# INDUS UNIVERSITY

Indus Institute of Technology and Engineering


Lab manual of

Core Java Programming

(CE0421)

# Indus Institute of Technology and Engineering

Branch: Computer Science & Engineering

Semester: IV

Subject: Core Java Programming (CE0421)

## INDEX

| | | | | | |
|---|---|---|---|---|---|
| | 4.3 | The employee list for a company contains employee code, name, designation and basic pay. The employee is given HRA of 10% of the basic and DA of 45% of the basic pay. The total pay of the employee is calculated as Basic pay +HRA+DA. Write a class to define the details of the employee.Write a constructor to assign the required initial values. Add a method to calculate HRA, DA and Total pay and print them out. Write another class with a main method. Create objects for three different employees and calculate the HRA, DA and total pay. | 12 | | |
| 5 | 5.1 | Write a program which defines base class Employee having three data members, namely name[30], emp_numb and gender and two methods namely input_data() and show_data(). Derive a class SalariedEmployee from Employee and adds a new data member, namely salary. It also adds two member methods. | 15 | | |
| | 5.2 | WAP that illustrates method overriding. Class A3 is extended by Class B3.Each of these classes defines a hello(string s) method that outputs the string "A3: Hello From" or "B3: Hello From" respectively. Use the concept Dynamic Method Dispatch and keyword super. | 19 | | |
| 6 | 6.1 | Write an abstract class shape, which defines abstract method area. Derive class circle from shape. It has data member radius and implementation for area function. Derive class Triangle from shape. It has data members height, base and implementation for area function. Derive class Square from shape. It has data member side and implementation for area function. In main class, use dynamic method dispatch in order to call correct version of method. | 21 | | |
| | 6.2 | Create an interface Shape2D which declares a getArea() method. Point 3D contains coordinates of a point. The abstract class Shape declares abstract display() method and is extended by Circle class. it implements the Shape2D interface. The Shapes class instantiates this class and exercises its methods. | 24 | | |

| | | | | | |
|---|---|---|---|---|---|
| 7 | 7.1 | Create a package “employee” and define a Class Employee havingthreedatamembers,name, emp_num,andgenderandtwomethodsnput_ dataanshow_data. Inherit class SalariedEmployee from this class and keep it in package “employee”. Add new variable salary and methods allowance(ifemalehra=0.1*salaryelse.09*salary.DA= 0.05*salary) and increment (salary=salary+0.01 * salary). Calculate gross salary in main class defined in the same package. | 27 | | |
| 8 | 8.1 | WAP using try catch block. User should enter two command line arguments. If only one argument is entered then exception should be caught. In case of two command line arguments, if fist is divided by second and if second command line argument is 0 then catch the appropriate exception | 31 | | |
| | 8.2 | Define an exception called “NoMatchException” that is thrown when a string is not equal to “India”. Write a program that uses**8**this exception. **.** | 34 | | |
| 9 | 9.1 | The program to creates and run the following three threads. The first thread prints the letter ‘a’ 100 times. The second thread prints the letter ‘b’ 100 times. The third thread prints the integer 1 to 100 | 37 | | |
| | | Write the thread p**9**rogram -1using Runnable interface | 40 | | |
| 10 | 10.1 | Write a program that takes two files names (source and destination) as command line argument. Copy source file’s content to destination file. Use character stream class.Also do using byte stream and buffer strem | 42 | | |
| | 10.2 | Write a program which generates random integers and stores them in a file named “rand.dat”. The program then reads the integers from the file and displays on the screen | 50 | | |

| | | | | | |
|---|---|---|---|---|---|
| 11 | 11.1 | Write the program that demonstrate the use of Stack, Vector and ArrayList classes. | 53 | | |
| 12 | 12.1 | Write a Network program that client sends the data as radius of circle to server and server received that data and sendthere**1**sultant**1**areaofcircletorequestedclient. | 56 | | |
| 13 | 13.1 | Write a program to count occurrence of character in a string. | 60 | | |

## Practical -1

### 1.1 Write a program to display “Welcome to java World”.

File Name: demo.java

```
class demo
{
        public static void main(String[] args)
    {
        System.out.println(“Welcome to java World ”);

    }
}
```

Output:

**1.2 Write a program to find whether the number is prime or not.**

File Name: demo.java

```
class demo
    {
        public static void main(String [] args)
    {
        int num=29; Boolean flag=false; for(int I=2;i<=num/2;i++)
    {
         if(num% I==0)
         {
             flag=true; break;
         }
    }
   if (!flag)
      System.out.println (num+"is a prime number .");

    else
   System.out.println (num+"is not a prime number.");
    }
  }
```

Output:

**1.3. (a) Write a program to find a greater number among given three number using aternary operator.**

File Name: MaximumNum.java

```
class MaximumNum
{
        public static void main(String args[])
        {
        int a=10,
        b=25,c=15, max;
        max= (a>b)?
        (a>c?a:c):
        (b>c?b:c);
        System.out.println("Maximum number among "+a+", "+b+ "and"+c+ “is"+max);
        }
}
```

Output:

**(b) Write a program to find a greater number among give three numbers using a nestedif statement.**

File Name: main.java

```
class main
 {
    public static void main(String[] args)
        {
         double n1=-4.5, n2=3.9,
     n3=2.5;
if (n1>=n2&8n1>=n3)
System.out.println(n1 +"is the largest number.”);
else if (n2>=n1&&n2>=3)
System.out .println (n2+ "is the largest
number.”);
 else
System.out .println(n3 + "is the largest number.”);
}
}
```

Output:

```
/Users/dharitapatel/Library/Java/JavaVirtualMachines/openjdk-19.0.2/Contents/Home/bin/java -javaagent:/Applications/IntelliJ IDEA CE.app/Contents/lib/idea_rt.jar=4955
3.9is the largest number.

Process finished with exit code 0
```

**1.4 Write a program to print the fibonacci series**

File Name: main.java

```
class main
 {
     public static void main(String args[]) { int
    n1=0, n2 = 1, n3, i, count = 10;
    System.out.print(n1 + "" + n2) ;
    for (i = 2; i < count; i++)
    {
            n3 = n1 + n2;
            System.out.print("'+ n3) ;
             n1 = n2; n2 n3;
     }
}
```

Output:

## Practical -2

### 2.1 Write a program to find the average of n numbers stored in an array.

File Name: main.java

```
class main
{  public static void main(String[] args)
        {
                double[] numArray={20,30,40,50, 60};
                double sum=0.0;
                for (double num : numArray)\
                {
                        Sum+=num;
                }
                 10 double average = sum/numArray. length;
                System.out.average("The average is %.2f",average);
        }
}
```

Output:

**Practical-3**

**3.1 Write a program to replace a string with other substring in the given string.**

File Name: main.java

```
class main {


public static void main(String args[])

{

String a = "Welcome to Earth”;

String b = “Mars";

String c = a.replace ( target: "Earth", replacement: “Mars");

System.out.printin("String Before Replace=" + a) ;

System.out.printin("String After Replace"+ C: );
String d = "Hello Apple”;

String e = “Apple";

String f = “Samsung";

String g = d.replace (e, f);

System.out.printin(g) ;

                }
}
```

Output:

```
/Users/dharitapatel/Library/Java/JavaVirtualMachines/openjdk-19.0.2/Contents/Home/bin/java -javaagent:/Applications/IntelliJ IDEA CE.app/Contents/lib/idea_rt.jar=4961
The average is:40.00
Process finished with exit code 0
```

**3.2 Write a program that to sort given strings in alphabetical order.**

File Name: s.java

```
import java.util.*;
class s
{  public static void main(String Args [])
        {
        Scanner ob=new Scanner (System in); String
        originalString = ob.nextLine();
        char[]charArray=originalString.toCharArray()
        ;
        Arrays. sort (charArray);
        String sortedString = new String (charArray);
        System.out.printin(sortedString);
        }
}
```

Output:

```
/Users/dharitapatel/Library/Java/JavaVirtualMachines/openjdk-19.0.2/Contents/Home/bin/java -javaagent:/Applications/IntelliJ IDEA CE.app/Contents/lib/idea_rt.jar=4963
String Before Replace=Welcome to Earth
String After ReplaceWelcome to Mars
Hello Samsung

Process finished with exit code 0
```

