the volume of rectangular box and volume of cube using the concept of constructor overloading.**

```
class Overload {  
    double area(float l, float w, float h) {  
        return l * w * h;  
    }  
    double area(float l) {  
        return l * l * l;  
    }  
}  
public class Main {  
  
    public static void main(String[] args) {  
        Overload = new Overload();  
        double rectangleBox = overload.area(5,10,15);  
        System.out.println("Area of ractangular box is " + rectangleBox);  
        System.out.println("");  
        double cube = overload.area(15);  
        System.out.println("Area of cube is " + cube);  
    }  
}
```

Output:-



## Practical:16

**Design a class which finds out the factorial of a given number using recursion.**

```
import java.util.Scanner;

public class Main {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter the number:");
        int num = scanner.nextInt();
```

```
int factorial = fact(num);
```

```
System.out.println("Factorial of entered number is: "+factorial);
```

```
}
```

```
static int fact(int n)
```

```
{
```

```
int output;
```

```
if(n==1){
```

```
return 1;
```

```
}yes
```

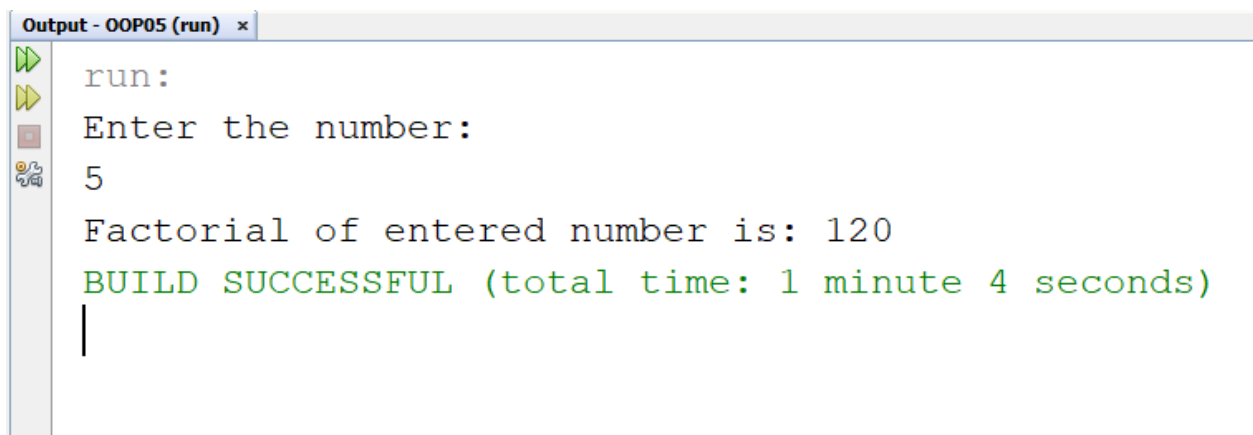
```
output = fact(n-1)* n;
```

```
return output;
```

```
}
```

```
}
```

Output:-



The screenshot shows an IDE output window titled "Output - OOP05 (run) x". On the left side, there are four icons: a green play button, a yellow play button, a red stop button, and a bug icon. The output text is as follows:

```
run:
Enter the number:
5
Factorial of entered number is: 120
BUILD SUCCESSFUL (total time: 1 minute 4 seconds)
|
```



