Lab Record

of

Java Programming Concepts
(CS202)

Submitted to:

Dr. Garima Verma
Assistant Prof.
School of Computing
DIT University

Submitted by:

Student Name-Ambhik Ishan
Rollno-190102305
CSE-Section-E
2nd Yr (3rd Sem)

Session 2020-21

Index

S.No	Title of Experiment/Objective	Date of Conduction	Signature of Faculty
1	Write a program in java to calculate simple interest.	26-08-2020	
2	Write a program to design simple calculator for (+, -, *, and /) using switch case.	02-09-2020	
3	Write a program to design accounts class and two instance functions withdraw () and deposit (). Also, declare two variables accno and balance. Display total balance updations after withdrawl and deposit.	16-09-2020	
4	Write a program in Java to implement a class car with engineno, car_type (sedan or hatchback), and a constructor with two parameters. Create a child class alto with color and a constructor. Display all the information of alto car using a display() function.	23-09-2020	
5	Write a program to the concept of Method overloading by designing functions to sum different type of numbers (Minimum 3 functions).	30-09-2020	
6	Write a program to sort 10 names in ascending order using the String array.	07-09-2020	
7	Write a program to create a copy of text file n harddisk using I/O stream class in java.	21-10-2020	
8	WAP to Handle the user defined Exception using throw keyword to check user age is valid or not (must be >=18)	28-10-2020	
9	WAP to create a thread that Implement the Runnable interface for implementing loop in forward and reverse order simultaneously.	17-11-2020	
10	WAP to Draw the line, Rectangle, oval,text using the graphics method in applet.		

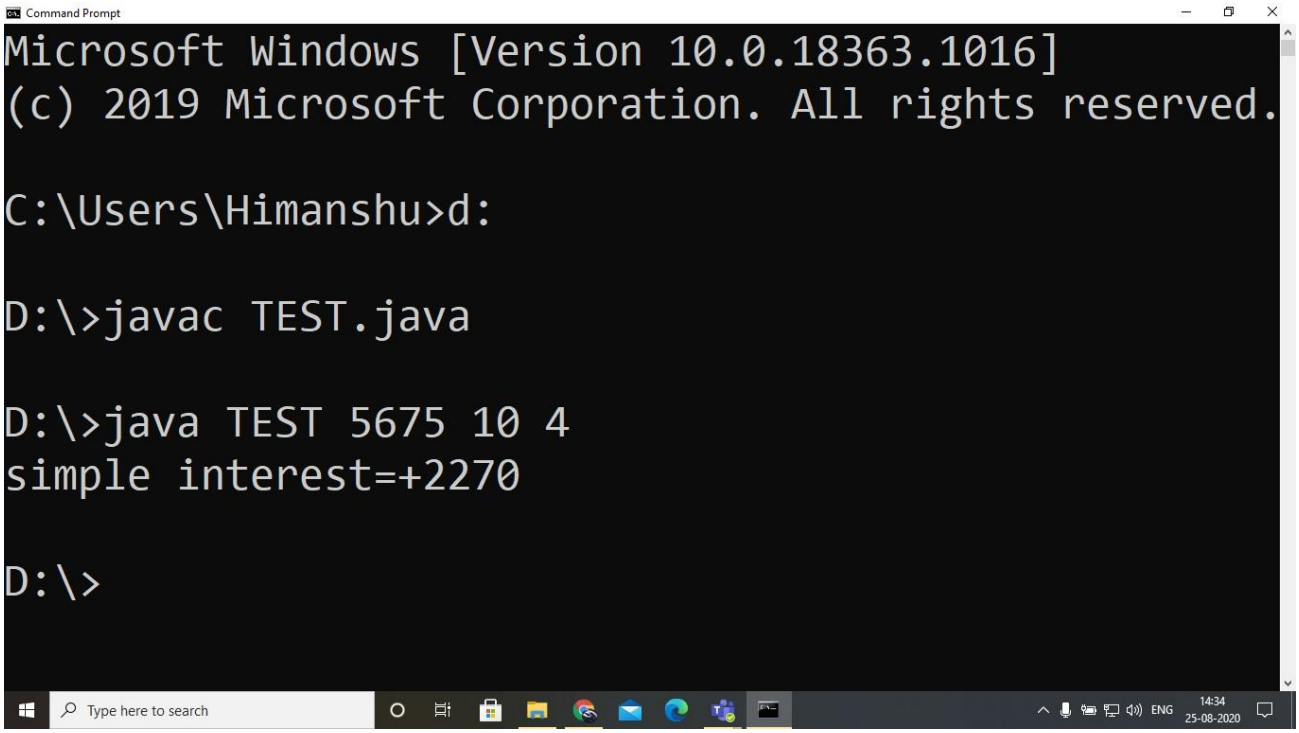
Experiment – 1

Objective: Write a program in java to calculate simple interest.