**3.3 Create a string buffer with some default using append any string to I Position using original string and display the modified string . Also display the reverse ofthe modified string .**

```
class main
{

public static void main (String[] args)

StringBuffer a=new StringBuffer("Hello I am Iron Man.”);

System.out.println("Original String: “+a);

a.    insert ( offset: 5, str: " Thanos,”);
b.
System.out.printin("Modified String: "+a); a.reverse);

System. out.println(“Reverse of modified string: “+a);

}
}
```

Output:

**Practical - 4**

**4.1 Write a program that declares a class name person .it should be in instance variables to record name, age and salary.Use new operator to create a person object. Set and display its instance variable.**

File Name: Person.java

```
class Person
{
String name;     int age;    float salary;
voidset()
{     name=“Sunny";    age=21;
          salary=20000;
  }
  void display()
{
          System.out.println(“Name:”+name);
          System.out.println(“Age:"+age);
          System.out.println("Salary:"+salary);
}
   public static void main(String[] args)
{
          Person p = new Person();
            p.set();
            p.display();
  }
}
```

Output:

```
/Users/dharitapatel/Library/Java/JavaVirtualMachines/openjdk-19.0.2/Contents/Home/bin/java -javaagent:/Applications/IntelliJ IDEA CE.app/Contents/lib/idea_rt.jar=4966
Name:Dharita
Age:21
Salary:20000.0

Process finished with exit code 0
```

**4.2 Write a program to add a constructor to the person class developed above.**

File Name: Person.java

```
class Person
{
        String name;  int age;  float salary;
Person()
{
        System.out.println("Person Details\n");
}  void
set()
 {
         name="Sunny";     age=21;     salary=20000;
}

void display()
{
         System.out.println("Name:"+name);
        System.out.println("Age:"+age);
        System.out.println("Salary:"+salary);
}
  public static void main(String[] args)
{
         Person p = new Person();
            p.set();
        p.display();
  }
}
```

Output:

```
/Users/dharitapatel/Library/Java/JavaVirtualMachines/openjdk-19.0.2/Contents/Home/bin/java -javaagent:/Applications/IntelliJ IDEA CE.app/Contents/lib/idea_rt.jar=4967
Person Details

Name:Dharita
Age:21
Salary:20000.0

Process finished with exit code 0
```

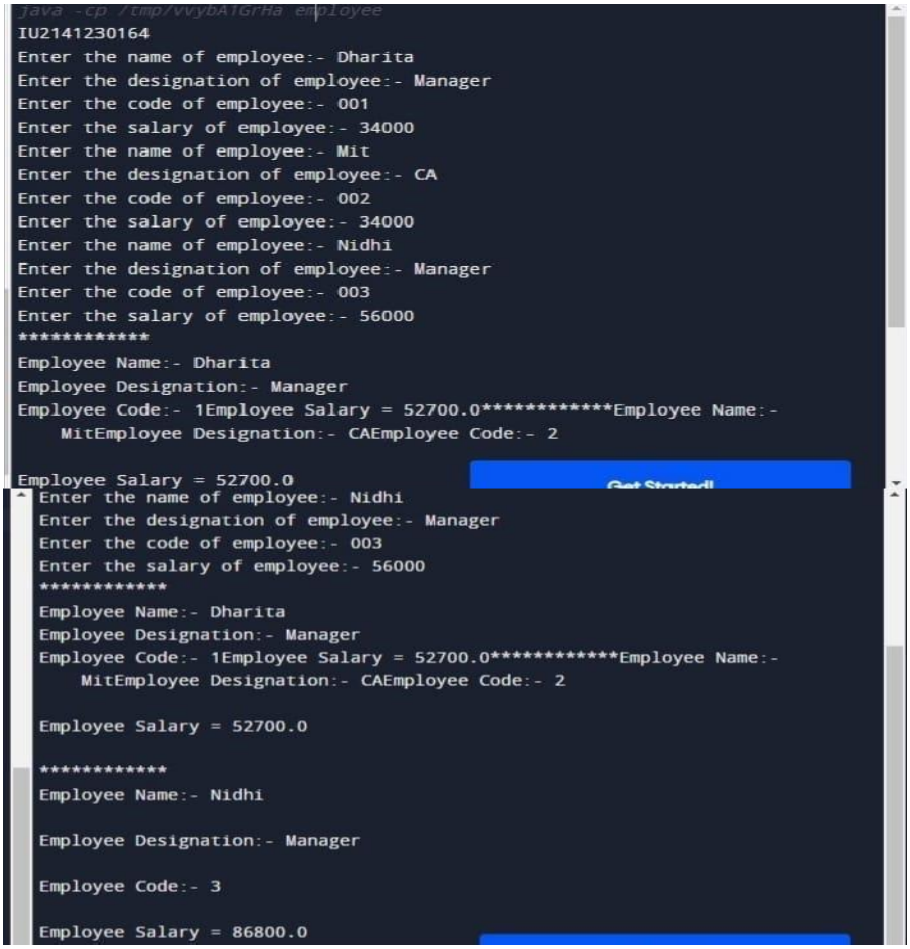
**4.3 The employee list for a company contains employee code, name, designation and basic pay. The employee is given HRA of 10% of the basic and DA of 45% of the basic pay. The total pay of the employee is calculated as Basic pay+HRA+ DA. Write a class to define the details of the employee. Write a constructor to assign the required initial values. Add a method to calculate HRA, DA and Total pay and print them out. Write another class with a main method. Create objects for three different employees and calculate the HRA, DA and total pay**

File Name: demo.java

```
import java.util.*;
class demo
{

        String name, dec;
        int code, salary;
        demo()
        {

                Scanner sc = new Scanner(System.in);

                System.out.print("\nEnter the name of employee:- ");
                name = sc.nextLine();

                System.out.print("\nEnter the designation of employee:- ");
                dec = sc.nextLine();

                System.out.print("\nEnter the code of employee:- ");
                code = sc.nextInt();

                System.out.print("\nEnter the salary of employee:- ");
                salary = sc.nextInt();
        }

        void calculate()
```

```
        {
                float hra, da, totalpay;

                hra = salary*10/100;
                da= salary*45/100;
                totalpay =salary+ hra + da;
                System.out.println("\n******");
                System.out.println("\nEmployee Name:- " + name);
                System.out.println("\nEmployee Designation:- " + dec);
                System.out.println("\nEmployee Code:- " + code);
                System.out.println("\nEmployee Salary = " + totalpay);
        }
    }


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

      System.out.println("\nIU2141230164");
                demo obj1 = new demo();
        demo obj2 = new demo();
        demo obj3 = new demo();
        obj1.calculate();
        obj2.calculate();
                obj3.calculate();
        }
}
```

Output:


java -cp /tmp/vvybAIGrHa employee
IU2141230164
Enter the name of employee:- Dharita
Enter the designation of employee:- Manager
Enter the code of employee:- 001
Enter the salary of employee:- 34000
Enter the name of employee:- Mit
Enter the designation of employee:- CA
Enter the code of employee:- 002
Enter the salary of employee:- 34000
Enter the name of employee:- Nidhi
Enter the designation of employee:- Manager
Enter the code of employee:- 003
Enter the salary of employee:- 56000
**********
Employee Name:- Dharita
Employee Designation:- Manager
Employee Code:- 1Employee Salary = 52700.0**********Employee Name:-
MitEmployee Designation:- CAEmployee Code:- 2
Employee Salary = 52700.0
Enter the name of employee:- Nidhi
Enter the designation of employee:- Manager
Enter the code of employee:- 003
Enter the salary of employee:- 56000
**********
Employee Name:- Dharita
Employee Designation:- Manager
Employee Code:- 1Employee Salary = 52700.0**********Employee Name:-
MitEmployee Designation:- CAEmployee Code:- 2
Employee Salary = 52700.0
**********
Employee Name:- Nidhi
Employee Designation:- Manager
Employee Code:- 3
Employee Salary = 86800.0