### **Practical:17**

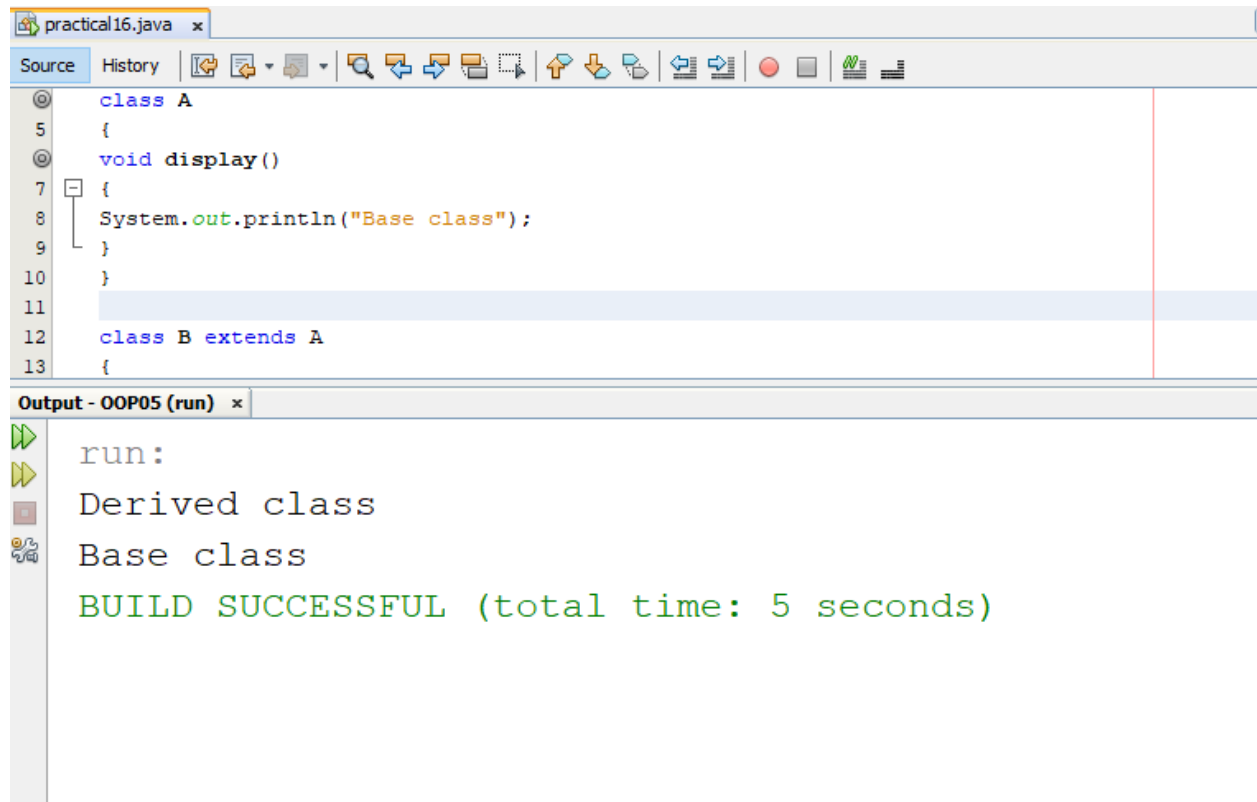
**Write a java program to demonstrate the use of dynamic method dispatch.**

```
class A
{
void display()
{
System.out.println("Base class");
}
}

class B extends A
{
void display ()
{
System.out.println("Derived class");
}
}

class p17
{
public static void main(String args[])
{
B x=new B();
x.display();
A y=new A();
y.display();
}
}
```

Output:-



The screenshot shows an IDE window titled 'practical16.java'. The code editor contains the following Java code:

```
class A
{
    void display()
    {
        System.out.println("Base class");
    }
}

class B extends A
{
    }
```

Below the code editor, the 'Output - OOP05 (run)' window displays the following text:

```
run:
Derived class
Base class
BUILD SUCCESSFUL (total time: 5 seconds)
```

### Practical:18

Write a java program to demonstrate the concept of interface.

```
interface A
{
    public void B();

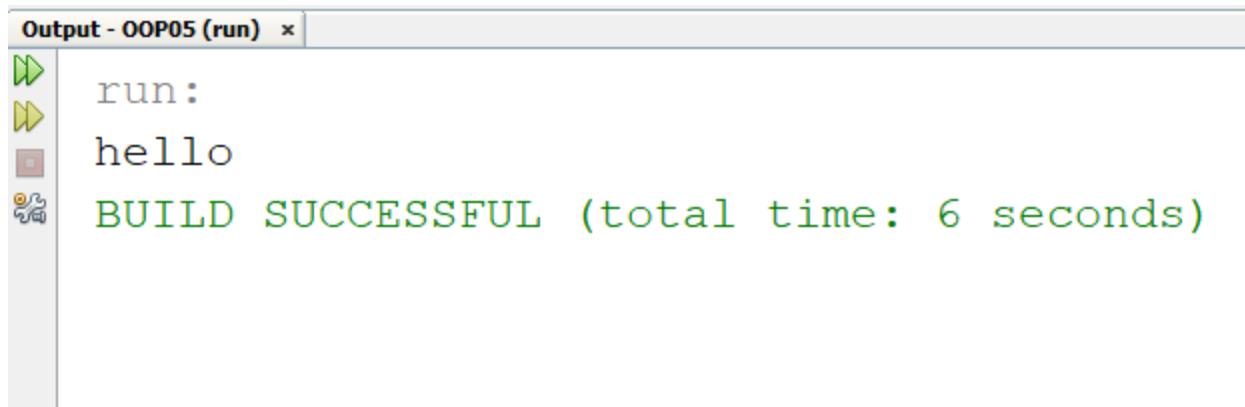
    public void C();
}

class p18 implements A
{
    public void B()
    {
        System.out.println("hello");
    }

    public void C()
```

```
{  
System.out.println("Welcome");  
}  
public static void main(String args[])  
{  
A x= new p18();  
x.B();  
}  
}
```