CODE:

```
public class TEST {

    public static void main(String[] ar) {
        int p,r,t,si;
        p=Integer.parseInt(ar[0]);
        r=Integer.parseInt(ar[1]);
        t=Integer.parseInt(ar[2]);
        si=p*r*t/100;
   System.out.println("simple interest=+"+si);

    }
}
```

OUTPUT:



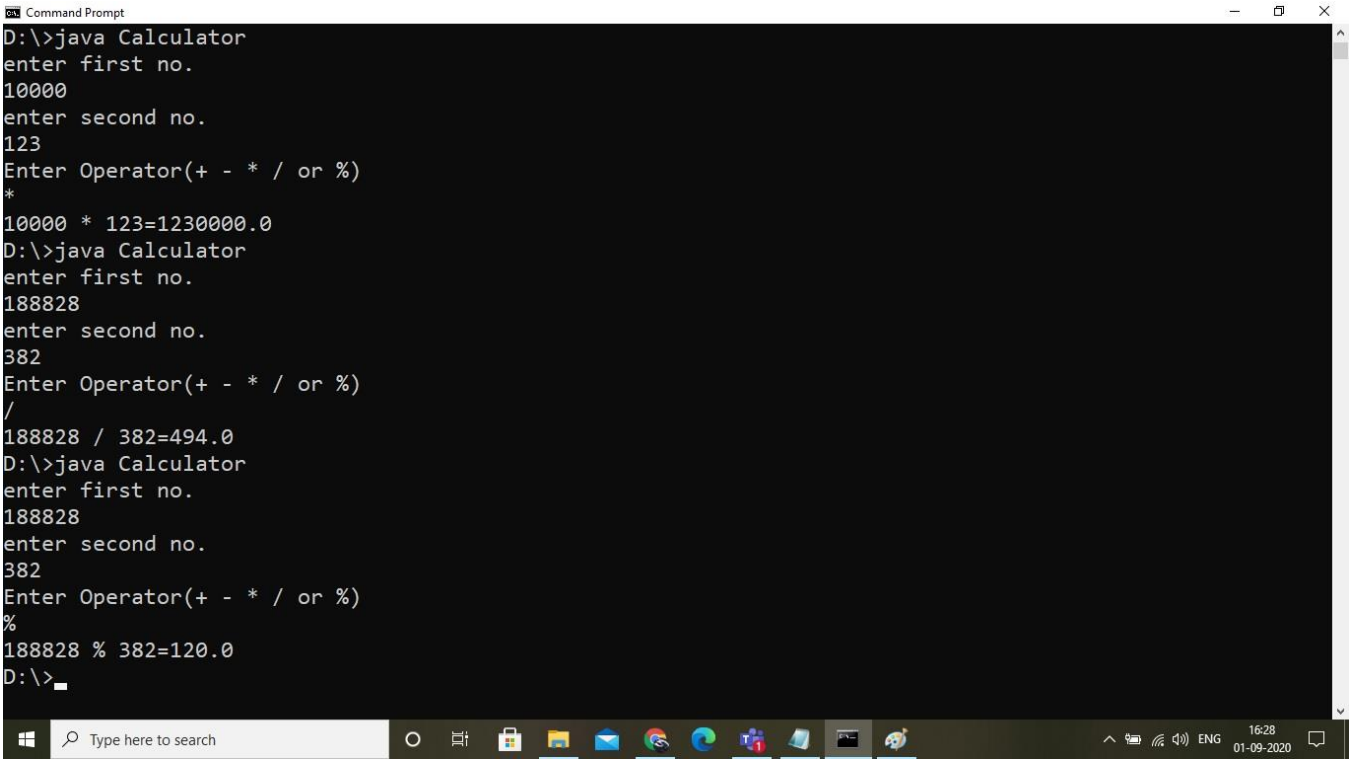
Experiment – 2

Objective: Write a program to design simple calculator for (+, -, *, and /) using switch case.

Code:

```
import java.util.Scanner;
public class Calculator{
public static void main(String[] ar){
Scanner sc=new Scanner(System.in);
System.out.println("enter first no.");
int a=sc.nextInt();
System.out.println("enter second no.");
int b=sc.nextInt();
System.out.println("Enter Operator(+ - * / or %)");
char operator=sc.next().charAt(0);
float result=0;
switch(operator)
{ case '+':{result=a+b;
           break;}
  case '-':{result=a-b;
           break;}
  case '*':{result=a*b;
           break;}
  case '/':{result=a/b;
           break;}
  case '%':{result=a%b;
           break;}
}
System.out.print(a+" "+operator+" "+b+"="+result);
}
}
```

OUTPUT:



Experiment – 3

Objective:

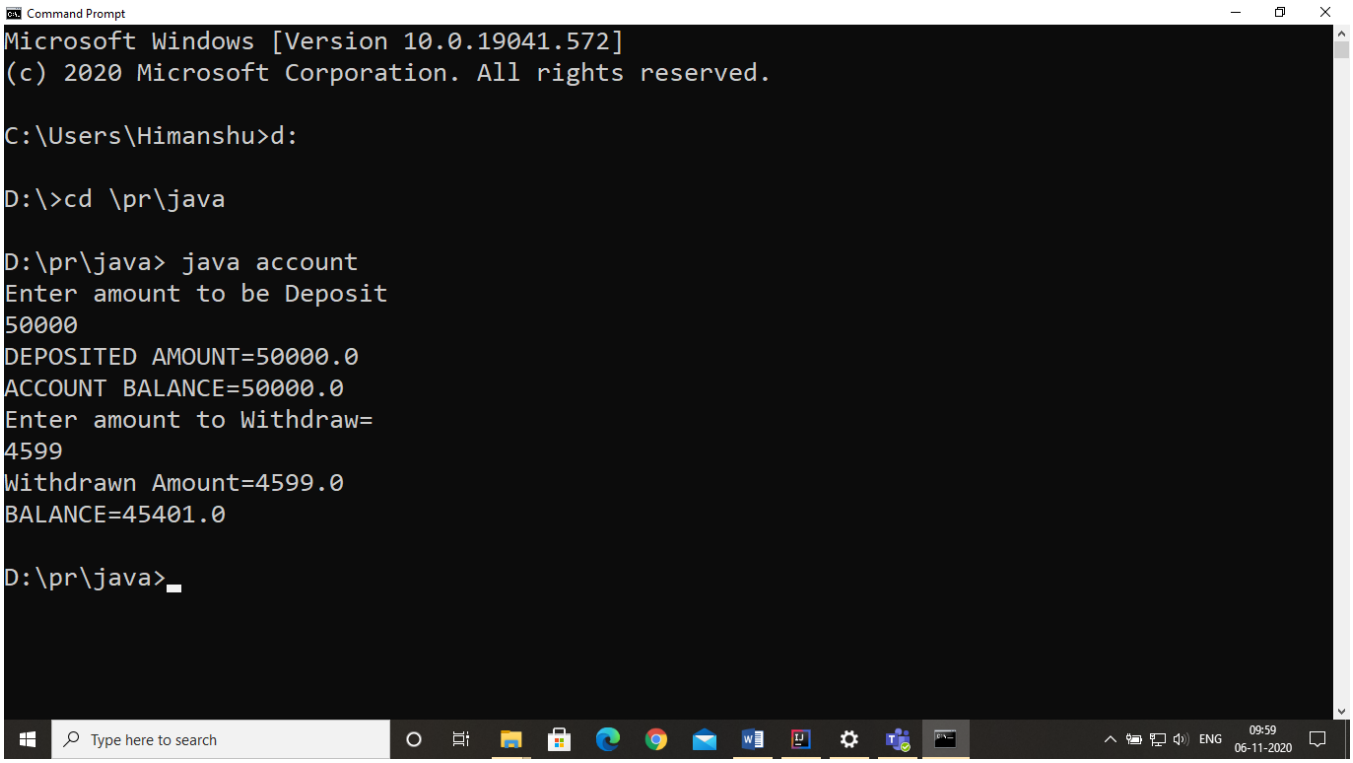
Write a program to design accounts class and two instance functions withdraw () and deposit (). Also, declare two variables accno and balance. Display total balance updations after withdrawl and deposit.

Code:

```
import java.util.Scanner;
public class account{
   static float balance=0;
   void deposit(){
       Scanner in=new Scanner(System.in);
       System.out.println("Enter amount to be Deposit");
       float d=in.nextFloat();
       balance=d+balance;
       System.out.println("DEPOSITED AMOUNT="+d);
       System.out.println("ACCOUNT BALANCE="+balance);

   }
   void withdraw(){
       Scanner in=new Scanner(System.in);
```

```
        System.out.println("Enter amount to Withdraw=");
        float w=in.nextFloat();
        balance=balance-w;
        System.out.println("Withdrawn Amount="+w);
        System.out.println("BALANCE="+balance);


    }
    public static void main(String[] args){
        account a=new account();
        a.deposit();
        a.withdraw();

    }
}
```

OUTPUT:



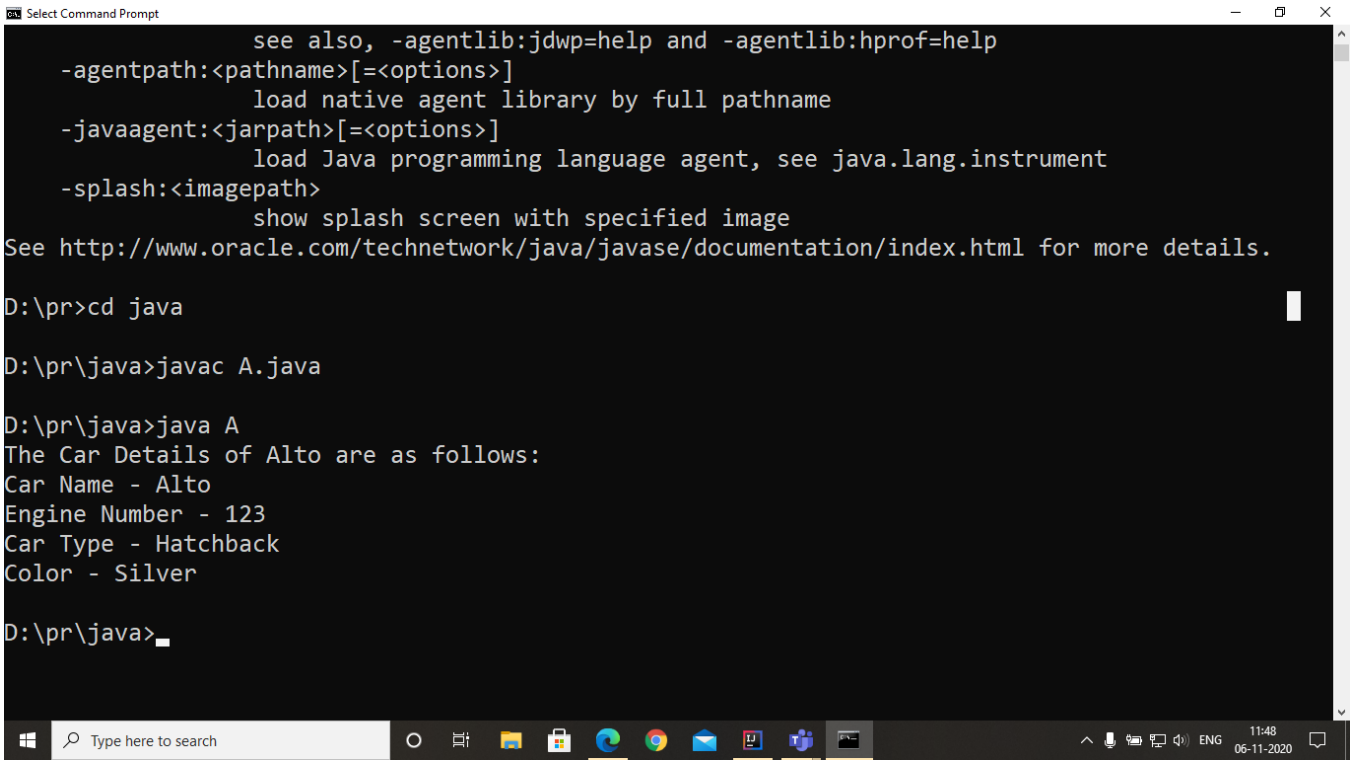
Experiment-4

Objective: Write a program in Java to implement a class car with engineno, car_type (sedan or hatchback), and a constructor with two parameters. Create a child class alto with color and a constructor. Display all the information of alto car using a display() function.

Code:

```
import java.util.*;
class car
{
  int engineno;
  String car_type;
  car(int a,String b)
  {
    engineno=a;
    car_type=b;
  }
}
class alto extends car
{
  String color;
   alto()
   {
    super(123,"Hatchback");
    color="Silver";
    }
   void display()
   {
     System.out.println("The Car Details of Alto are as follows:");
    System.out.println("Car Name - Alto");
    System.out.println("Engine Number - "+engineno);
    System.out.println("Car Type - "+car_type);
   System.out.println("Color - "+color);
   }
 }
public class A
{
  public static void main( String args[])
{

    alto n= new alto();
    n.display();
}
}
```

Output:



Experiment-5

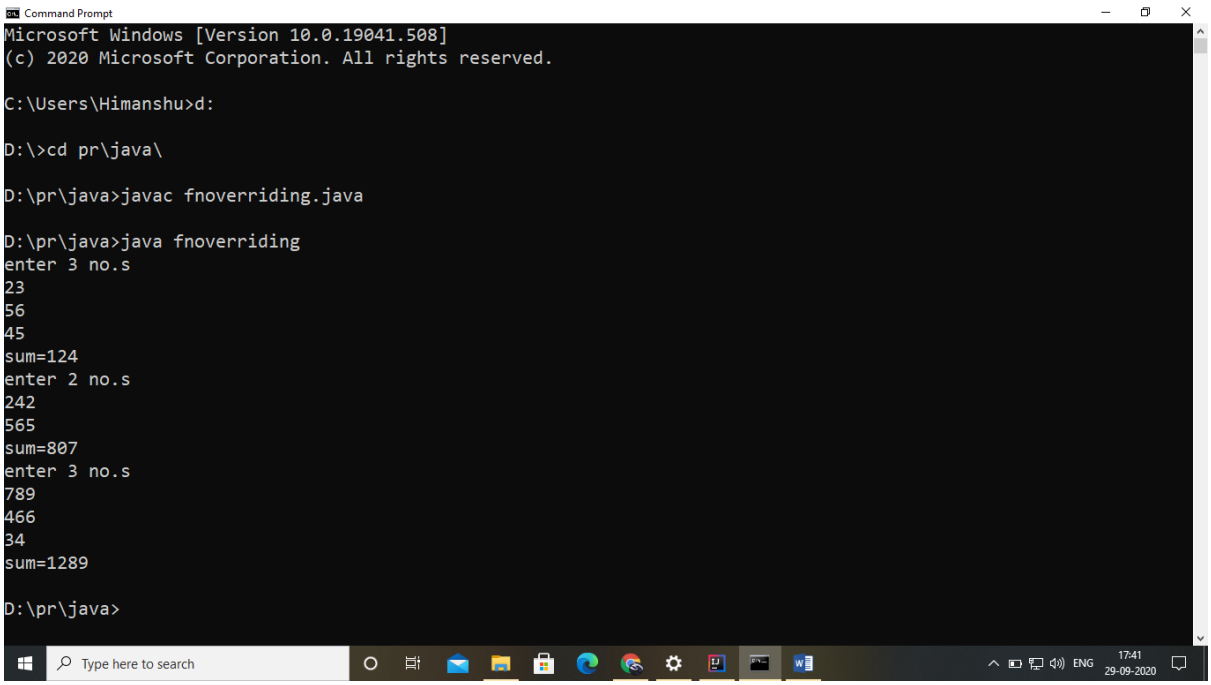
Objective: Write a program to the concept of Method overloading by designing functions to sum different type of numbers.

Code:

```
import java.util.*;
public class fnoverriding{
    int sum(int a,int b,int c)
    {

        int s=a+b+c;
        return(s);

    }
```

```
    int sum(int a,int b)
    {

        int s=a+b;
        return(s);

    }
    void sum(){
        Scanner sc=new Scanner(System.in);
        System.out.println("enter 3 no.s");
        int a= sc.nextInt();
        int b=sc.nextInt();
        int c=sc.nextInt();
        int s=a+b+c;
        System.out.println("sum="+s);
    }
    public static void main(String[] args){
        fnoverriding fn=new fnoverriding();
        Scanner sc=new Scanner(System.in);
        System.out.println("enter 3 no.s");
        int g=sc.nextInt();
        int h=sc.nextInt();
        int i=sc.nextInt();
        System.out.println("sum="+fn.sum(g,h,i));
        System.out.println("enter 2 no.s");
        int d=sc.nextInt();
        int e=sc.nextInt();

        System.out.println("sum="+fn.sum(d,e));

        fn.sum();









    }
}
```

Output:



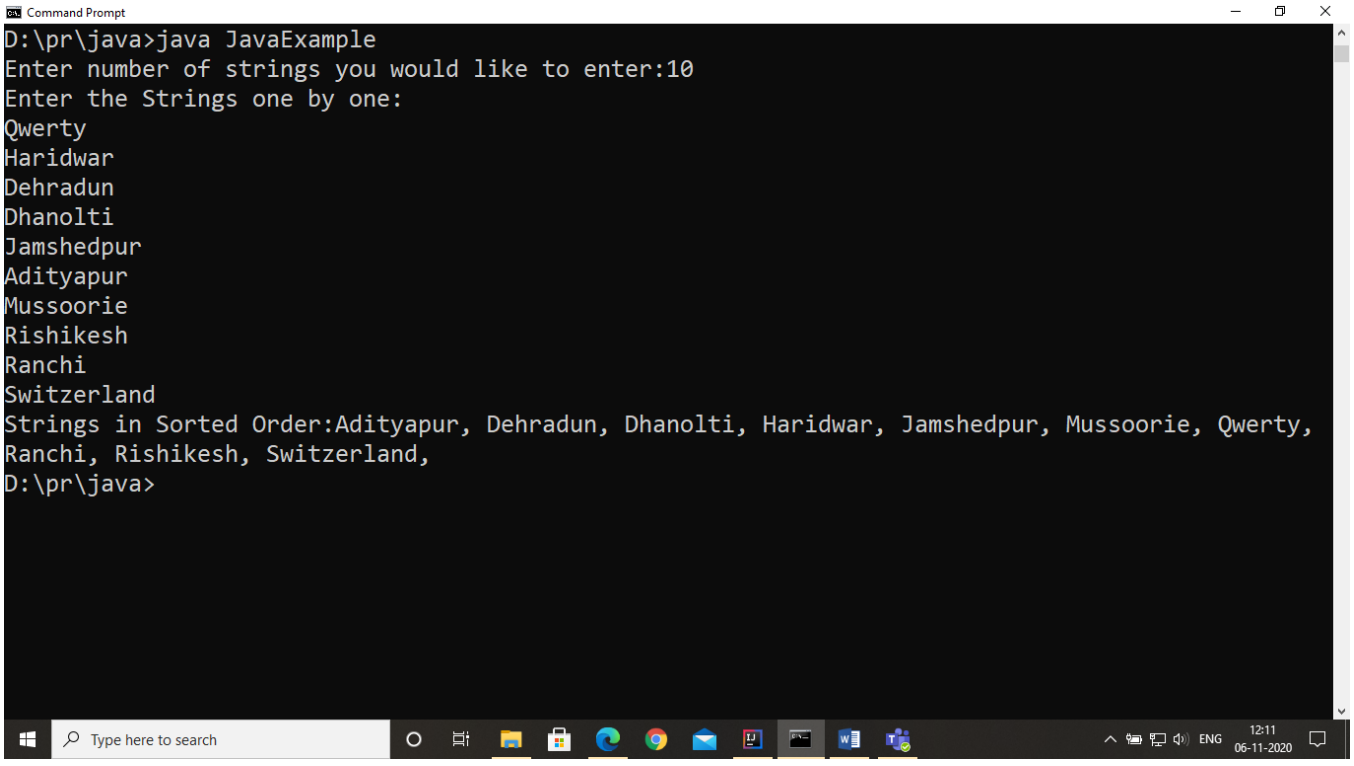
Experiment-6

Objective:

```
import java.util.Scanner;
public class JavaExample
{
    public static void main(String[] args)
    {
        int count;
        String temp;
        Scanner scan = new Scanner(System.in);

        //User will be asked to enter the count of strings
        System.out.print("Enter number of strings you would like to enter:");
        count = scan.nextInt();


        String str[] = new String[count];
        Scanner scan2 = new Scanner(System.in);

        //User is entering the strings and they are stored in an array
        System.out.println("Enter the Strings one by one:");
        for(int i = 0; i < count; i++)
        {
            str[i] = scan2.nextLine();
        }
        scan.close();
        scan2.close();

        //Sorting the strings
```

```
        for (int i = 0; i < count; i++)
        {
            for (int j = i + 1; j < count; j++) {
                if (str[i].compareTo(str[j])>0)
                {
                    temp = str[i];
                    str[i] = str[j];
                    str[j] = temp;
                }
            }
        }

        //Displaying the strings after sorting them based on alphabetical order
        System.out.print("Strings in Sorted Order:");
        for (int i = 0; i <= count - 1; i++)
        {
            System.out.print(str[i] + ", ");
        }
    }
}
```

Output:



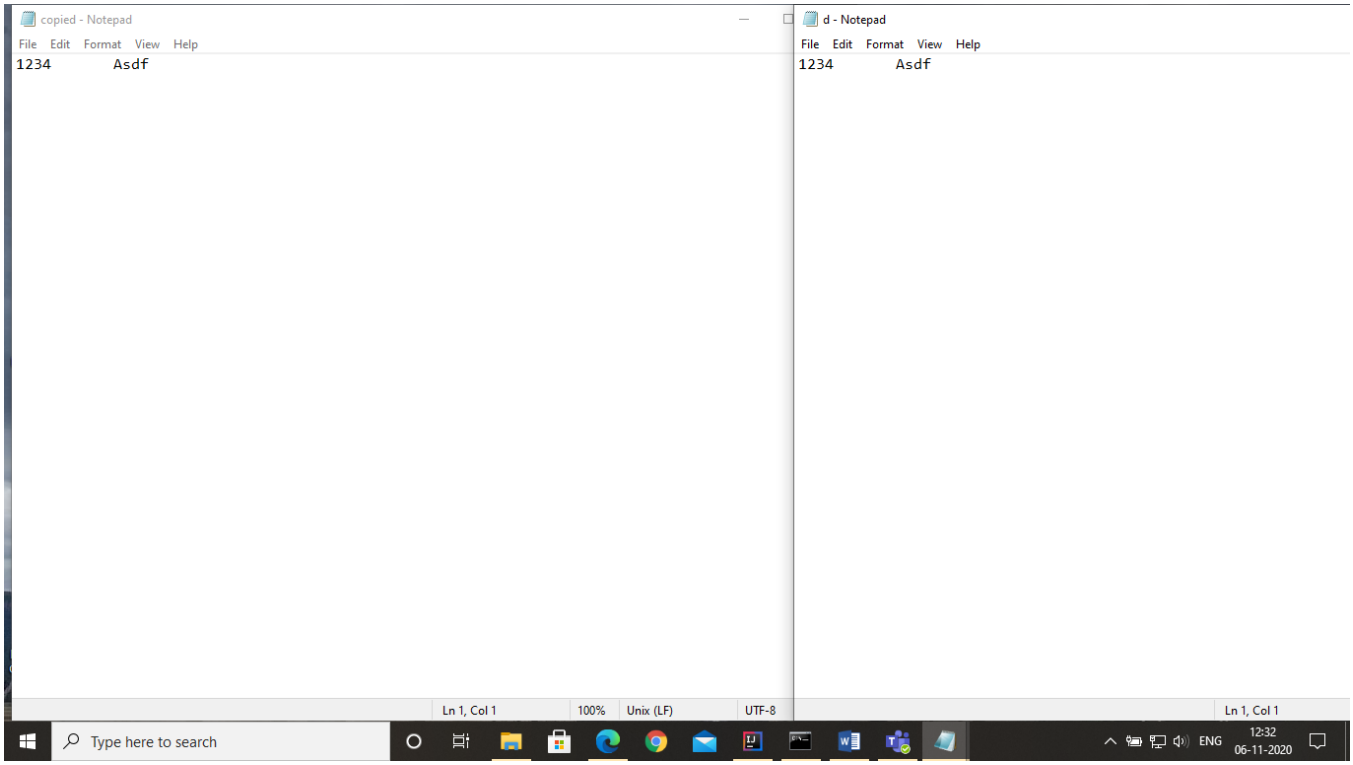
Experiment-7

Objective: Write a program to create a copy of text file n harddisk using I/O stream class in java.

Code:

```
import java.io.*;
public class copyfile{
    public static void main(String[] args){
       try {
           FileInputStream fin = new FileInputStream("d.txt");

           FileOutputStream fout = new FileOutputStream("copied.txt");
           int c;
           while ((c = fin.read()) != -1)
               fout.write(c);
       }catch (IOException e){}
    }
}
```

Output:



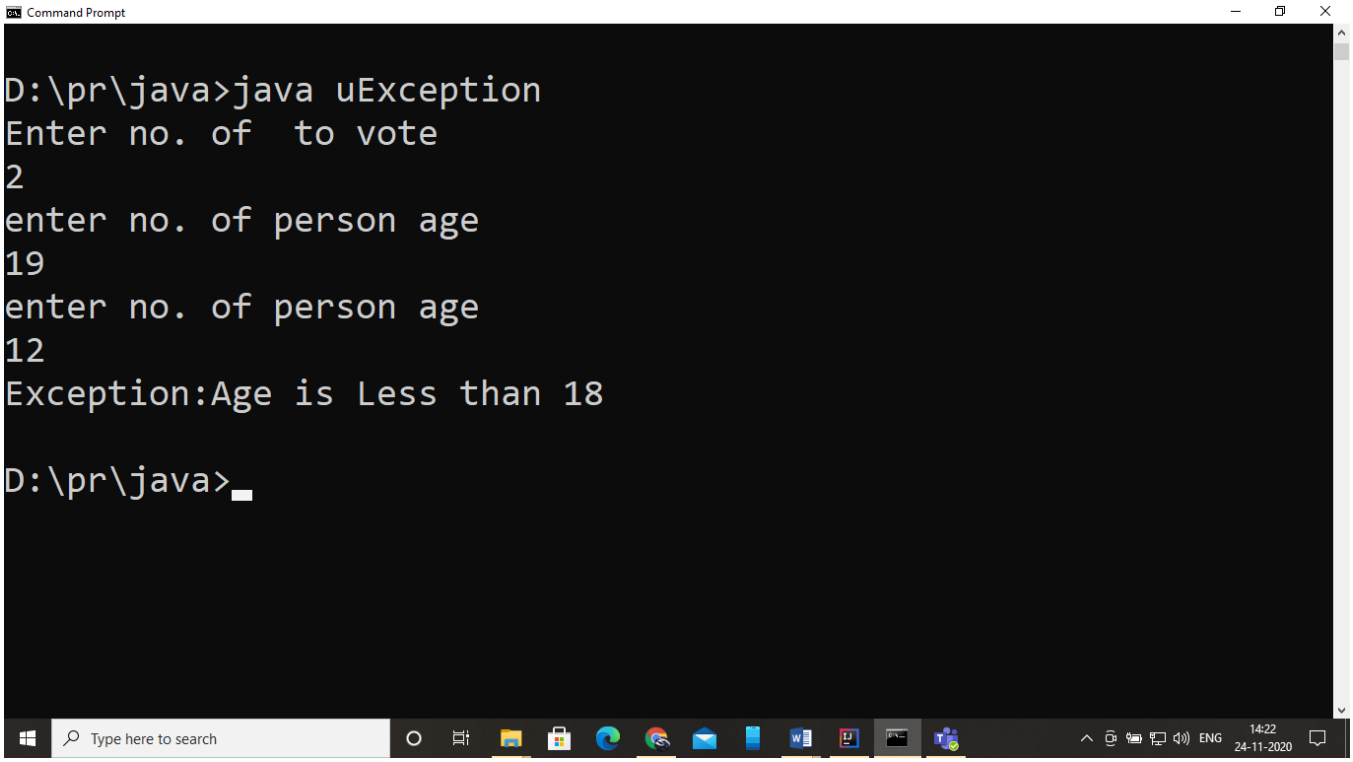
Experiment-8

Objective: WAP to Handle the user defined Exception using throw keyword to check user age is valid or not (must be >=18).

Code:

```
import java.util.*;
class jexception extends Exception{
    String m;
    jexception(String message){
        m=message;
    }
public String toString(){
        return(m);
}
}

public class uException{
    public static void main(String[] args){
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter no. of  to vote");
        int i=0,n;
        n=sc.nextInt();
        int age[]=new int[n];
        try{
            for(i=0;i<n;i++)
            {
                System.out.println("enter no. of person age");
                age[i]=sc.nextInt();
                if(age[i]<18) {
                    throw new jexception("Age is Less than 18");

                }
            }
        }catch (jexception j){
            System.out.println("Exception:"+j);
        }

    }
}
```

Output:



Experiment-9

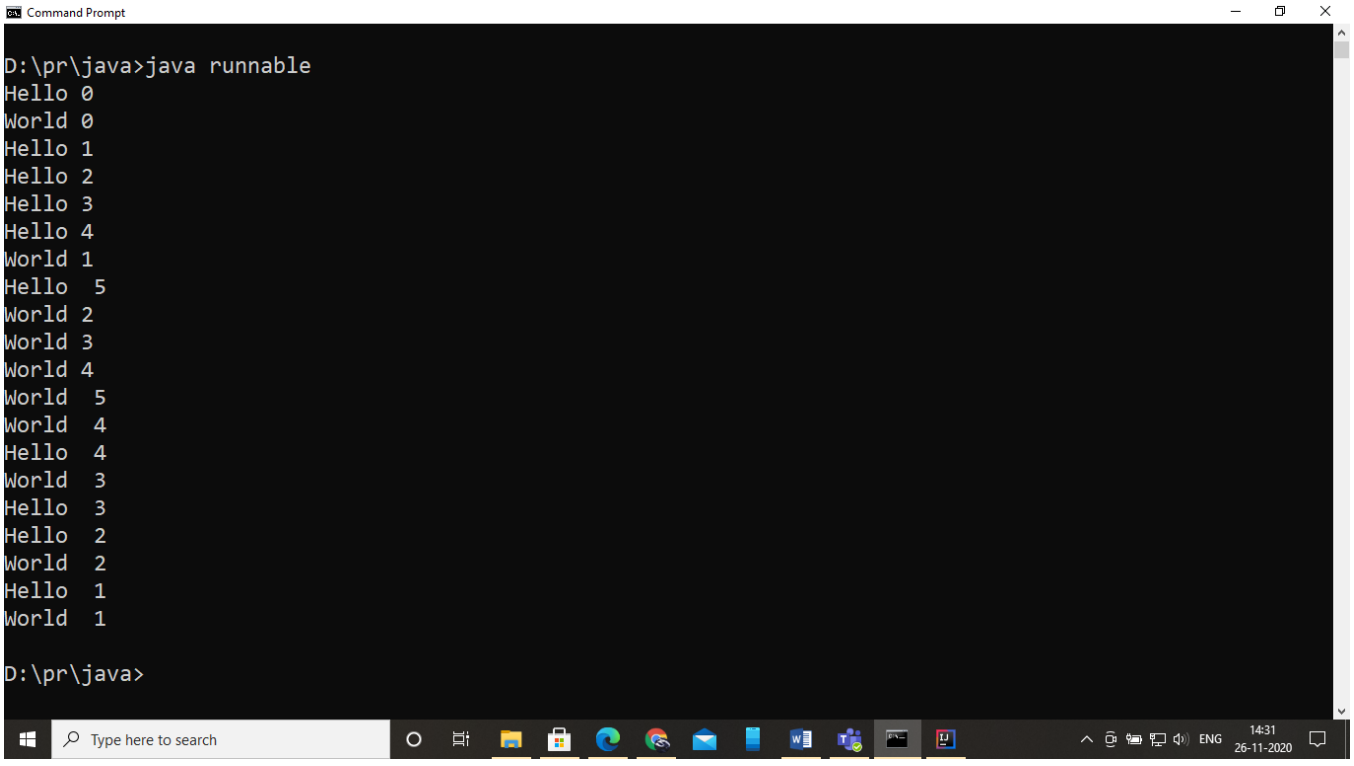
Objective: WAP to create a thread that Implement the Runnable interface for implementing loop in forward and reverse order simultaneously.

Code:

```
class thread implements Runnable{
    String message;
    thread(String s){
    message=s;
    }
    public void run(){
        try {
            for(int i=0;i<5;i++)
                System.out.println(message+" "+i);
            for (int i = 5; i>0 ; i--) {
                System.out.println(message+"  "+i);
                    Thread.sleep(1000);
            }


        }catch (Exception e){
            System.out.println("Exception:"+e);
```

```
        }

    }
}
public class runnable{
    public static void main(String args[]){
    thread t1=new thread("Hello");
    thread t2=new thread("World");

    Thread T1=new Thread(t1);
    Thread T2=new Thread(t2);
    T1.start();
    T2.start();


    }
}
```

Output:



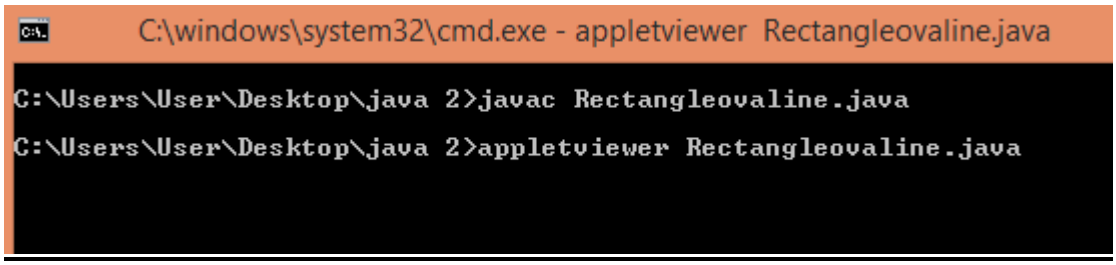
EXPERIMENT – 10

Objective: WAP to Draw the line, Rectangle, oval,text using the graphics method in applet

CODE:

```
import java.applet.Applet;

import java.awt.*;

public class Rectangleovaline extends Applet
{

public void paint(Graphics g)
{

g.setColor(Color.red);

g.drawString("Welcome",50, 50);

g.drawLine(20,30,20,300);

g.drawRect(70,100,30,30);

g.fillRect(170,100,30,30);

g.drawOval(70,200,30,30);

g.setColor(Color.pink);

g.fillOval(170,200,30,30);

g.drawArc(90,150,30,30,30,270);

g.fillArc(270,150,30,30,0,180);

}

}
```

OUTPUT :


C:\windows\system32\cmd.exe - appletviewer Rectangleovaline.java
C:\Users\User\Desktop\java 2>javac Rectangleovaline.java
C:\Users\User\Desktop\java 2>appletviewer Rectangleovaline.java