**Practical 5**

**5.1 Write a program which defines base class Employee having three data members, namely name[30], emp_numb and gender and two methods namely input_data() and show_data(). Derive a class SalariedEmployee from Employee and adds a new data member, namely salary. It also adds two member methods, namely allowance (if gender is female HRA=0.1 *salary else 0.09* salary. DA=0.05*salary) and increment (salary= salary+0.1*salary). Display the gross salary in main class. (Tip: Use super to call base class's constructor0).**

File Name: SalariedEmployee.java

```
import java.util.*; class Employee
{
                Scanner s = new Scanner(System.in);

                String name;
                int emp_numb;
               char gender;

               void input_data()

        {

                  System.out.print("Enter name: ");
                  name = s.nextLine();

                 System.out.print("Enter gender(M/F): ");

                  gender = s.next().charAt(0);

                 while((gender != 'M') && (gender != 'F') )

                {

                        System.out.print("please enter gender again: ");

                        gender = s.next().charAt(0);

                }

               System.out.print("Enter emp_number: ");
```

```
            emp_numb = s.nextInt();
        }
         void display_data()
        {
            System.out.println("Employee name: "+name);
            System.out.println("Employee number: "+emp_numb);
            System.out.println("Employee gender: "+gender);
        }
}
 class SalariedEmployee extends Employee
{
         double salary, hra, da;
          SalariedEmployee()
         {
                System.out.println("Enter salary: ");
                 salary = s.nextFloat();
                 s.nextLine();
         }
         void allowance()
         {
                  if(gender == 'F')
                  {
                          hra = 0.1 * salary;
                   }
                 else
                {
```

```
                    hra = 0.09 * salary;
                }
                da = 0.05 * salary;
                salary = salary + hra + da;
}
void increment()
 {
                salary = salary + (0.1) * salary ;
                System.out.println();
 }
public static void main(String[] args)
 {
                SalariedEmployee emp = new SalariedEmployee();
                emp.input_data();
               emp.display_data();
                emp.allowance();
                emp.increment();
               System.out.println("Gross salary = " + emp.salary);
 }
}
```

Output:

```
/Users/dharitapatel/Library/Java/JavaVirtualMachines/openjdk-19.0.2/Contents/Home/bin/java -
Enter salary:
16000
Enter name: Dharita
Enter gender(M/F): F
Enter emp_number: 164
Employee name: Dharita
Employee number: 164
Employee gender: F

Gross salary = 20240.0

Process finished with exit code 0
```

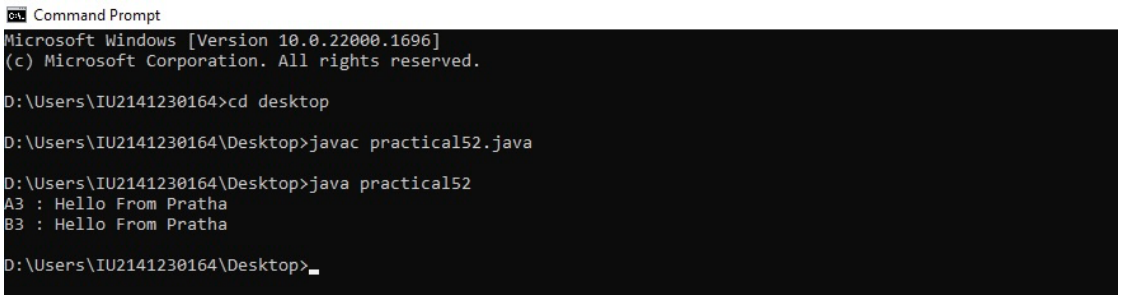
**5.2 WAP that illustrates method overriding. Class A3 is extended by Class B3. Each of these classes defines a hello(string s) method that outputs the string "A3: Hello From" or "B3: Hello From" respectively. Use the concept Dynamic Method Dispatch and keyword super.**

File Name: practical52.java

```
import java.util.*;

class A3
{
        String S;
        A3(String s)
        {
                hello(s);
        }
        public void hello(String s)
        {
                S=s;
                System.out.println("A3 : Hello From "+S);
        }
}
class B3 extends A3
{
        B3(String s)
        {
                super(s);
        }
      public void hello(String s)
```

```
        {
          S=s;
          System.out.println("B3 : Hello From "+S);
         }
 }
class practical52
{
        public static void main(String[] args)
{

        A3 a = new A3("Pratha");
         A3 b = new B3("Pratha");

}
}
```

Output:



**Practical - 6**

**6.1 Write an abstract class shape, which defines abstract method area. Derive class circle from shape. It has data member radius and implementation for area function. Derive class Triangle from shape. It has data members height, base and implementation for area function. Derive class Square from shape. It has data member side and implementation for area function. In main class, use dynamic method dispatch in order to call correct version of method.**

File Name: pr61.java

```
import java.util.*;

abstract class Shape

{

        Scanner input = new Scanner(System.in);

        abstract void area();

 }

class Circle extends Shape

{

         float radius;

         void area()

         {

                  System.out.print("Enter Radius: ");
                  radius = input.nextFloat();
                 System.out.println("Area of Circle: "+(3.14*radius*radius));

          }

}
```

```
class Triangle extends Shape
 {
        float height,base;
        void area()
        {
                System.out.print("Enter Height: ");
                height = input.nextFloat();
                System.out.print("Enter base: ");
                base = input.nextFloat();
                System.out.println("Area of Triangle: "+((height*base)/2));
         }
}
class Square extends Shape
{
        float side;
        void area()
         {
                System.out.print("Enter Side: ");
                side = input.nextFloat();

                System.out.println("Area of Square: "+(side*2));
          }
 }
class pr61
```

```
{
        public static void main(String[] args)
        {
            System.out.println("IU2141230142");
            Circle cir = new Circle();
            cir.area();
           Triangle tri = new Triangle();
           tri.area();
           Square sq = new Square();
           sq.area();
        }
}
```

**6.2 Create an interface Shape2D which declares a getArea() method. Point 3D contains coordinates of a point. The abstract class Shape declares abstract display() method and is extended by Circle class. it implements the Shape2D interface. The Shapes class instantiates this class and exercises its methods.**

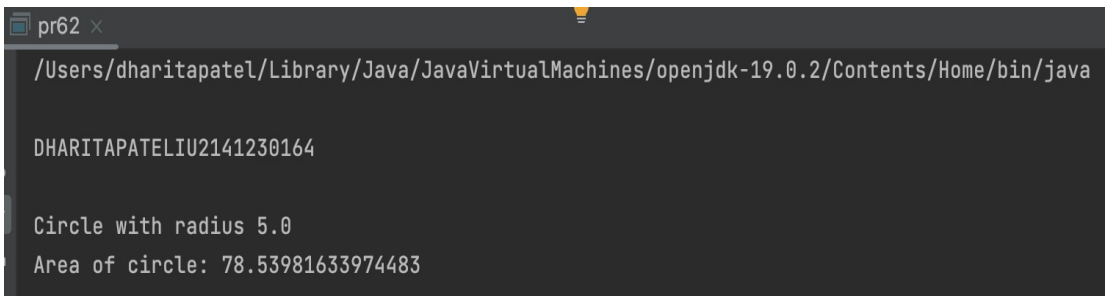
File Name: Shapes.java

```
interface Shape2D
 {
        void getArea();
 }
abstract class Shape
{
        abstract void display();
}
class Point3D
{
        double x,y,z;
        Point3D(double x,double y,double z)
       {
            this.x = x;
            this.y = y;
            this.z = z;
       }
```

```
}

class Circle extends shape implements Shape2D

{

        Point3D CenterPoint, OtherPoint;

        double area;

        Circle(Point3D P1, Point3D P2)

        {

             CenterPoint = P1;

            OtherPoint = P2;

         }

public void getArea()

 {

              double d1 = CenterPoint.x - OtherPoint.x;

             double d2 = CenterPoint.y - OtherPoint.y;

             double d = (d1*d1) + (d2*d2);
             double radius = Math.sqrt(d);

              area = Math.PI*radius*radius;

 }

public void display()

{
```

```
 System.out.println("\nGiven points-> Center Point: ("+CenterPoint.x+",       "+CenterPoint.y+",
"+CenterPoint.z+") , Other Point: ("+OtherPoint.x+", "+OtherPoint.y+", "+OtherPoint.z+")\nArea
for a circle having given points: "+area+" __\ n");

   }

}

class Shapes

{

public static void main(String args[])

{

                    Point3D point1 = new Point3D(6,5,7);

                   Point3D point2 = new Point3D(8,3,5);

                   Circle obj = new Circle(point1 , point2);

                    obj.getArea();

                   obj.display();

 }

}
```