Output:-



```
Output - OOP05 (run) x  
run:  
hello  
BUILD SUCCESSFUL (total time: 6 seconds)
```

**Write a program to show divide by zero error through exception, and also try to catch the exception.**

```
public class Main {  
  
    public static void main(String[] args) {  
        int num1 = 17, num2 = 70, result = 0;  
        try{  
            result = num1/num2;  
            System.out.println("The result is : " +result);  
        }  
        catch (ArithmeticException e) {  
            System.out.println ("Can't be divided by Zero " + e);  
        }  
    }  
}
```

Output:-

```
1 public class Main {
2
3
4     public static void main(String[] args) {
5         int num1 = 17, num2 = 70, result = 0;
6         try {
7             result = num1 / num2;
8             System.out.println("The result is : " + result);
9         } catch (ArithmeticException e) {
10             System.out.println("Can't be divided by Zero " + e);
11         }
12     }
13 }
14
```

Output - OOP05 (run) x

```
run:
The result is : 0
BUILD SUCCESSFUL (total time: 0 seconds)
```

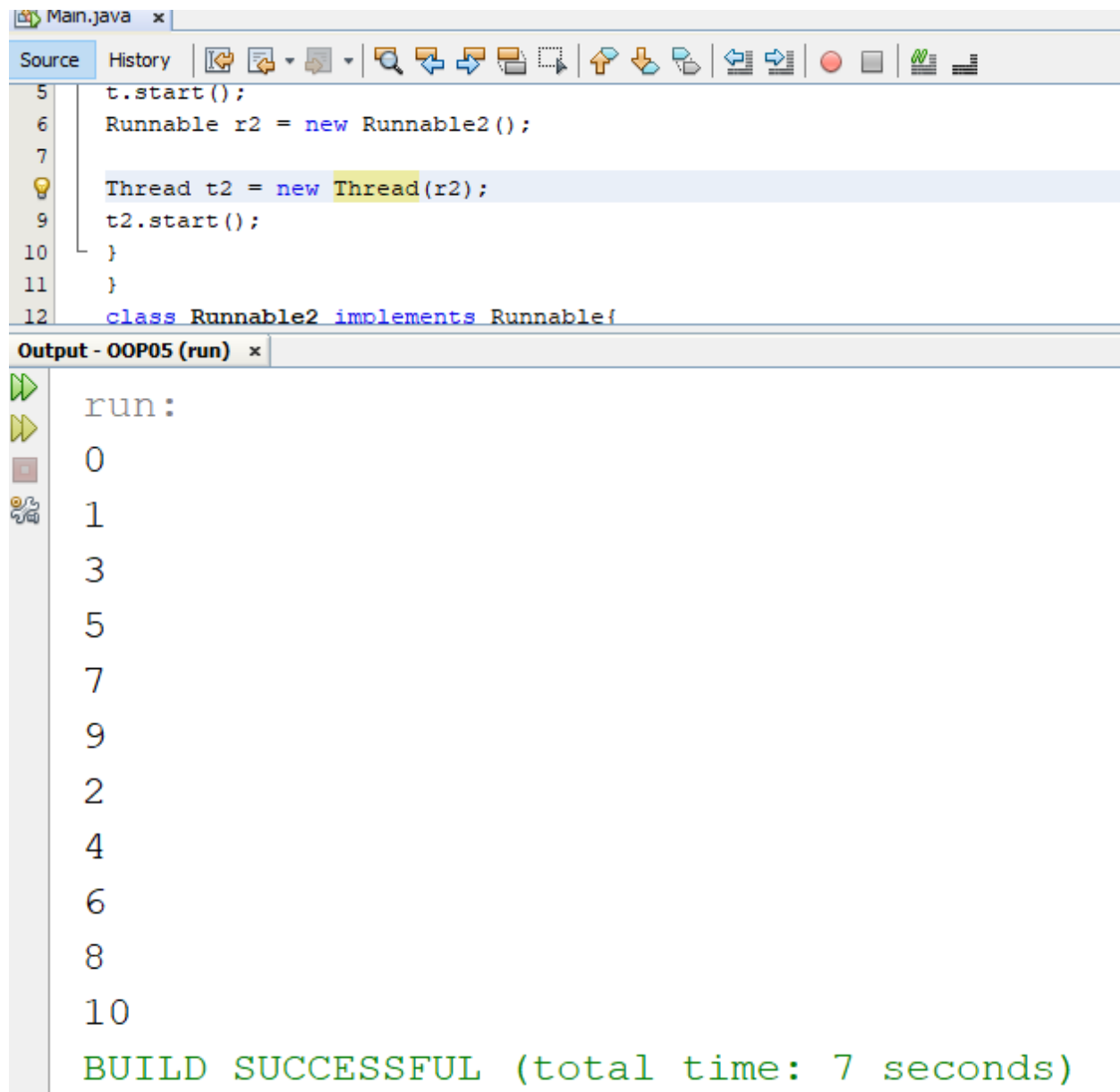
## Practical:20

**Write a program to create two threads, one thread will print odd numbers and second thread will print even numbers between 1 to 20 numbers.**

```
public class Main {
    public static void main(String[] args) {
        Runnable r = new Runnable1();
        Thread t = new Thread(r);
        t.start();
        Runnable r2 = new Runnable2();
```

```
Thread t2 = new Thread(r2);
t2.start();
}
}
class Runnable2 implements Runnable{
public void run(){
for(int i=0;i<11;i++){
if(i%2 == 1)
System.out.println(i);
}
}
}
class Runnable1 implements Runnable{
public void run(){
for(int i=0;i<11;i++){
if(i%2 == 0)
System.out.println(i);
}
}
}
}
```

Output:-



The screenshot shows an IDE window with a tab titled 'Main.java'. The 'Source' editor displays the following Java code:

```
5 t.start();
6 Runnable r2 = new Runnable2();
7
8 Thread t2 = new Thread(r2);
9 t2.start();
10 }
11 }
12 class Runnable2 implements Runnable{
```

Below the code editor is the 'Output - OOP05 (run)' window. It shows the output of the program, which is a sequence of numbers from 0 to 10, followed by a green message: 'BUILD SUCCESSFUL (total time: 7 seconds)'.

## Practical:21

**Create a class called Student. Write a student manager program to manipulate the student information from files by using FileInputStream and FileOutputStream.**

