Chapter – 4 JAVA

4.18 Write a Java program to Shuffle the list elements using all the possible Permutations.

CHAPTER 4

JAVA

Java programming language was originally developed by Sun Microsystems which was initiated by James Gosling and released in 1995. James Gosling initiated Java language project in June 1991 for use in one of his many set-top box projects. The language, initially called 'Oak'.

LAB PROGRAM - 1

1) Write a java program to calculate Harmonic series using command line arguments.

AIM : Write a java program to calculate Harmonic series using command line arguments.

PROGRAM

```
import java.io.*;
class Harmonic
{
    public static void main(String args[]) //main method definition
    {
        int n,i;
        double sum=0.0;
        n=Integer.parseInt(args[0]);        //converting into integer
        for(i=1;i<=n;i++)                   // looping statement
        {
            sum=sum+1/(double)i;            //type casting
            int to double
        }
        System.out.println("Sum="+sum);     //output statement
    }
}
```

TEST DATA AND OUTPUT

For Compilation :

C:\java>javac Harmonic.java

For Execution :

C:\java>java Harmonic 5

Sum=2.283333333333333

LAB PROGRAM - 2

2) Write a java program to perform matrix multiplication.

AIM : Write a java program to perform matrix multiplication.

PROGRAM

```
class Array
{
    public static void main(String args[])
    {
        int a[][]={{1,2},{3,4}}; // 2-dimensional array static initialization
        int b[][]={{5,6},{7,8}};
        int c[][]=new int[2][2];
        int i,j,k;
        for(i=0;i<2;i++)
        {
            for(j=0;j<2;j++)
            {
                c[i][j]=0;
                for(k=0;k<2;k++)
                {
                    c[i][j]=c[i][j]+a[i][k]*b[k][j];
                }
            }
        }
        System.out.println("\n\n    MATRIX A \t      MATRIX B
        \t     MATRIX C (Multiplication of A & B) \n");
        for(i=0;i<2;i++)
        {
            for(j=0;j<2;j++)
            {
                System.out.print("     "+a[i][j]);
            }
            System.out.print("  \t ");
            for(j=0;j<2;j++)
            {
                System.out.print("     "+b[i][j]);
            }
            System.out.print("    \t ");
            for(j=0;j<2;j++)
            {
                System.out.print("     "+c[i][j]);
            }
            System.out.println("\n");
        }
    }
}
```


TEST DATA AND OUTPUT

For Compilation :

C:\java>javac Array.java

For Execution :

C:\java>java Array

MATRIX A	MATRIX B	MATRIX C (Multiplication of A & B)
1 2	5 6	19 22
3 4	7 8	43 50

LAB PROGRAM - 3

3) Write a java program to illustrate class and object.

AIM : Write a java program to illustrate class and object.

Class Diagram :

Student
+ sno,m1,m2,m2,tot : int + sname,result : String + avg : double
+getData( ) : void +putData( ) : void

PROGRAM

```
class Student                          // class definition
{
        int sno,m1,m2,m3,tot;          //member variables
        String sname,result;
        double avg;
        void getData()                 //method definition
        {
                sno=101;
                sname="HARISH";
                m1=95;
                m2=98;
                m3=97;
       }
       void putData()
       {
               tot=m1+m2+m3;
               avg=tot/3;
               if(m1>34&&m2>34&&m3>34)
                       result="Pass";
               else
                       result="Fail";
```



```
               System.out.println("Student Number  Student Name
               Marks1  Marks2  Marks3  Total  Average  Result \n");
               System.out.println(sno+" \t\t"+ sname+" \t\t"+ m1+"
               \t"+m2+" \t"+ m3+"\t"+ tot+"\t" +avg+" \t "+result);
       }
}
class ExClass
{
       public static void main(String args[])
       {
               Student s=new Student();        // creating object for class
               s.getData();                    // calling method through object
               s.putData();
       }
}
```

TEST DATA AND OUTPUT

For Compilation :

C:\java>javac ExClass.java

For Execution :

C:\java>java ExClass

Student Number	Student Name	Marks1	Marks2	Marks3	Total	Average	Result
101	HARISH	95	98	97	290	96.0	Pass

LAB PROGRAM - 4

4) Write a java program to check whether one string is sub string of another string or not.