Output:



## Practical – 7

**7.1 Create a package "employee" and define a Class Employee having three data members, name, emp_num, and gender and two methods- input_data and show_data. Inherit class SalariedEmployee from this class and keep it in package "employee". Add new variable salary and methods allowance (if female hra=0.1*salary else 0.09* salary. DA= 0.05*salary) and increment (salary=salary+0.01 * salary). Calculate gross salary in main class defined in the same package.**

File1 Name: SalariedEmployee.java

```
package employee;

import java.util.*;

class Employee
{

        public Scanner s = new Scanner(System.in);

        public String name;
        public int emp_numb;
       public char gender;

       public void input_data()

       {

              System.out.println("Enter name: ");
             name = s.nextLine();

             System.out.println("Enter gender(M/F): ");

            gender = s.next().charAt(0);

            while((gender != 'M') && (gender != 'F') )

           {

                System.out.println("please enter gender again: ");
```

```
                    gender = s.next().charAt(0);

                }
                System.out.println("Enter emp_number: ");

                emp_numb = s.nextInt();

 }
  public void display_data()

        {

                 System.out.println("Employee name: "+name);

                 System.out.println("Employee number: "+emp_numb);

                 System.out.println("Employee gender: "+gender);

         }

}

public class SalariedEmployee extends Employee

 {

                  public double salary, hra, da;

                 public SalariedEmployee()
                {

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

                         salary = s.nextFloat();
                        s.nextLine();

                }
                 public void allowance()

                {
```

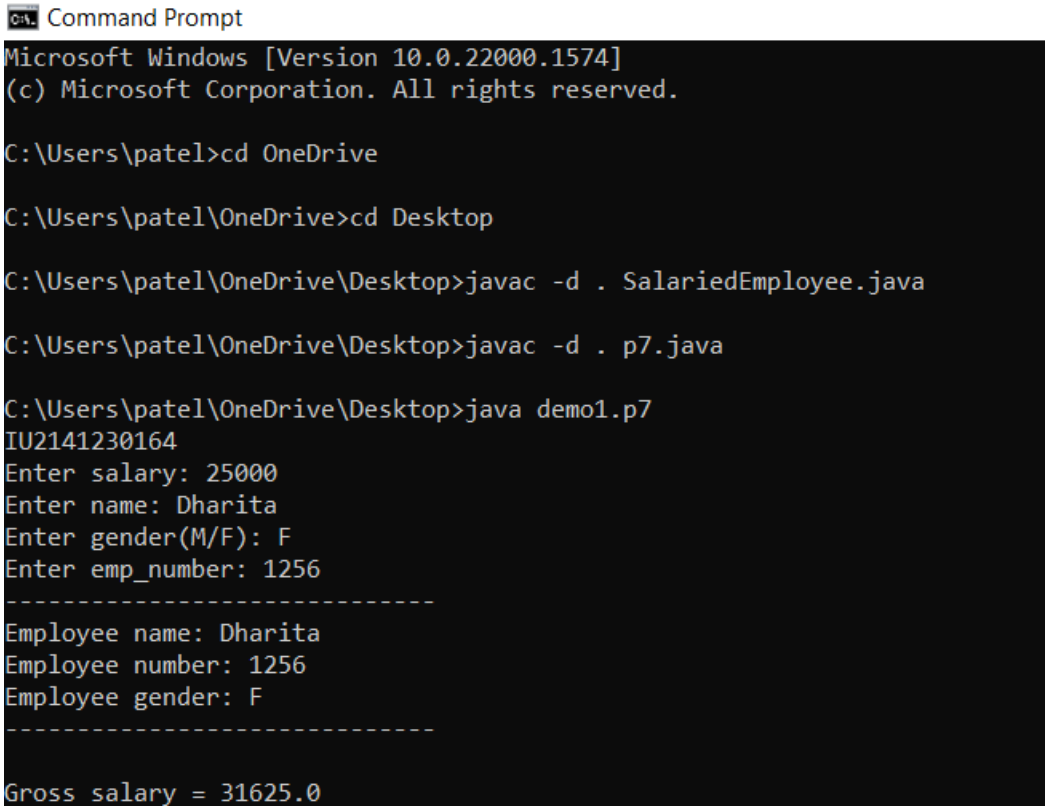
```
                    if(gender == 'F')
                    {
                            hra = 0.1 * salary;
                     }
                    Else
                      {
                    hra = 0.09 * salary;
                     }
                    da = 0.05 * salary;
                    salary = salary + hra + da;
              }
              public void increment()
              {
                    salary = salary + (0.1) * salary ;
                    System.out.println();
              }
}
```

File2 Name: main.java

```
package demo;
import java.util.*;
import employee.SalariedEmployee;
```

```
class main
{
        public static void main(String[] args)
        {
                SalariedEmployee emp = new SalariedEmployee();
               emp.input_data();
               emp.display_data();
               emp.allowance();
                emp.increment();
                System.out.println("Gross salary = "+emp.salary);
        }
}
```

Output:



## Practical - 8

**8.1 WAP using try catch block. User should enter two command line arguments. If only one argument is entered then exception should be caught. In case of two command line arguments, if fist is divided by second and if second command line argument is 0 then catch the appropriate exception.**

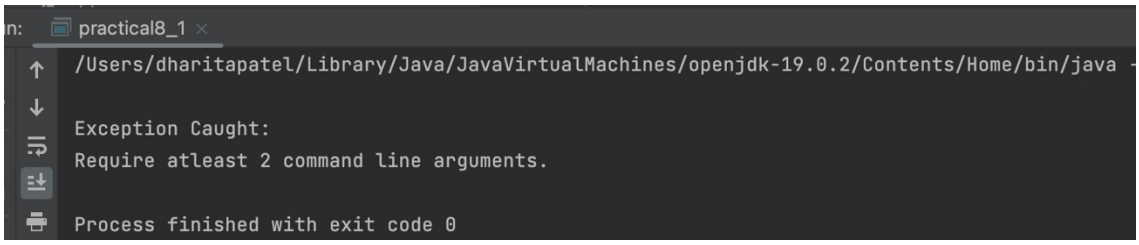
File Name: practical8_1.java

```
class NewException extends Exception
 {
        String m;
        NewException(String message)
        {
                m = message;
        }
        void printmsg()
        {
                 System.out.println(m);
        }
}
class practical8_1
{
        public static void main(String args[])
        {
                int num[] = new int[args.length];
```

```
        try
          {
                if(args.length<2)
    {
          throw new NewException("Require atleast 2 command line arguments.");
    }
    for(int i=0; i<args.length; i++)
    {
     num[i] = Integer.parseInt(args[i]);
}
 int a = (int)num[0];
int b = (int)num[1];
double div = a/b;
System.out.println("\nDivision successful. No exceptions found.\n"+a+"/"+b+" =
"+div);
}
catch(NewException e)
 {
System.out.println("\nException Caught: ");
e.printmsg();
 }
catch(NumberFormatException e2)
{
```

```
            System.out.println("\nException Caught: ");
            System.out.println("Require Integer arguments.");
}
 catch(ArithmeticException e3)
{
            System.out.println("\nException Caught: ");
           System.out.println("Division by Zero.");
}
}
}
```

Output:



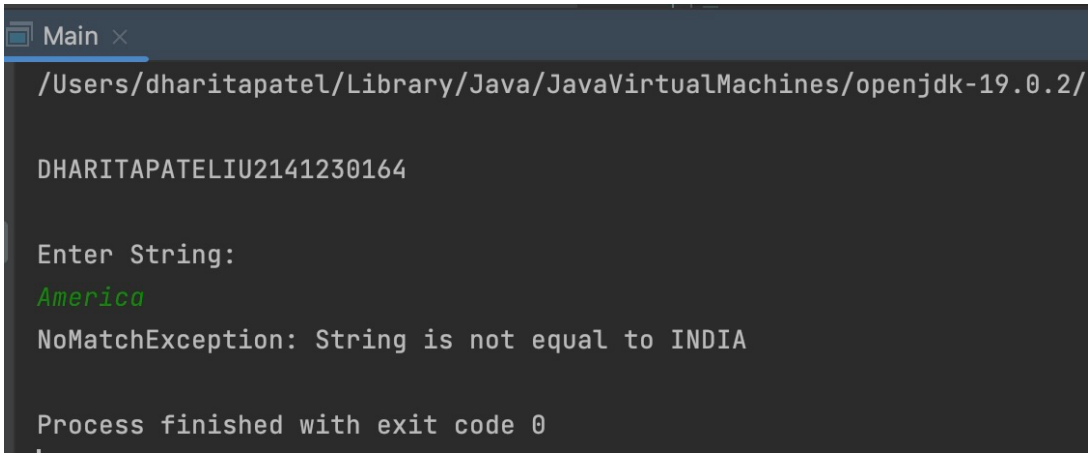
**8.2 Define an exception called "NoMatchException" that is thrown when a string is not equal to "India". Write a program that uses this exception.**

File Name: practical8_2.java

```
class NoMatchException extends Exception
{
    String m;
    NoMatchException(String message)
    {
        m = message;
    }
    void printmsg()
    {
        System.out.println(m); }
    }
class practical8_2
{
public static void main(String args[])
 {
        if(args.length!=1)
{
        System.out.println("\nEnter only one String argument");
}
```

```
            Else
 {
try
            {
               if(args[0].compareTo("India") == 0)
            {
               System.out.println("\nString is equal to 'India'.");
             }
else
           {
                throw new NoMatchException("Arguments is not equal to 'India'.");
           }
 }
catch(NoMatchException e)
           {
                System.out.println("\nException Caught: ");
               e.printmsg();
           }
        }
    }
}
```

Output:


Main
/Users/dharitapatel/Library/Java/JavaVirtualMachines/openjdk-19.0.2/
DHARITAPATELIU2141230164
Enter String:
America
NoMatchException: String is not equal to INDIA
Process finished with exit code 0

## Practical - 9

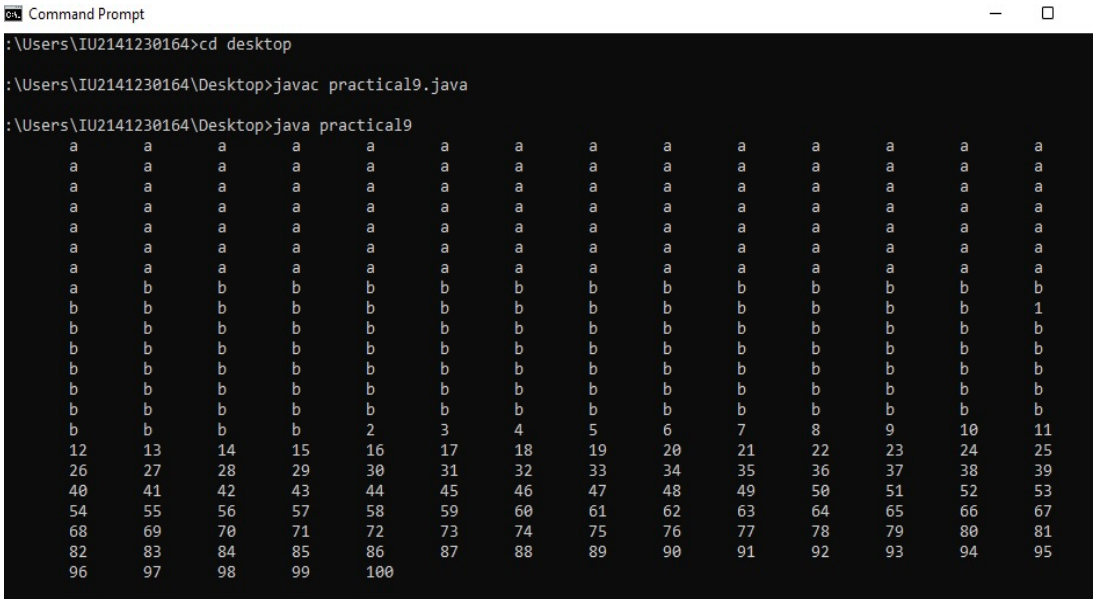
**9.1 The program to creates and run the following three threads. The first thread prints the letter 'a' 100 times. The second thread prints the letter 'b' 100 times. The third thread prints the integer 1 to 100.**

File Name: practical9.java

```
class A extends Thread
 {
          public void run()
         {
                 for(int i=0; i<100; i++)
                {
                        System.out.print("a \t");
                 }
          }
}
class B extends Thread
{
          public void run()
 {
        for(int i=0; i<100; i++)
         {
                System.out.print("b \t");
```

```
        }
    }
}
class C extends Thread
{
public void run()
 {
            for(int i=0; i<100; i++)
            {
                    System.out.print((i+1)+" \t");
             }
             System.out.println("\n"); }
   }
class practical9
{
        public static void main(String args[])
       {
                new A().start();
                new B().start();
                new C().start();
}
}
```

Output:


Command Prompt
:\Users\IU2141230164>cd desktop
:\Users\IU2141230164\Desktop>javac practical9.java
:\Users\IU2141230164\Desktop>java practical9
a a a a a a a a a a a a a a
a a a a a a a a a a a a a a
a a a a a a a a a a a a a a
a a a a a a a a a a a a a a
a a a a a a a a a a a a a a
a a a a a a a a a a a a a a
a a a a a a a a a a a a a a
a b b b b b b b b b b b b b
b b b b b b b b b b b b b 1
b b b b b b b b b b b b b b
b b b b b b b b b b b b b b
b b b b b b b b b b b b b b
b b b b b b b b b b b b b b
b b b b 2 3 4 5 6 7 8 9 10 11
12 13 14 15 16 17 18 19 20 21 22 23 24 25
26 27 28 29 30 31 32 33 34 35 36 37 38 39
40 41 42 43 44 45 46 47 48 49 50 51 52 53
54 55 56 57 58 59 60 61 62 63 64 65 66 67
68 69 70 71 72 73 74 75 76 77 78 79 80 81
82 83 84 85 86 87 88 89 90 91 92 93 94 95
96 97 98 99 100

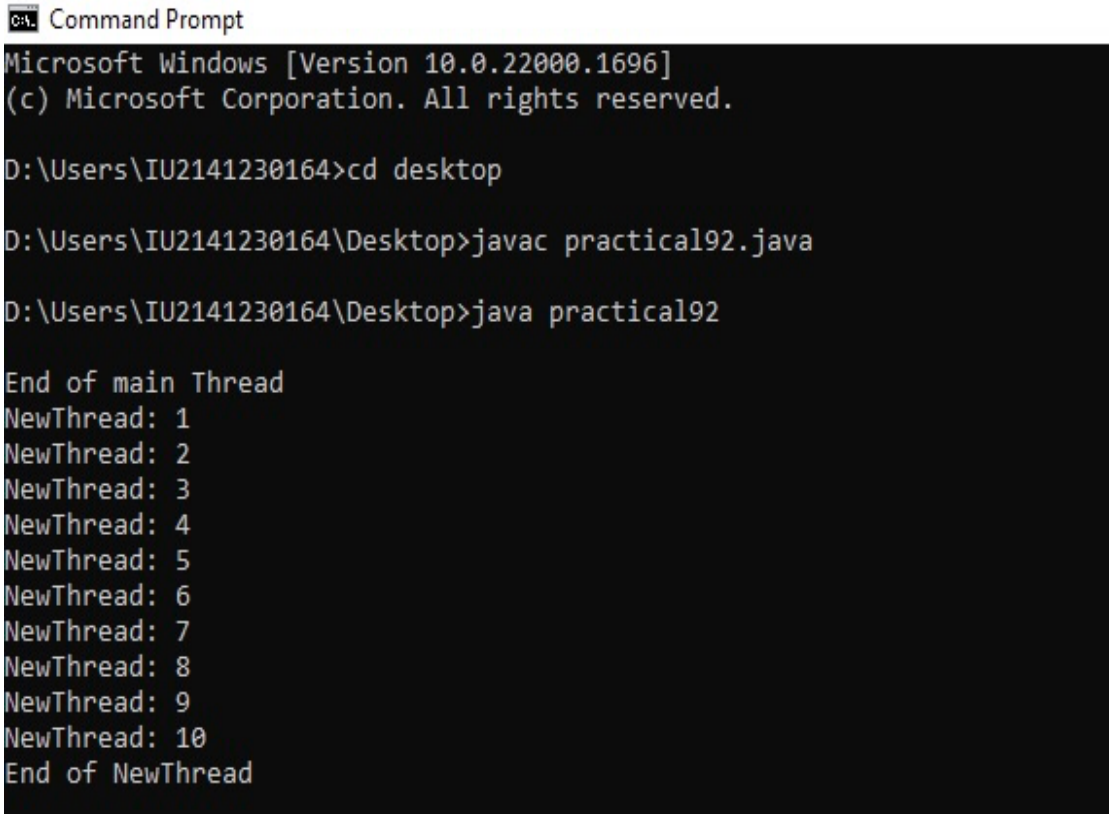
**9.2 Write the thread program -1using Runnable interface.**