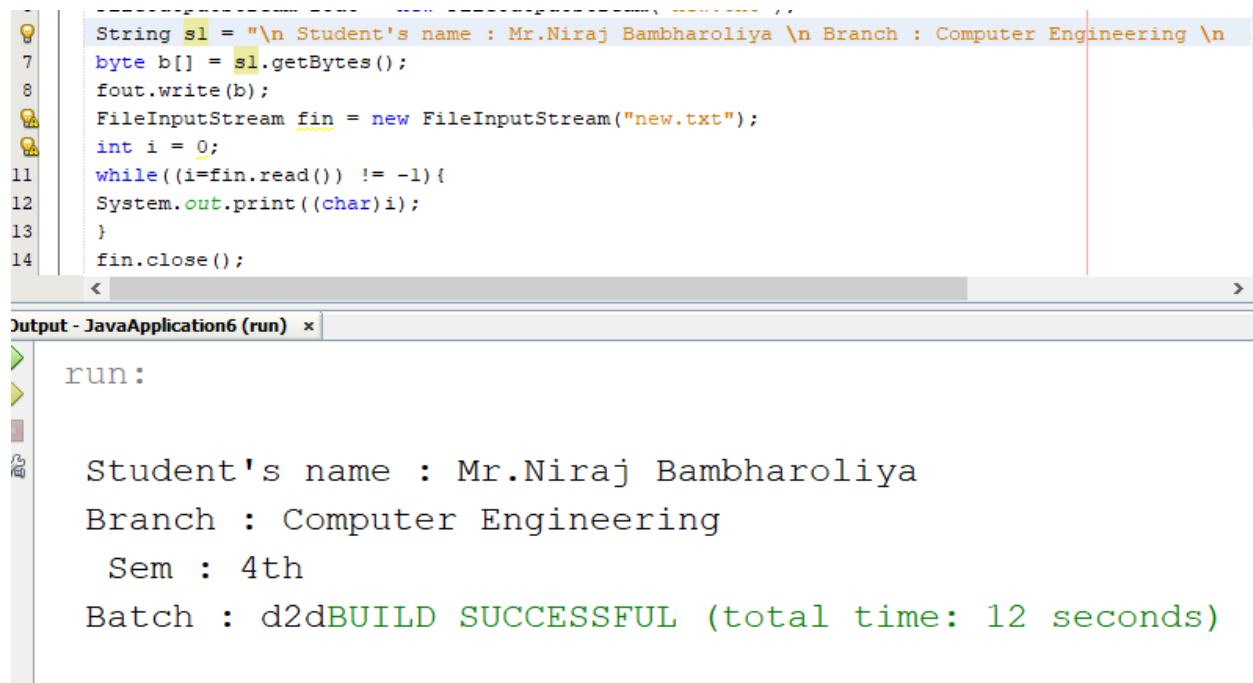
```
import java.io.FileInputStream;
import java.io.FileOutputStream;
public class student {
```

```

public static void main(String arg[])throws Exception{
    FileOutputStream fout = new FileOutputStream("new.txt");
    String s1 = "\n Student's name : Mr.Niraj Bambharoliya \n Branch : Computer Engineering \n
    Sem : 4th \n Batch : d2d";
    byte b[] = s1.getBytes();
    fout.write(b);
    FileInputStream fin = new FileInputStream("new.txt");
    int i = 0;
    while((i=fin.read()) != -1){
        System.out.print((char)i);
    }
    fin.close();
}
}

```

Output:-



```

String s1 = "\n Student's name : Mr.Niraj Bambharoliya \n Branch : Computer Engineering \n
7 byte b[] = s1.getBytes();
8 fout.write(b);
FileInputStream fin = new FileInputStream("new.txt");
int i = 0;
11 while((i=fin.read()) != -1){
12     System.out.print((char)i);
13 }
14 fin.close();

```

Output - JavaApplication6 (run) x

```

run:

Student's name : Mr.Niraj Bambharoliya
Branch : Computer Engineering
Sem : 4th
Batch : d2dBUILD SUCCESSFUL (total time: 12 seconds)

```

## Practical:22

**Define the student manager program to manipulate the student information from files by using the BufferedReader and BufferedWriter.**

```
import java.io.BufferedReader;
```



```

import java.io.BufferedReader;
import java.io.FileReader;
import java.io.FileWriter;
import java.io.FileInputStream;
public class P22
{
public static void main (String args[]) throws Exception
{
    BufferedReader bufRead = new BufferedReader (new FileReader("niraj.txt"));
    BufferedWriter bufWrite = new BufferedWriter (new FileWriter("newtriinfo.txt"));
    int i;
    do{
        i=bufRead.read();
        if(i !=1)
        {
            if(Character.isUpperCase((char)i))
            {
                bufWrite.write(Character.toLowerCase((char)i));
            }
            else{ bufWrite.write((char)i); }
        }
    }while (i != -1);
    bufRead.close();
    bufWrite.close();
    System.out.println("Written sucessful...");
    System.out.println("Fetching the data...");
    try{
        FileInputStream fin= new FileInputStream("newtriinfo.txt");
        i=0;
        while ((i= fin.read()) != -1){
            System.out.print((char)i);
        }
        fin.close();
    }catch(Exception e)
    { System.out.println(e); }

}
}

```

Output:

Output - JavaApplication6 (run) x

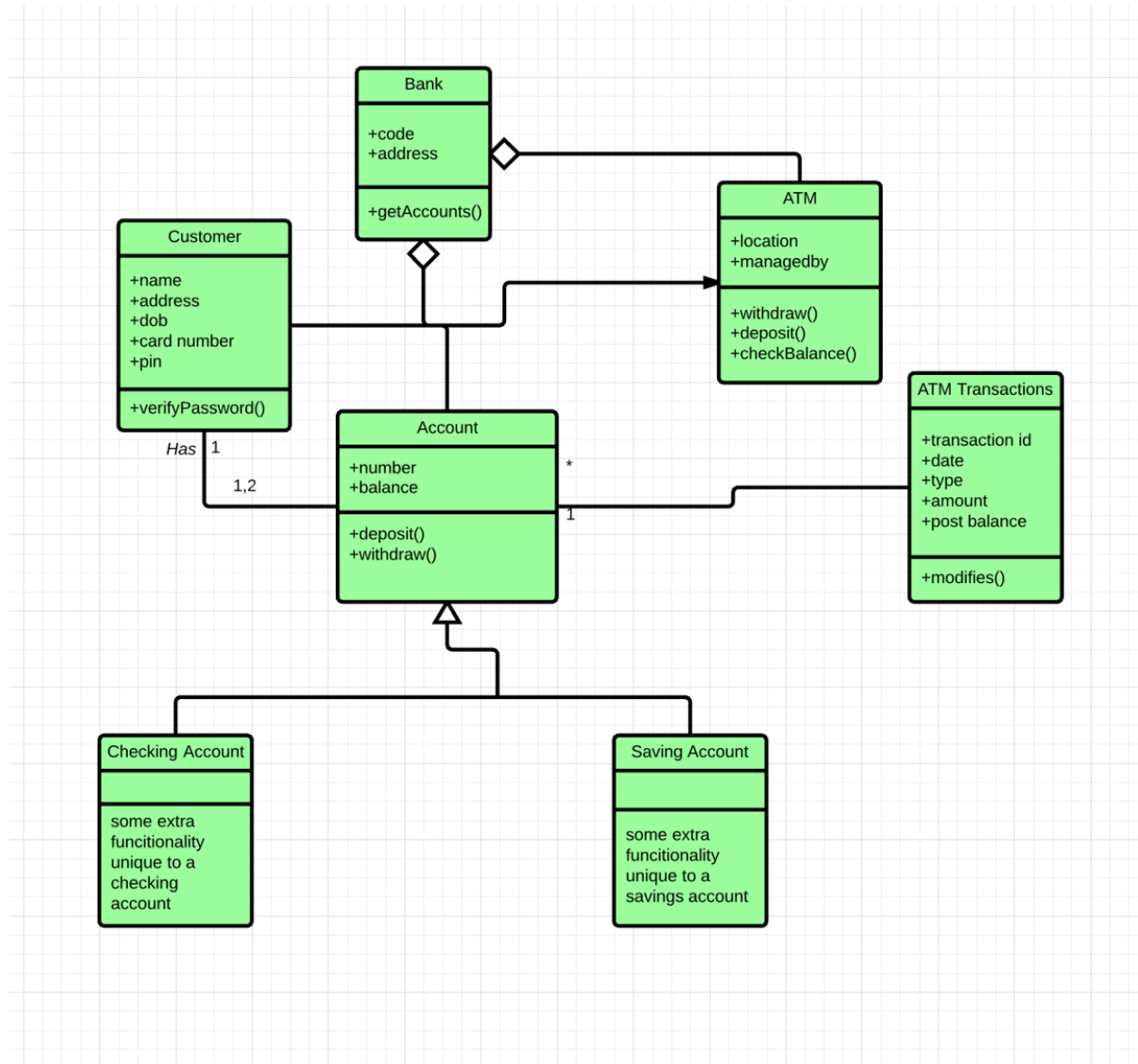
```
✓✓BUILD SUCCESSFUL (total time: 4 seconds)
```

```
graph LR
    HG[Hotel Guest]
    R[Receptionist]
    SA((Search Availability))
    BR((Book Room))
    CR((Cancel Reservation))
    CI((Check In))
    CO((Check Out))
    FS((Food Serving))

    HG --- SA
    HG --- BR
    HG --- CI
    HG --- CO
    HG --- FS
    R --- BR
    R --- CI
    R --- CO
    CR -- "<<include>>" --> BR
```

The diagram illustrates the interactions between two actors, a Hotel Guest and a Receptionist, and six use cases: Search Availability, Book Room, Cancel Reservation, Check In, Check Out, and Food Serving. The Hotel Guest is connected to all six use cases. The Receptionist is connected to Book Room, Check In, and Check Out. Additionally, the Cancel Reservation use case includes the Book Room use case, indicated by an arrow labeled <<include>>.

24) Draw class diagram of bank management system.



25) Prepare a Sequence diagram for issuing a book in the library management system.