AIM : Write a java program to check whether one string is sub string of another string or not.

PROGRAM

```
import java.io.*;
class Substring
{
    public static void main(String args[])throws IOException
    {
          BufferedReader br=new BufferedReader(new
               InputStreamReader(System.in));        //input statement
          String str1,str2;
          System.out.println("Enter first string");
          str1=br.readLine();                         //reading data
          System.out.println("Enter second string");
          str2=br.readLine();
          int n=str1.indexOf(str2);
          if(n!=-1)
            System.out.println(str2+" is substring of "+str1);
          else
            System.out.println(str2+" is not substring of "+str1);
    }
}
```

Test data and Output

For Compilation :

C:\java>javac Substring.java

For Execution :

1) C:\java>java Substring
Enter first string
ANANTAPURAMU
Enter second string
ANT
ANT is substring of ANANTAPURAMU

2) C:\java>java Substring
Enter first string
ANANTAPURAMU
Enter second string
ANN
ANN is not substring of ANANTAPURAMU

LAB PROGRAM - 5

5) Write a java program to sort given strings in alphabetical order.

AIM : Write a java program to sort given strings in alphabetical order.

PROGRAM

```
import java.io.*;
import java.lang.*;
class StringSort
{
    public static void main(String args[]) throws IOException
    {
        BufferedReader br=new BufferedReader(new
            InputStreamReader(System.in));
        int i,j,n;
        String temp=null;
        System.out.println("Enter Number of Strings to sort:");
        n=Integer.parseInt(br.readLine());
        String name[]=new String[n];
        for(i=0;i<n;i++)
        {
            System.out.println("Enter String "+(i+1)+":");
            name[i]=br.readLine();
        }
        System.out.println("Before sorting");
        for(i=0;i<n;i++)
        {
            System.out.println("String ["+(i+1)+"]:"+name[i]);
        }
        for(i=0;i<n-1;i++)
        {
            for(j=i+1;j<n;j++)
            {
                if(name[j].compareTo(name[i])<0)
                {
                    temp=name[i];
                    name[i]=name[j];
                    name[j]=temp;
                }
            }
        }
        System.out.println("After sorting");
        for(i=0;i<n;i++)
        {
            System.out.println("String [" +(i+1)+ "]: "+name[i])
        }
    }
}
```

TEST DATA AND OUTPUT

```
Enter Number of Strings to sort :    4
Enter String 1 :
PRAVEEN
Enter String 2 :
AVINASH
Enter String 3 :
CHETHANA
Enter String 4 :
BHARATHI
Before sorting
String [1] : PRAVEEN
String [2] : AVINASH
String [3] : CHETHANA
String [4] : BHARATHI
After sorting
String [1] : AVINASH
String [2] : BHARATHI
String [3] : CHETHANA
String [4] : PRAVEEN
```

LAB PROGRAM - 6

6) Write a java program to illustrate constructor.

AIM : Write a java program to illustrate constructor.

PROGRAM

```
class Constructor
{
        int a,b,c;
        Constructor(int x,int y)
        {
                a=x;
                b=y;
                c=0;
        }
        void display()
        {
                c=a+b;
                System.out.println("Addition : "+c);
                c=a-b;
                System.out.println("Subtraction : "+c);
                c=a*b;
                System.out.println("Multiplication : "+c);
                c=a/b;
                System.out.println("Division : "+c);
        }
}
class ExConstructor
{
        public static void main(String args[])
        {
                Constructor c=new Constructor(10,2);
                c.display();
        }
}
```


TEST DATA AND OUTPUT

```
C:\java>java ExConstructor
Addition : 12
Subtraction : 8
Multiplication : 20
Division : 5
```

LAB PROGRAM - 7

7) Write a java program to illustrate Method overloading.

AIM : Write a java program to illustrate Method overloading.

PROGRAM

```
class ExOver
{
     void volume(int s)
     {
          int vol;
          vol=s*s*s;
          System.out.println("Volume of a cube : "+vol);
     }
     void volume(float r,int h)
     {
          float vol;
          vol=3.14f *r*r*h;
          System.out.println("Volume of a Cylinder : "+vol);
     }
     void volume(int l,int b,int h)
     {
          int vol=l*b*h;
          System.out.println("Volume of a Rectangular Box:"+vol);
     }
}
class MethodOverLoading
{
     public static void main(String args[])
     {
          ExOver obj=new ExOver();
          obj.volume(5);
          obj.volume(3.5f ,8);
          obj.volume(5,7,9);
     }
}
```


TEST DATA AND OUTPUT

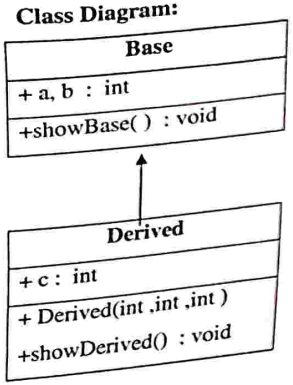
```
C:\java>java MethodOverLoading
Volume of a cube : 125
Volume of a Cylinder : 307.72003
Volume of a Rectangular Box : 315
```

LAB PROGRAM - 8

8) Write a java program to illustrate Single Inheritance.

AIM: Write a java program to illustrate Single Inheritance.

Class Diagram:



PROGRAM

```
class Base
{
    public int a,b;
    void showBase()
    {
        System.out.println(" Base Class \n a="+a+"\t b="+b);
    }
}
class Derived extends Base
{
    int c;
    Derived(int x,int y,int z)
    {
        a=x;
        b=y;
        c=z;
    }
    void showDerived()
```



```
    {
        System.out.println(" Derived Class \n c = "+c);
    }
}
class SingleInheritance
{
    public static void main(String args[])
    {
        Derived d=new Derived(2,3,4);
        d.showBase();     //calling base class function through derived class object
        d.showDerived();
    }
}
```

TEST DATA AND OUTPUT

C:\java>java SingleInheritance

Base Class

a = 2 b = 3

Derived Class

c = 4

LAB PROGRAM - 9

9) Write a java program to illustrate Multi Level Inheritance.

AIM: Write a java program to illustrate Multi Level Inheritance.

PROGRAM

```
class Student
{
        int rollno;
        void getNo(int a)
        {
          rollno=a;
        }
        void putNo()
        {
          System.out.println("Rollno : "+rollno);
        }
}
class Test extends Student
{
        float sub1,sub2;
        void getMarks(float x,float y)
        {
          sub1=x;
          sub2=y;
        }
        void putMarks()
        {
          System.out.println("Marks in sub1 : "+sub1);
          System.out.println("Marks in sub2 : "+sub2);
        }
}
class Result extends Test
{
        float total;
        void display()
```



```
        {
          total=sub1+sub2;
          putNo();
          putMarks();
          System.out.println("Total : "+total);
        }
}
class MultiLevel
{
        public static void main(String args[])
        {
          Result obj=new Result();
          obj.getNo(10);
          obj.getMarks(90,90.5f);
          obj.display();
        }
}
```

TEST DATA AND OUTPUT

C:\java>java MultiLevel

Rollno : 10

Marks in sub1 : 90.0

Marks in sub2 : 90.5

Total : 180.5

LAB PROGRAM - 10

10) Write a java program to illustrate Multiple Inheritance using interface.

AIM: Write a java program to illustrate Multiple Inheritance using interface.

PROGRAM

```
interface Exam       // interface definition
{
     void percentage();
}
class Student
{
     String name;
     int roll_no,mark1,mark2;
     Student(String n, int r, int m1, int m2)
     {
          name=n;
          roll_no=r;
          mark1=m1;
          mark2=m2;
     }
     void display()
     {
          System.out.println ("Name of Student:"+name);
          System.out.println ("Roll No. of Student:"+roll_no);
          System.out.println ("Marks of Subject 1:"+mark1);
          System.out.println ("Marks of Subject 2:"+mark2);
     }
}
class Result extends Student implements Exam
{
     Result(String n, int r, int m1, int m2)
     {
          super(n,r,m1,m2);
     }
     public void percentage()
```



```
     {
          int total=(mark1+mark2);
          float percent=total*100/200;
          System.out.println ("Percentage: "+percent+"%");
     }

}
class MultipleInheritance
{
     public static void main(String args[])
     {
          Result R = new Result("ZUNAID",101,93,84);
          R.display();
          R.percentage();
     }
}
```

TEST DATA AND OUTPUT

C:\java>java MultipleInheritance

Name of Student: ZUNAID

Roll No. of Student: 101

Marks of Subject 1: 93

Marks of Subject 2: 84

Percentage: 88.0%

LAB PROGRAM - 11

11) Write a java program to illustrate Method Overriding.

AIM: Write a java program to illustrate Method Overriding.

PROGRAM

```
class First
{
   void display()
   {
      System.out.println("Calling Super class display method");
   }
}
class Second extends First
{
      void display()          //Overriding method
      {
        super.display();   //Calling Super class display method
        System.out.println("Calling Sub class display method");
      }
}
class MethodOverride
{
     public static void main(String args[])
     {
          Second obj=new Second();
          obj.display();          //Calling Sub class display method

     }
}
```

TEST DATA AND OUTPUT

C:\java>java MethodOverride

Calling Super class display method

Calling Sub class display method

LAB PROGRAM - 12

12) Write a java program to read number of elements in the vector and check whether duplicate values are available or not.

AIM : Write a java program to read number of elements in the vector and check whether duplicate values are available or not.

PROGRAM

```
import java.io.*;
import java.lang.*;
import java.util.*;
class Vect
{
    public static void main(String args[])throws IOException
    {
        BufferedReader br=new BufferedReader(new
             InputStreamReader(System.in));
        String str;
        int n,i,j,c;
        System.out.println("Enter No.of values to read : ");
        n=Integer.parseInt(br.readLine());
        Vector v=new Vector(n);
        System.out.println("Enter vector values");
        for(i=0;i<n;i++)
        {
             str=br.readLine();
             int iobj;
             iobj=Integer.valueOf(str);
             v.addElement(iobj);
        }
        Integer aobj[]=new Integer[n];
        v.copyInto(aobj);
        c=0;
        for(i=0;i<n-1;i++)
        {
             for(j=i+1;j<n;j++)
```

```
                {
                    if(aobj[i].intValue()==aobj[j].intValue())
                    {
                        c=1;
                        System.out.println("Duplicate values
                            in the vector  : "+aobj[i]);
                    }
                }
            }
        if(c==0)
          System.out.println("Duplicate values are not in the vector");
        }
}
```

TEST DATA AND OUTPUT

1) C:\java>java Vect
Enter No.of values to read :
5
Enter vector values
25
35
12
56
25
Duplicate values in the vector : 25

2) C:\java>java Vect
Enter No.of values to read :
5
Enter vector values
14
2
3
5
6
Duplicate values are not in the vector

LAB PROGRAM - 13

13) Write a java program to implement a package on arithmetic operations.

AIM: Write a java program to implement a package on arithmetic operations

PROGRAM -1 **Arithmetic.java**

```
package arithmetic;          //user defined package
public class Arithmetic
{
    public double add(double a,double b)
    {
        return(a+b);
    }
    public double sub(double a,double b)
    {
        return(a-b);
    }
    public double mul(double a,double b)
    {
        return(a*b);
    }
    public double div(double a,double b)
    {
        return(a/b);
    }
    public double mod(double a,double b)
    {
        return(a%b);
    }
}
```

Note : Compile c:\java>javac –d . Arithmetic.java

PROGRAM -2 **UserPackage.java**

```
import java.io.*;
import arithmetic.Arithmetic;          //importing user defined package
```

```
class UserPackage
{
     public static void main(String args[])throws IOException
     {
          double a,b;
          int opt;
          Arithmetic obj=new Arithmetic();
          BufferedReader br=new BufferedReader(new
               InputStreamReader(System.in));
          System.out.println("Enter a and b values : ");
          a=Double.parseDouble(br.readLine());
          b=Double.parseDouble(br.readLine());
          System.out.println("\n1.Addition \n 2.Subtraction \n
           3.Multiplication \n 4.Division\n5.Modulodivision");
          System.out.println("\n Enter your choice");
          opt=Integer.parseInt(br.readLine());
          switch(opt)
          {
          case 1:
            System.out.println("Addition:"+obj.add(a,b));
            break;
         case 2:
            System.out.println("Substraction:"+obj.sub(a,b));
            break;
        case 3:
           System.out.println("Multiplication:"+obj.mul(a,b));
           break;
      case 4:
         System.out.println("Division:"+obj.div(a,b));
         break;
    case 5:
      System.out.println("Modulodivision:"+obj.mod(a,b));
     break;
  default :
```



```
                System.out.println("Invalid choice");
            }
        }
}
```

TEST DATA AND OUTPUT

1)
```
C:\java>java UserPackage
Enter a and b values :
10
5
1.Addition
2.Subtraction
3.Multiplication
4.Division
5.Modulodivision
Enter your choice
1
Addition : 15.0
```

2)
```
C:\java>java UserPackage
Enter a and b values :
10
5
1.Addition
2.Subtraction
3.Multiplication
4.Division
5.Modulodivision
Enter your choice
4
Division : 2.0
```

LAB PROGRAM - 14

14) Write a java program to implement exception handling based on the marks

AIM : Write a java program to implement exception handling based on the marks

PROGRAM

```
import java.io.*;
class MyException extends Exception
{
     public MyException(String message)
     {
          super(message);
     }
}
class MarksException
{
     public static void main(String args[])throws Exception
     {
          int sno,marks;
          String name;
          BufferedReader br=new BufferedReader(new
               InputStreamReader(System.in));
          try
          {
               System.out.println("Enter Student No. ");
               sno=Integer.parseInt(br.readLine());
               System.out.println("Enter Student Name ");
               name=br.readLine();
               System.out.println("Enter Student Marks ");
               marks=Integer.parseInt(br.readLine());
               if(marks>100)
                    throw new MyException("Invalid marks should
                         be between 0-100");
               System.out.println("Student No : "+sno);
               System.out.println("Student Name: "+name);
               System.out.println("Student Marks: "+marks);
```



```
          }
          catch(MyException e)
          {
               System.out.println("Caught Marks out of bound exception");
               System.out.println(e.getMessage());
          }
     }
}
```

TEST DATA AND OUTPUT

1) C:\java>java MarksException
Enter Student No.
101
Enter Student Name
AKHIL
Enter Student Marks
89
Student No : 101
Student Name: AKHIL
Student Marks: 89

2) C:\java>java MarksException
Enter Student No.
102
Enter Student Name
THARUN
Enter Student Marks
110
Caught Marks out of bound exception
Invalid marks should be between 0-100

LAB PROGRAM - 15

15) Write a java program to implement thread concept.

AIM : Write a java program to implement thread concept.

PROGRAM

```
class Even extends Thread
{
      int i;
      public void run()
      {
           for(i=1;i<=6;i++)
           {
                if(i%2==0)
                     System.out.println("Even Number"+i);
           }
      }
}
class Odd extends Thread
{
     int i;
     public void run()
     {
          for(i=1;i<=6;i++)
          {
               if(i%2!=0)
                    System.out.println("Odd Numbers:"+i);
          }
     }
}
class ThreadNumbers
{
     public static void main(String args[])
     {
          Even e=new Even();
          Odd o=new Odd();
```



```
          e.start();
          o.start();
     }
}
```

TEST DATA AND OUTPUT

```
C:\java>java ThreadNumbers
Even Number2
Odd Numbers:1
Even Number4
Odd Numbers:3
Even Number6
Odd Numbers:5
```

LAB PROGRAM - 16

16) Write a java program to Prompt for the cost price and selling price of an item and display the profit or loss percentage.

AIM : Write a java program to Prompt for the cost price and selling price of an item and display the profit or loss percentage.

PROGRAM – 1 **ItemApplet.java**

```
import java.awt.event.*;
import java.awt.event.ActionListener.*;
import java.awt.*;
import java.applet.*;
public class ItemApplet extends Applet implements
ActionListener
{
    Label l1,l2,l3;
    TextField t1,t2,t3;
    Button b;
    public void init()
    {
        setBackground(Color.cyan);
        l1=new Label("Cost Price : ");
        t1=new TextField(20);
        l2=new Label("Selling Price : ");
        t2=new TextField(20);
        l3=new Label("Result : ")  ;
        t3=new TextField(20);
        b=new Button("Profit/Loss");
        add(l1);
        add(t1);
        add(l2);
        add(t2);
        add(l3);
        add(t3);
        add(b);
        b.addActionListener(this);
```



```
    }
    public void actionPerformed(ActionEvent e)
    {
        double cp,sp,res,profit,loss;
        if(e.getSource()==b)
        {
            cp=Double.parseDouble(t1.getText());
            sp=Double.parseDouble(t2.getText());
            if(cp<sp)
            {
              res=sp-cp;
              profit=(res/cp)*100;
              t3.setText(String.valueOf(profit)+"%Profit");
            }
            else
            {
                res=cp-sp;
                loss=(res/cp)*100;
                t3.setText(String.valueOf(loss)+"% Loss");
            }
        }
    }
}
```

PROGRAM – 2 **Main.html**

```
<applet code="ItemApplet" width=280 height=280>
</applet>
```

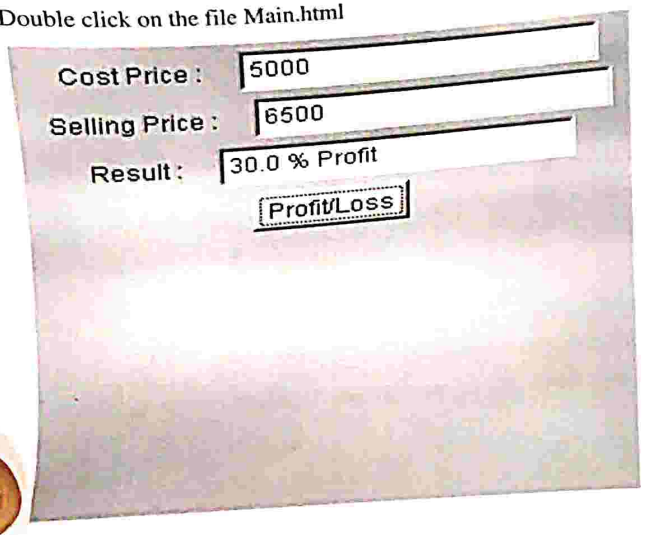
TEST DATA AND OUTPUT

For Compilation :

C:\java>javac ItemApplet.java

For Execution :

Double click on the file Main.html



LAB PROGRAM - 17

17) Write a java program to wish the user at different time intervals by using applet.

AIM : Write a java program to wish the user at different time intervals by using applet.

PROGRAM - 1 **WishUser.java**

```
import java.applet.*;
import java.util.*;
import java.awt.*;
public class WishUser extends Applet
{
     int hours=0;
     String str=null;
     Calendar c=null;
     public void init()
     {
          c=new GregorianCalendar();
          hours=c.get(Calendar.HOUR_OF_DAY);
          System.out.println("---hours: "+hours);
     }
     public void paint(Graphics g)
     {
          g.drawString(str,10,100);
     }
     public void start()
     {
          if(hours<=12)
               str="GOOD   MORNING";
          else if(hours>12 && hours<=16)
               str="GOOD  AFTERNOON";
          else if(hours>16 && hours<=20)
               str="GOOD   EVENING";
          else
               str="GOOD   NIGHT";
     }
}
```

PROGRAM - 2 Applet.html

```
<html>
      <head>
           <title>New Document</title>
      </head>
      <body>
           <centre><h1>Welcome to the USER</h1><br>
           <applet code="WishUser.class" width=300 height=300>
           </applet>
           </centre>
      </body>
</html>
```

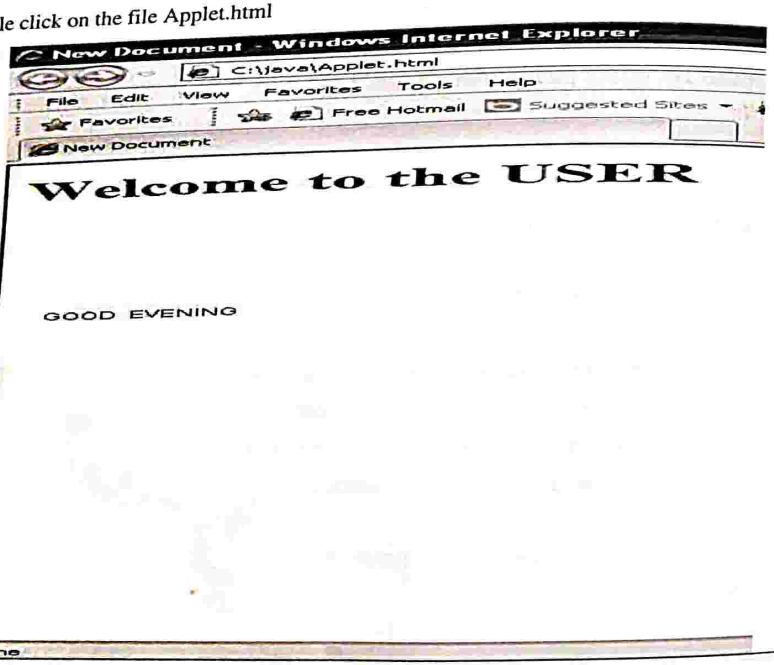
Test data and Output

For Compilation :

C:\java>javac WishUser.java

For Execution :

Double click on the file Applet.html



LAB PROGRAM - 18

18) Write a Java program to Shuffle the list elements using all the possible Permutations.

AIM : Write a Java program to Shuffle the list elements using all the possible Permutations.

PROGRAM

```
import java.util.Arrays;
import java.util.Collections;
import java.util.List;
import java.util.Scanner;
public class ShuffleList
 {
   public static void main(String args[])
   {
        String list[]=new String[5];
        Scanner in=new Scanner(System.in);
        int n,i;
        System.out.println("Enter No. of strings : ");
        n=in.nextInt();
        in.nextLine();
        for(i=0;i<n;i++)
        {
             System.out.println(" Enter String"+(i+1)+":");
             list[i]=in.nextLine();
        }
        List l1=Arrays.asList(list);
        System.out.println("The list before Shuffling");
        System.out.println(l1);
        Collections.shuffle(l1);
        System.out.println("The list After Shuffling");
        System.out.println(l1);
   }
}
```

TEST DATA AND OUTPUT

C:\java>java ShuffleList

Enter No. of strings : 5

```
Enter String 1 :        COKE
Enter String 2 :        7UP
Enter String 3 :        PEPSI
Enter String 4 :        SPRITE
Enter String 5 :        SLICE
The list before Shuffling
[COKE, 7UP, PEPSI, SPRITE, SLICE]
The list After Shuffling
[PEPSI, 7UP, SLICE, SPRITE, COKE]
C:\java>
```