File Name: practical92.java

```
class MyThread implements Runnable
 {
        public void run()
        {
                for(int i=0; i<10; i++)
                {
                        System.out.println("NewThread: "+(i+1));
                }
               System.out.println("End of NewThread\n");
         }
  }
class practical92
{
       public static void main(String args[])
      {
             MyThread runn = new MyThread();
             Thread NewThread = new Thread(runn);
            NewThread.start();
             System.out.println("\nEnd of main Thread");
       }
```

```
}
```

Output:


Command Prompt
Microsoft Windows [Version 10.0.22000.1696]
(c) Microsoft Corporation. All rights reserved.
D:\Users\IU2141230164>cd desktop
D:\Users\IU2141230164\Desktop>javac practical92.java
D:\Users\IU2141230164\Desktop>java practical92
End of main Thread
NewThread: 1
NewThread: 2
NewThread: 3
NewThread: 4
NewThread: 5
NewThread: 6
NewThread: 7
NewThread: 8
NewThread: 9
NewThread: 10
End of NewThread

## Practical - 10

### 10.1 Write a program that takes two files names (source and destination) as command line argument. Copy source file's content to destination file. Use character stream class. Also do same using byte stream and buffer stream

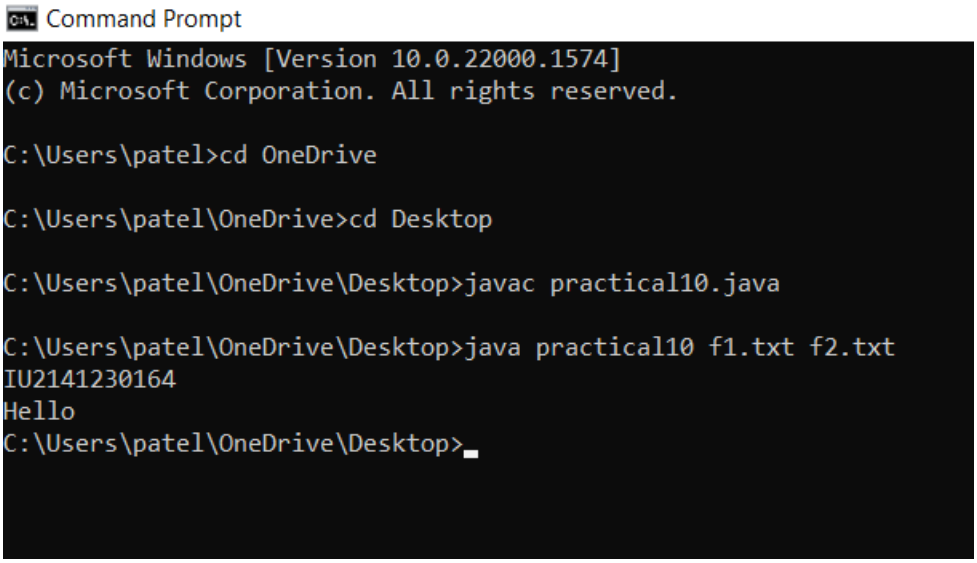
File1 Name: practical10_1a.java

```
//Using Character Stream

import java.io.*;
class practical10_1a
{

        public static void main(String args[])

        {

            String f_name1 = args[0];
            String f_name2 = args[1];
            FileReader Infile = null;

            FileWriter Outfile = null;

            int ReadC;
try
{

      Infile = new FileReader(f_name1);

      Outfile = new FileWriter(f_name2);

      while((ReadC = Infile.read()) != -1)

     {

              System.out.print((char)ReadC);

              Outfile.write(ReadC);

      }
```

```
 }
catch(FileNotFoundException e)
        {
                System.out.println("File Not Found!");
        }
         catch(IOException e)
         {
                 System.out.println(e.getMessage());
         }
 finally
        {
try
         {
                 Infile.close();
                Outfile.close(); }
                 catch(IOException e)
              {
                   System.out.println(e.getMessage());
               }
           }
        }
```

}

Output:




File2 Name: practical10_1b.java

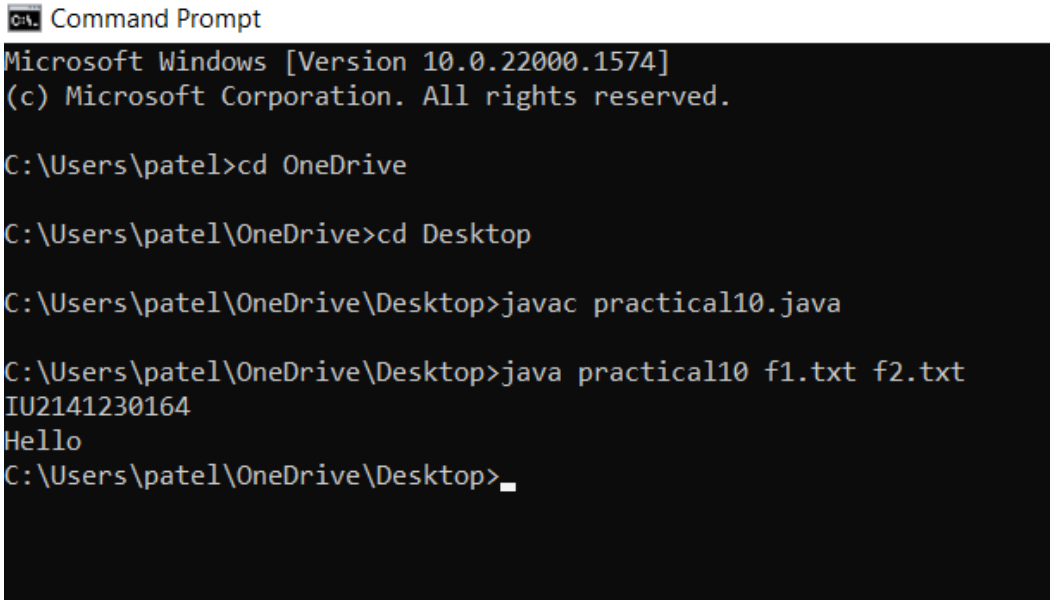
//Using Byte Stream:

```
import java.io.*;

class practical10_1b

 {

          public static void main(String args[])

        {

             String f_name1 = args[0];
             String f_name2 = args[1];

            FileInputStream Infile = null;

            FileOutputStream Outfile = null;

             byte ReadB;
```

```
 try
{

Infile = new FileInputStream(f_name1);

Outfile = new FileOutputStream(f_name2);

do

      {
            ReadB = (byte)Infile.read();

           System.out.print((char)ReadB);

            Outfile.write(ReadB);

       }

while(ReadB != -1);

 }

catch(FileNotFoundException e)

           {

                    System.out.println("File Not Found!");

             }

catch(IOException e)

 {

                     System.out.println(e.getMessage());

}

finally

 {
```

```
Try

 {

              Infile.close();

             Outfile.close();

}

             catch(IOException e)

            {

             System.out.println(e.getMessage());

            }

        }

   }

}
```

Output:

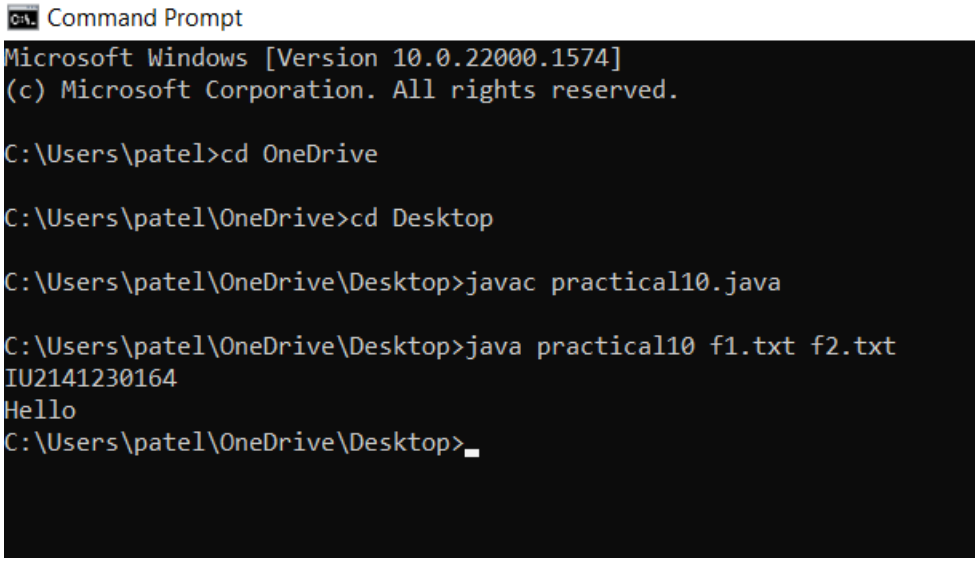


File3 Name: practical10_1c.java

```
//Using Buffer Stream

import java.io.*;
class practical10_1c
{
        public static void main(String args[])
 {
            String f_name1 = args[0];
            String f_name2 = args[1];

            BufferedReader Infile = null;

            BufferedWriter Outfile = null;

           int ReadBuff;

try
{
         Infile = new BufferedReader(new FileReader(f_name1));

         Outfile = new BufferedWriter(new FileWriter(f_name2));

         while((ReadBuff = Infile.read()) != -1)

        {
                System.out.print((char)ReadBuff);

                Outfile.write(ReadBuff);

        }
}
catch(FileNotFoundException e)
```

```
 {
            System.out.println("File Not Found!");
 }
catch(IOException e)
 {
             System.out.println(e.getMessage());
}
             finally
            {
             try
            {
                        Infile.close();
                        Outfile.close();
              }
                        catch(IOException e)
                       {
                              System.out.println(e.getMessage());
                       }
               }
       }
}
```

Output:


Command Prompt
Microsoft Windows [Version 10.0.22000.1574]
(c) Microsoft Corporation. All rights reserved.
C:\Users\patel>cd OneDrive
C:\Users\patel\OneDrive>cd Desktop
C:\Users\patel\OneDrive\Desktop>javac practical10.java
C:\Users\patel\OneDrive\Desktop>java practical10 f1.txt f2.txt
IU2141230164
Hello
C:\Users\patel\OneDrive\Desktop>

**10.2 Write a program which generates random integers and stores them in a file named "rand.dat". The program then reads the integers from the file and displays on the screen.**

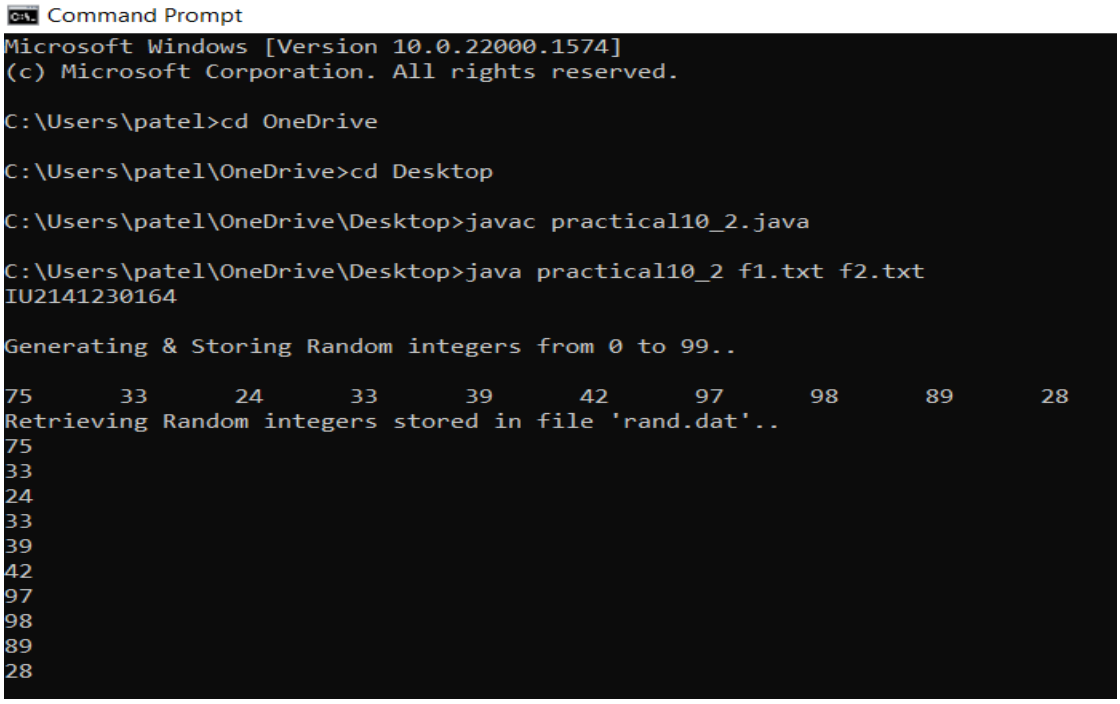
File Name: practical10_2.java

```
import java.io.*;

import java.util.*;

class practical10_2

{

        public static void main(String args[])

        {

              FileWriter FW = null;

              BufferedWriter bfwr = null;

             File inFile = new File("rand.dat");

          try

          {

                int rand_int;

                FW = new FileWriter(inFile);

                bfwr = new BufferedWriter(FW);

                Random generate_rand = new Random();

         System.out.println("\nGenerating & Storing Random integers from 0 to 99..\n");

         for(int i=0; i<10; i++)

        {

             rand_int = generate_rand.nextInt(100);

             System.out.print(rand_int+"\t");

            bfwr.write(rand_int+"\r\n");
```

```
            }
        }
catch(IOException e)
    {
            System.out.println(e.getMessage());
    }
            finally
           {
            try
                  {
                        bfwr.close();
                   }
              catch(IOException e){ }
            }
 try
{
System.out.println("\nRetrieving Random integers stored in file 'rand.dat'..");
 Scanner scanner = new Scanner(inFile);
while(scanner.hasNextInt())
    {
          System.out.println(scanner.nextInt()+"\t");
        }
```

```
}
catch(NullPointerException e)
        {
                System.out.println(e.getMessage());
        }
        catch(FileNotFoundException e)
        {
                System.out.println(e.getMessage());
        }
    }
}
```

Output:




**Practical - 11**

**11.1 Write the program that demonstrate the use of Stack, Vector and ArrayList classes.**

File Name: practical11.java

```
import java.util.*;

public class practical11

{

        public static void main(String[] args)

                {

   // Example of using Stack Stack<Integer> stack = new Stack<>();

                        stack.push(10);

                         stack.push(20);
                         stack.push(30);
                         System.out.println("\nDemonstration of Stack");

                         System.out.println("Elements in Stack: " + stack);

                         System.out.println("Pop element from Stack: " + stack.pop());

                         System.out.println("Elements in Stack after pop operation: " + stack);

 // Example of using Vector
                        Vector<Integer> vector = new Vector<>();

                         vector.add(40);
                         vector.add(50);
                         vector.add(60);

                        System.out.println("\nDemonstration of Vector");

                        System.out.println("Elements in Vector: " + vector);
```

```
        System.out.println("Element at index 1 in Vector: " + vector.get(1));
```

IU2141230164

```
// Example of using ArrayList
  ArrayList<String> arrayList = new ArrayList<>();
  arrayList.add("Apple");

  arrayList.add("Banana");
  arrayList.add("Orange");
  System.out.println("\nDemonstration of ArrayList");

 System.out.println("Elements in ArrayList: " + arrayList);

 System.out.println("Element at index 2 in ArrayList: " + arrayList.get(2));

 arrayList.set(1, "Mango");
System.out.println("Elements in ArrayList after set operation: " + arrayList);

        }

}
```

Output:

Command Prompt

```
Microsoft Windows [Version 10.0.22000.1696]
(c) Microsoft Corporation. All rights reserved.

D:\Users\IU2141230164>cd desktop

D:\Users\IU2141230164\Desktop>javac practical11.java

D:\Users\IU2141230164\Desktop>java practical11

Demonstration of Stack
Elements in Stack: [10, 20, 30]
Pop element from Stack: 30
Elements in Stack after pop operation: [10, 20]

Demonstration of Vector
Elements in Vector: [40, 50, 60]
Element at index 1 in Vector: 50

Demonstration of ArrayList
Elements in ArrayList: [Apple, Banana, Orange]
Element at index 2 in ArrayList: Orange
Elements in ArrayList after set operation: [Apple, Mango, Orange]

D:\Users\IU2141230164\Desktop>
```

## Practical – 12

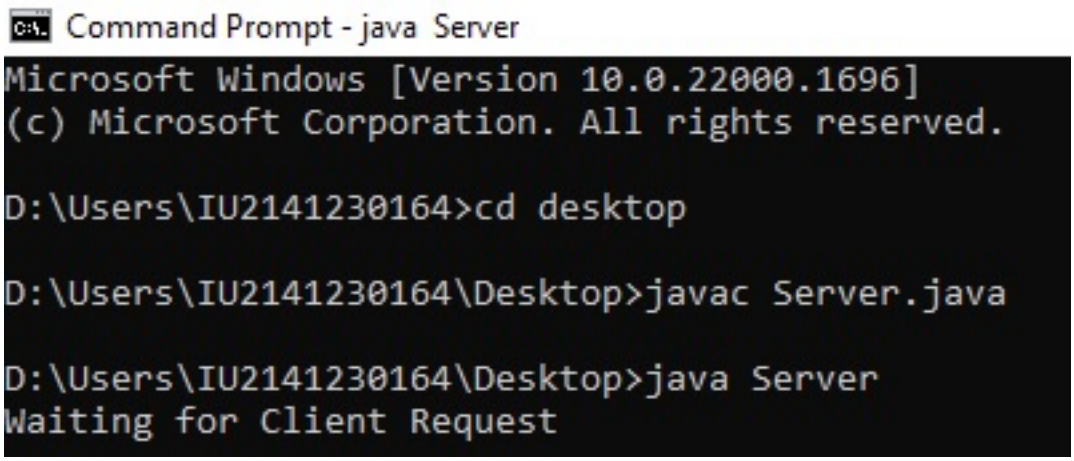
**12.1 Write a Network program that client sends the data as radius of circle to server and server received that data and send the resultant area of circle to requested client.**

File1 Name: Server.java

```
// Server Program import java.io.*;

import java.net.*;

class Server

 {
        public static void main(String args[])

         {

          try
          {

            ServerSocket ss=new ServerSocket(4096);

            System.out.println("Waiting for Client Request");

            Socket s=ss.accept();
            BufferedReader br;

            PrintStream ps;
            String str;
            br=new BufferedReader(new InputStreamReader(s.getInputStream()));

           str=br.readLine();
          System.out.println("Received radius");
          double r=Double.parseDouble(str);
          double area=3.14*r*r;
          ps=new PrintStream(s.getOutputStream());

           ps.println(String.valueOf(area));
```

```
            br.close();

            ps.close();

            s.close();

            ss.close();

        }
catch(Exception e)

        {

                System.out.println(e);

         }

    }

}
```

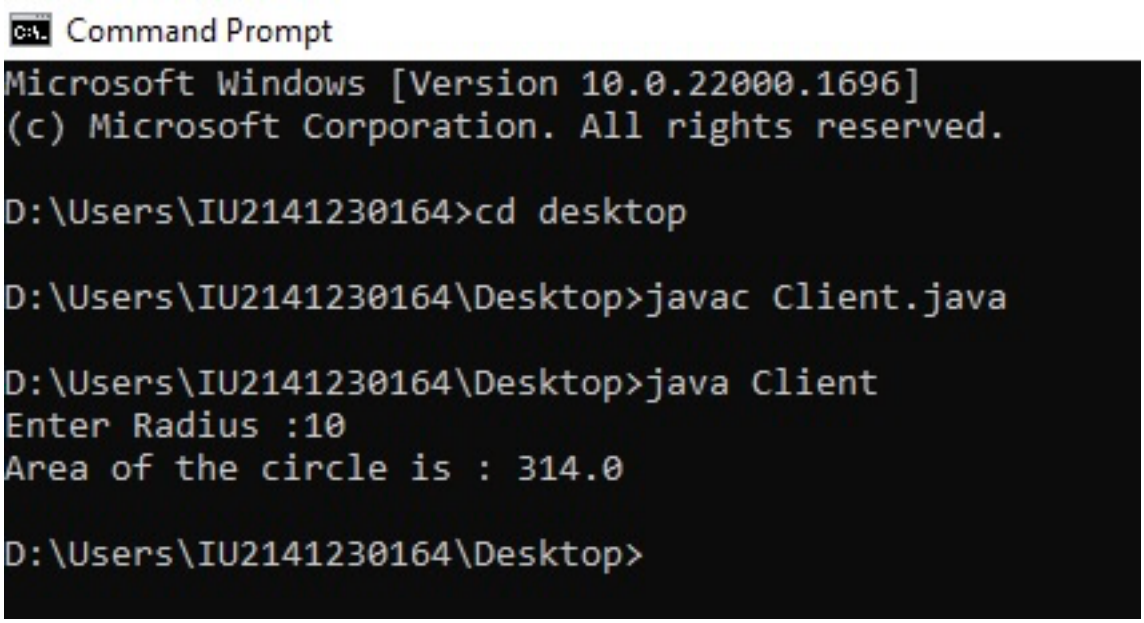
Output:



File2 Name: Client.java

```
//Client Program import java.io.*;

import java.net.*;

class Client

{
public static void main(String args[])throws IOException

        {

                Socket s=new Socket(InetAddress.getLocalHost(),4096);

                 BufferedReader br;
                 PrintStream ps;
                 String str;
                 System.out.print("Enter Radius :");
                 br=new BufferedReader(new InputStreamReader(System.in));
                 ps=new PrintStream(s.getOutputStream());
                 ps.println(br.readLine());
                 br=new BufferedReader(new InputStreamReader(s.getInputStream()));

                 str=br.readLine();
                 System.out.println("Area of the circle is : "+str);
                br.close();
                ps.close();

          }

   }
```

Output:


Command Prompt
Microsoft Windows [Version 10.0.22000.1696]
(c) Microsoft Corporation. All rights reserved.
D:\Users\IU2141230164>cd desktop
D:\Users\IU2141230164\Desktop>javac Client.java
D:\Users\IU2141230164\Desktop>java Client
Enter Radius :10
Area of the circle is : 314.0
D:\Users\IU2141230164\Desktop>

## Practical - 13

### 13.1 Write a program to count occurrence of character in a string.

File Name: practical13.java

```
import java.util.*;

class practical13

{

        public static void main(String[] args)

{

        char ch;
        int count=0;
        String str;
        Scanner sc = new Scanner(System.in);

 System.out.print("Enter string: "); str=sc.nextLine();

System.out.print("Enter Character whose occurence you want to find: ");
ch=sc.next().charAt(0);

        for(int i=0;i<str.length();i++)

{

        if(str.charAt(i)==ch)

        {

                count++;

         }

}
```

```
            System.out.println("Character Occurs " +count +"times in a string");
        }
}
```

Output: