COMMAND LINE ARGUMENTS

EX.NO:1

DATE:

AIM:

To write a java program to display the count of all command line arguments and list each in a line.

REQUIREMENTS:

Computer with Pentium/Dual Core Processors

NETBEANS 8.2

ALGORITHM

1. Start the program

2.Declare &initialize variables

3. Calculate length using args.length

4.Print arguments using while loop

5.Stop the program

Program

```
import java.io.*;
public class Cmdlinearg
{
public static void main(String[] args)
{
int count,i=0;
count=args.length;
System.out.println("total number of args"+count);
while(count>0)
{
System.out.println("argno"+(i+1)+"is"+args[i]);
i++;
count--;

   }
   }
   }
```

output

total number of args4

argno1 is 8

argno2 is 6

argno3 is 4

argno4 is 2

Result

Thus the program for display the count of all command line arguments is compiled, executed and the output is verified.

SUM OF DIGITS

EX.NO:2

DATE:

AIM:

To write a java program to find out sum of digits of given number

REQUIREMENTS:

Computer with Pentium/Dual Core Processors

NETBEANS 8.2

ALGORITHM

1.Start the program

2.Declare & initialise variables

3.Get one digit from number i using a=i%10;

4.Sum the number using sum=sum+a;

5.Calculate i=i/10

6.Print the result

7.Stop the program.

Program

```
import java.io.*;
import java.util.*;

public class Sumdigit {
   public static void main(String[] args) {
   int num,a=0,sum=0;
   Scanner scan=new Scanner(System.in);
   System.out.println("Enter a number");
   num=scan.nextInt();
   int i=num;
   while(i!=0)
   {
      a=i%10;
      i=i/10;
      sum=sum+a;
       }
   System.out.println("Sum of the Digits of" +num+"is:"+sum);
   }

}
```

Output

```
Enter a number
456
Sum of the Digits of456is:15
```

Result

Thus the program to find out sum of digits of given number is compiled, executed and the output is verified.

MULTIPLICATION TABLE

EX.NO:3

DATE:

AIM:

To write a java program to display multiplication table in row, columnformat.

REQUIREMENTS:

Computer with Pentium/Dual Core Processors

NETBEANS 8.2

ALGORITHM

1.Start the program

2.Declare variables

3.Get input from user

4.using for loop calculate & print the multiplication table

5.Stop the program

Program

```
import java.io.*;
import java.util.*;

public class Multable {

      public static void main(String[] args) {

       int n;
       Scanner scan=new Scanner(System.in);
       System.out.println("Enter the table number");
       n=scan.nextInt();
       System.out.println("Multiplication table for"+n);
       System.out.println("-------------------------");
for(int i=1;i<=10;i++)
{
   System.out.format("%2dx%d=%3d\n", i,n,i*n);

}
   }

}
```

Output

```
Enter the table number

4
Multiplication table for4
-------------------------
 1x4=  4
 2x4=  8
```

3x4= 12
4x4= 16
5x4= 20
6x4= 24
7x4= 28
8x4= 32
9x4= 36
10x4= 40

RESULT:

Thus the program to display multiplication table in row,column format is compiled, executed and the output is verified.

PRIME NUMBER

EX.NO:4a

DATE:

AIM:

To write a java program to find whether the given number is prime or not.

REQUIREMENTS:

Computer with Pentium/Dual Core Processors

NETBEANS 8.2

ALGORITHM

1.Start the program

2.Declare the necessary variables

3.Get input from user

4.Using for loop and if statement to check whether it is a prime or not

5.Print the prime number

6.Stop the program.

Program

```
import java.io.*;
import java.util.*
public class primecheck
{
public static void main(String[] args)
{
int flag=0;
Scanner scan=new Scanner(System.in);
System.out.println("Enter a number to check for prime");
int num=scan.nextInt();
for(int i=2;i<num;i++)
{
if(num%i==0)
{
System.out.println(num+"is not a prime number");
flag=1;
break;
}
}
```

```
if(flag==0)

System.out.println(num+"is a prime number");

}

}
```

Output

```
Enter a number to check for prime
5
5is a prime number

Enter a number to check for prime
4
4is not a prime number
```

RESULT:

Thus the program prime number checking is compiled, executed and the output is verified.

PRIME NUMBER GIVEN RANGE OF NUMBER

EX.NO:4b

DATE:

AIM:

To write a java program to display all rime number in a given range of numbers.

REQUIREMENTS:

Computer with Pentium/Dual Core Processors

NETBEANS 8.2

ALGORITHM

1.Start the program

2.Declare the necessary variables

3.Get input from user

4.using nested for loop and if statement to check whether it is a prime or not

5.Print the given range of prime number

6.Stop the program.

Program

```
import java.io.*;
public class primerange
{
public static void main(String[] args)
{
int r=1
System.out.println("prime list");
for(int i=2;i<=100;i++)
{
for(int j=2;j<=i/2;j++)
{
r=i/j;
if(r==0)
break;
}
if(r!=0)
System.out.print(+i+"");
}
}
}
```

Output

prime list
2357111317192329313741434753596167717379838997

RESULT:

Thus the program for display all prime numbers in a given range of number is compiled, executed and the output is verified.

ARRAY OF INTEGERS USING EXCEPTION

EX.NO:5

DATE:

AIM:

To write a java program to create an array of integers and accept number check whether it exists in the array or not using exception.

REQUIREMENTS:

Computer with Pentium/Dual Core Processors

NETBEANS 8.2

ALGORITHM

1.Start the program

2.Declare the array variables

3.Get array input from user

4.Get one input to search in array, if it is found number will display else the number not found using execption

5.Stop the program.

program

```
import java.io.*;
import java.util.*;
public class Arrexcep {

   public static void main(String[] args) {
 // TODO code application logic here
       try
       {
          int n;
          boolean found=false;
          int s1[]=new int[5];
          Scanner s =new Scanner(System.in);
         System.out.println("Enter 5 numbers to insert in the array");
         for(int i=0;i<=4;i++)
         {
            s1[i]=s.nextInt();
         }
          System.out.println("Enter a number to search in the array");
          n=s.nextInt();
          int num=n;
          for(int j=0;j<=4;j++)
          {
            if(num==s1[j])
            {
               found=true;
               break;

            }
            else
               found=false;

          }
 if(found)
             System.out.println("The number"+n+"is found in the array");
          else
             throw new MyExcep(n);

       }
       catch(MyExcep e)
       {
       }
```

```
    }
  }
```

Myexcep.java

```
package arrexcep;

public class MyExcep extends Exception
{
MyExcep(int value)
{
   System.out.println("The number"+value+"is not found in the array ");
}
}
```

Output

```
Enter 5 numbers to insert in the array
90
50
60
78
92
Enter a number to search in the array
50
The number50is found in the array


Enter 5 numbers to insert in the array
96
74
85
62
98
Enter a number to search in the array
14
The number14is not found in the array
```

RESULT:

Thus the program create an array of integers using exception is compiled, executed and the output is verified.

IMPLEMENT STACK USING VECTOR

EX.NO:6

DATE:

AIM:

To write a java program to implement stack using Vector class or ArrayList

REQUIREMENTS:

Computer with Pentium/Dual Core Processors

NETBEANS 8.2

ALGORITHM

1.Start the program

2.Declare the array variables

3.Get input from user

4.Declare Vector Class, Constructor and methods

5.Read ch

6.if ch=2 then top is in incremented to push element

7.if ch=2 then top is in incremented to pop element

8.if ch=3 then exit from the menu.

9.Stop the program.

Program

```
import java.io.*;
public class Mystack {

   Vector myvector;
   public Mystack()
{
      myvector=new Vector();

   }
   public void push(Object obj)
   {
       myvector.add(obj);

   }
   public Object pop()
   {
      Object obj=null;
      if(myvector.size()>0)
      {
         obj=myvector.elementAt(myvector.size()-1);
         myvector.removeElementAt(myvector.size()-1);

      }
      else
         System.out.println("Stack underflow");
      return obj;
   }
   public Object peek()
   {
      Object obj=null;
      if(myvector.size()>0)
         obj=myvector.elementAt(myvector.size()-1);
      else
         System.out.println("Stack underflow");
      return obj;
   }
   public static void main(String[] args) {
Mystack stack=new Mystack();
int ch;
do
{
   System.out.println("Menu");
   System.out.println("1.Push");
   System.out.println("2.Pop");
   System.out.println("3.Exit");
```

```
    Scanner scan=new Scanner(System.in);
    ch=scan.nextInt();
    switch(ch)
    {
       case 1:
          System.out.println("Enter the item to be pushed");
          int n=scan.nextInt();
          stack.push(n);
          System.out.println("The top of stack now is "+stack.peek());
          break;
       case 2:
          stack.pop();
          System.out.println("the top of stack now is"+stack.peek());
          break;



    }
}
while(ch<3);

    }

}
```

Output

```
Menu
1.Push
2.Pop
3.Exit

1
Enter the item to be pushed
58
The top of stack now is 58
Menu
1.Push
2.Pop
3.Exit
2
Stack underflow
the top of stack now isnull
Menu
1.Push
2.Pop
3.Exit
3
```

RESULT:

Thus the program implementing stack using Vector is compiled, executed and the output is verified.

WINDOWS APPLICATION

EX.NO:7

DATE:

AIM:

To write a java program to execute any given windows application and report the exit status of the application.

REQUIREMENTS:

Computer with Pentium/Dual Core Processors

NETBEANS 8.2

ALGORITHM

1.Start the program

2.Using try block create process, if it is found error an exception thrown and is caught by catch block.

3.Stop the program.

Program

```
import java.io.*;

public class Appexec {


   public static void main(String[] args) {
   try
   {
```

```
        System.out.println("creating process");
        Process process=Runtime.getRuntime().exec("/root/Desktop/c.odt");
        process.waitFor();
        System.out.println("Program terminated");

    }
    catch(Exception e)
    {
        System.out.println("----------Exception-----\n"+e);
    }
    }
}
```

Output

```
creating process
Program terminated
```

RESULT:

Thus the program finding windows application exit status is compiled, executed and the output is verified.

FILE OR DIRECTORY

EX.NO:8

DATE:

AIM:

To write a java program to get a file name at run time and check for its existence check whether it is a dirctory or normal file.

REQUIREMENTS:

Computer with Pentium/Dual Core Processors

NETBEANS 8.2

ALGORITHM

1.Start the program

2.Create a class chkfiledir

3.Creat an object fo Scanner s

4.Get the path of the file or directory in file object

5.Check if it is a file or directory using isFile(),isDirectory().

6.Print the file or directory, size and last modified date.

7.Stop the program.

Program

```
import java.io.*;
```

```
import java.util.*;

public class Fileordir
{


   public static void main(String[] args)
   {
   Scanner s=new Scanner(System.in);
   System.out.println("Enter the path for file or directory");
   String path=s.next();
   File file=new File(path);
  boolean isFile=file.isFile();
  if(isFile)
       System.out.println(file.getPath()+"is a file");


  else

     System.out.println(file.getPath()+"is not a file");

  boolean isDirectory =file.isDirectory();
  if(isDirectory)

     System.out.println(file.getPath()+"is a directory");



     else

     System.out.println(file.getPath()+"is not a directory");


   System.out.println(file.length()+"bytes");
   System.out.println("last modified"+new Date(file.lastModified()));


  }
}
```

Output

```
Enter the path for file or directory
/root/Desktop/c
/root/Desktop/cis a file
/root/Desktop/cis not a directory
8bytes
last modifiedThu Nov 16 23:35:53 IST 2017

Enter the path for file or directory
/root
/rootis not a file
/rootis a directory
4096bytes
last modifiedFri Nov 17 00:10:00 IST 2017
BUILD SUCCESSFUL (total time: 5 seconds)
```

RESULT:

Thus the program directory or file is compiled, executed and the output is verified.

COPYING ONE FILE TO ANOTHER FILE

EX.NO:9

DATE:

AIM:

To write a java program to copy a file to another file using java.io package class.

REQUIREMENTS:

Computer with Pentium/Dual Core Processors

NETBEANS 8.2

ALGORITHM

1.Start the program

2.Create a class filecopy

3.Declare an integer c, create an object for the FileInuputStream in and also create for FileOutputStream out.

4.Instantiate memory location for in and get input from commad line

5.Read byte from the inputfile in fill the end of the file and message will display file copied.

6.Close in & out object.

7.Stop the program.

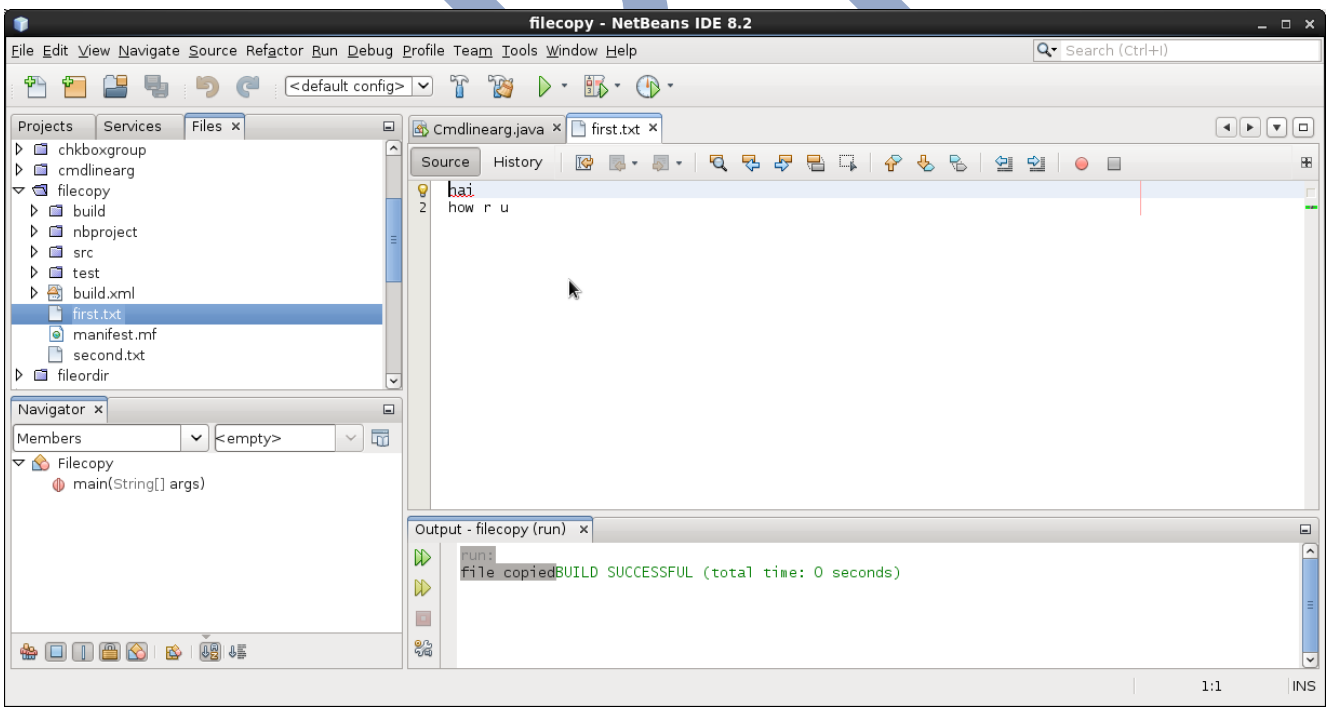
Program

```
import java.io.*;
```

```
public class Filecopy
{


    public static void main(String[] args) throws IOException {
    File inputfile=new File("first.txt");
    File outputfile=new File("second.txt");
    FileInputStream in = new FileInputStream(inputfile);
    FileOutputStream out=new FileOutputStream(outputfile);
    int c;
    while((c=in.read())!=-1)
    {
       out.write(c);

    }
    System.out.print("file copied");
    in.close();
    out.close();
    }

}
```

Output

file copied



RESULT:

Thus the program copying one file to another file is compiled, executed and the output is obtained.

FILE WORD COUNT

EX.NO:10

DATE:

AIM:

To write a java program to get a file at runtime and display the number of lines,words and characters in that file.

REQUIREMENTS:

Computer with Pentium/Dual Core Processors

NETBEANS 8.2

ALGORITHM

1.Start the program

2.Create a class wordcount

3.Declare the variables, create an object for the BufferedReader buf.

4.Instantiate memory location for buf and get input from command line

5.Read byte by byte from the buf in fill the end of the file.

6.print number of words, lines and characters

7.Stop the program.

Program

```
import java.io.*;
import java.util.*;


public class Wordcount {
```

```
   public static void main(String[] args) throws Exception
   {
      int ccount=0;
      int wcount=0;
      int lcount=0;
      String s;
      StringTokenizer st;
    BufferedReader buf=new BufferedReader(new InputStreamReader(System.in));
      System.out.print("enter filename");
      s=buf.readLine();
      buf=new BufferedReader(new FileReader(s));
      while((s=buf.readLine())!=null)
      {
         lcount++;
         st=new StringTokenizer(s," ");
         while(st.hasMoreTokens())
         {
            wcount++;
            s=st.nextToken();
            ccount+=s.length()+1;

         }
      }

      System.out.println("Character count:"+ccount);
      System.out.println("Word Count"+wcount);
      System.out.println("Line Count"+lcount);
      buf.close();
   }

}
```

Output
enter filename/root/Desktop/c
Character count:8
Word Count1
Line Count1

RESULT:

Thus the program file counting for number of lines, words and characters is compiled, executed and the output is obtained.

DISPLAY X,Y CO-ORDINATES IN LABELS USING FRAME

Ex.No :11
Date :

AIM :

To write a java program to create a Frame with two labels. At runtime display x and y co-ordinates of mouse pointer in the Labels.

REQUIREMENTS:

- Computer with Pentium/Dual Core Processors
- NETBEANS 8.2

ALGORITHM

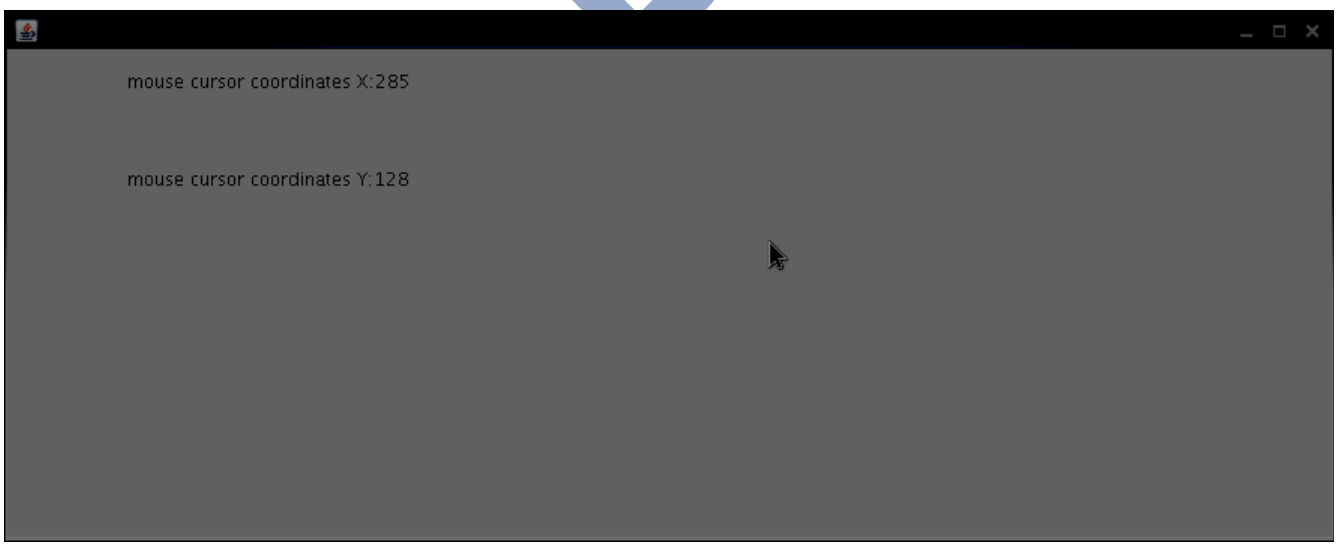
1.Start the program
2.create a class mouseevent2 which extends Frame class
3.Declare the variables, create and instantiate label
4.addMouseListener(),and addWindowListener method to listen the mouse event
5.add labels and set the coordinates using setText.
6.Using Mouseclicked event to get the coordinates of x,y place it in labels.
7.Stop the Program.

Program

```
import java.applet.*;
import java.awt.event.*;
import java.awt.*;
public class mouseevent2 extends Frame
{
Label l=new Label();
Label l1=new Label();
int x=0;
int y=0;
public void init() {
addMouseListener(new mymouselistener());
addWindowListener(new MyWindowAdapter());
add(l);
add(l1);
}

public void paint(Graphics g)
{
l.setText("mouse cursor coordinates X:"+x);
l.setBounds(80,35,190,25);
l1.setText("mouse cursor coordinates Y:"+y);
l1.setBounds(80,100,190,25);
}
```

```
public class mymouselistener extends MouseAdapter
{
public void mouseClicked(MouseEvent e)
    {
       x=e.getX();
        y=e.getY();
        repaint();
     }
}
class MyWindowAdapter extends WindowAdapter
{
   public void windowClosing(WindowEvent we)
    {
        System.exit(0);
    }
}
public static void main(String p[])
{
mouseevent2 me2=new mouseevent2();
me2.setVisible(true);
me2.init();
me2.setLayout(null);
    }
}
```

Output



Result

Thus the program display coordinates of x and y in labels using frame is compiled, executed and the required output is obtained.

SET BACKGROUND USING CHECKBOX

Ex:12
Date :
AIM :

To write a java program to create a Frame and Checkbox group with five checkboxes with labels as Red,Green,Blue,Yellow and White. At runtime change the background color of Frame using Checkboxes.

REQUIREMENTS:

- Computer with Pentium/Dual Core Processors
- NETBEANS 8.2

ALGORITHM

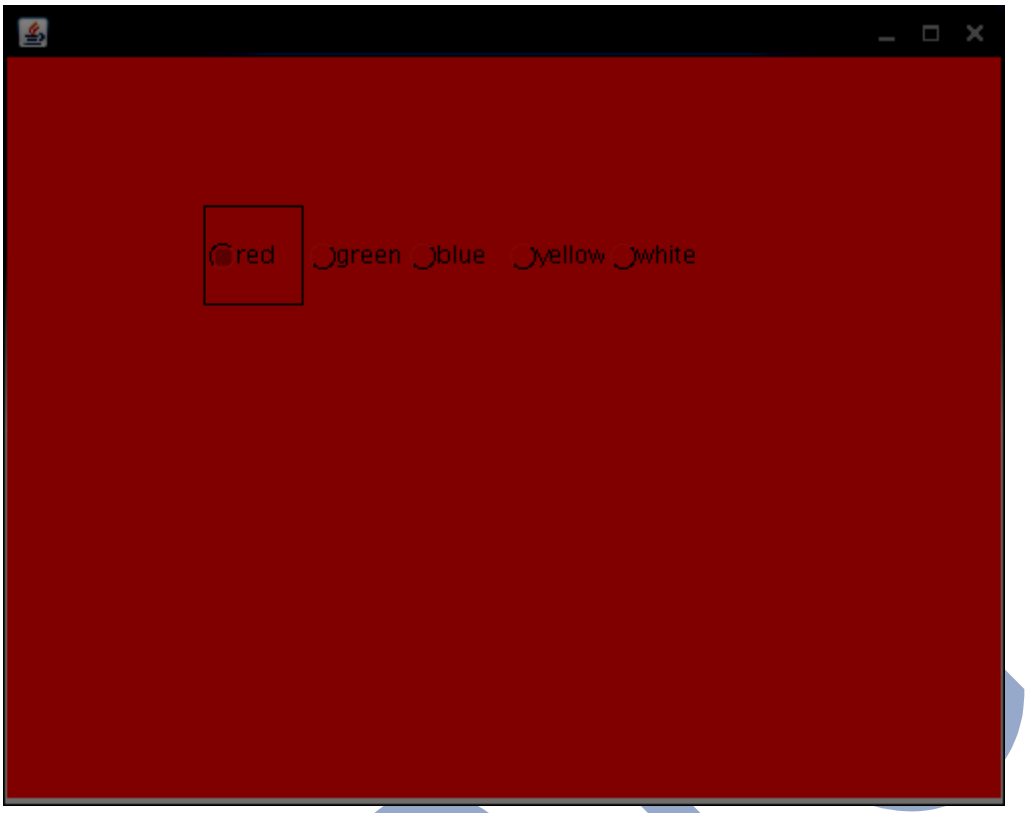
1.Start the program
2.create a class chkboxg which extends Frame implements ItemListener
3.Declare Checkbox redcheckbox,greencheckbox,bluecheckbox,yellowcheckbox,whitecheckbox and Checkboxgroup colorgroup.
4.Call add() method to add the components redcheckbox, greencheckbox, bluecheckbox, yellowcheckbox, whitecheckbox
5.call additemListener() method to listen the checkbox as action
6.Define itemStateChanged () method to set the background color.
7.Stop the Program.

Program

```
import java.applet.*;
import java.applet.Applet;
import java.awt.*;
import java.awt.event.*;

public class chkboxg extends Frame implements ItemListener {
public void init() {
  }
   chkboxg()
   {
      setLayout(null);
      CheckboxGroup colorgroup=new CheckboxGroup();
      Checkbox redcheckbox=new Checkbox("red",colorgroup,false);
      redcheckbox.addItemListener(this);
      redcheckbox.setBounds(100,100,50,50);
      add(redcheckbox);
      Checkbox greencheckbox=new Checkbox("green",colorgroup,false);
      greencheckbox.addItemListener(this);
      greencheckbox.setBounds(150,100,50,50);
      add(greencheckbox);
      Checkbox  bluecheckbox=new Checkbox("blue",colorgroup,false);
      bluecheckbox.addItemListener(this);
      bluecheckbox.setBounds(200,100,50,50);
```

```
        add(bluecheckbox);
        Checkbox  yellowcheckbox=new Checkbox("yellow",colorgroup,false);
        yellowcheckbox.addItemListener(this);
        yellowcheckbox.setBounds(250,100,50,50);
        add(yellowcheckbox);
        Checkbox  whitecheckbox=new Checkbox("white",colorgroup,false);
        whitecheckbox.addItemListener(this);
        whitecheckbox.setBounds(300,100,50,50);
        add(whitecheckbox);
        addWindowListener(new WindowAdapter()
{
public void windowClosing(WindowEvent we)
{
System.exit(0);
}
});
    }
    public static void main(String[] args)
{
Frame f=new chkboxg();
f.setSize(500,400);
f.setLayout(null);
f.setVisible(true);
}
public void paint(Graphics g)
{
   repaint();
}

public void itemStateChanged(ItemEvent evt)
   {
       if(evt.getItem().equals("red"))
          setBackground(Color.red);
       else if(evt.getItem().equals("green"))
          setBackground(Color.green);
       else if(evt.getItem().equals("blue"))
          setBackground(Color.blue);
       else if(evt.getItem().equals("yellow"))
          setBackground(Color.yellow);
       else if(evt.getItem().equals("white"))
          setBackground(Color.white);
       repaint();

   }

  }
```

Output



Result
Thus the program set background using checkbox is compiled, executed and the required output is obtained.

SET BACKGROUND USING SCROLLBAR

Ex.No :13
Date :

AIM :

To write a java program to create a Frame with 3 Scrollbars representing the three basic colors RED,GREEN and BLUE. Change the background color of the frame using the values of Scrollbars.

REQUIREMENTS:

- Computer with Pentium/Dual Core Processors
- NETBEANS 8.2

ALGORITHM

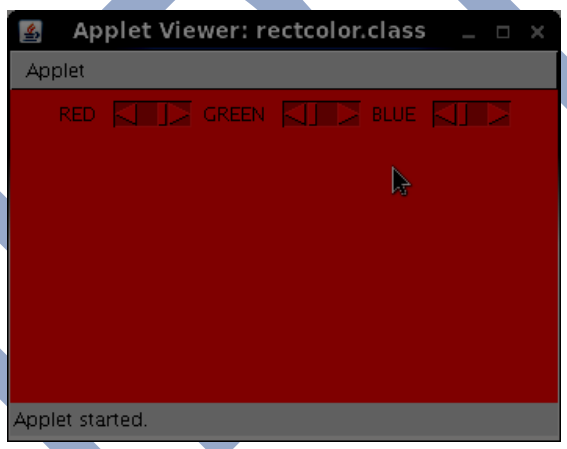
1.Start the program
2.create a class menuframe which extends Frame and implements the interface Adjustment items.
3.Declare the variables red,green,blue.
4.Declare the scrollbar s1,s2,s3
5.call add() method to set the frame and instantiate scrollbar s1,s2,s3
6.call addAdjustmentListener() method to listener the scrollbar action
7.Define Adjustmentvaluechanged() method to get the adjustment type of the corresponding scrollbar
8.call repaint()
9.Define paint() method to set the background color when the scrollbar are tracked.
10.Stop the Program.

Program

```
import java.applet.Applet;
import java.awt.*;
import java.awt.event.*;

public class menuframe extends Frame implements AdjustmentListener {
int red,green,blue;
Scrollbar s1,s2,s3;
Label l1,l2,l3;
menuframe()
{
 s1=new Scrollbar(Scrollbar.HORIZONTAL,0,0,0,255);
 s2=new Scrollbar(Scrollbar.HORIZONTAL,0,0,0,255);
 s3=new Scrollbar(Scrollbar.HORIZONTAL,0,0,0,255);
 s1.addAdjustmentListener(this);
 s2.addAdjustmentListener(this);
 s3.addAdjustmentListener(this);
l1=new Label("RED");
l2=new Label("GREEN");
l3=new Label("BLUE");
s1.setBounds(30,50,80,30);
s2.setBounds(130,50,80,30);
s3.setBounds(230,50,80,30);
add(l1);
add(s1);
add(l2);
add(s2);
add(l3);
add(s3);
addWindowListener(new WindowAdapter()
{
public void windowClosing(WindowEvent we)
{
System.exit(0);
```

```
}
});

}
public void adjustmentValueChanged(AdjustmentEvent e)
{
red=s1.getValue();
green=s2.getValue();
blue=s3.getValue();
repaint();
}
public static void main(String[] args)
{
Frame f=new menuframe();
f.setSize(500,400);
f.setLayout(null);
f.setVisible(true);
}
public void paint(Graphics g)
{
 setBackground(new Color(red,green,blue));
}
}
```

Output



Result

Thus the program for set background using scroll bar is compiled, executed and the required output is obtained.

SIMPLE AND COMPOUND INTEREST

Ex.No :14
Date :

AIM :

To write a java program to create An applet to calculate simple and compound interest by passing parameters through <param> tags of HTML file.

REQUIREMENTS:

- Computer with Pentium/Dual Core Processors
- NETBEANS 8.2

ALGORITHM

1.Start the program
2.create a class appmsga which extends Applet class
3.Declare the variables integer variable p,n,r si,ci and string variable prin,year,rates
4.Get the values of p,n,r using getParameter tag
5.calculate simple and compound interest
6.convert the integer values to string using toString () method
7.Display the values
8.Stop the Program.

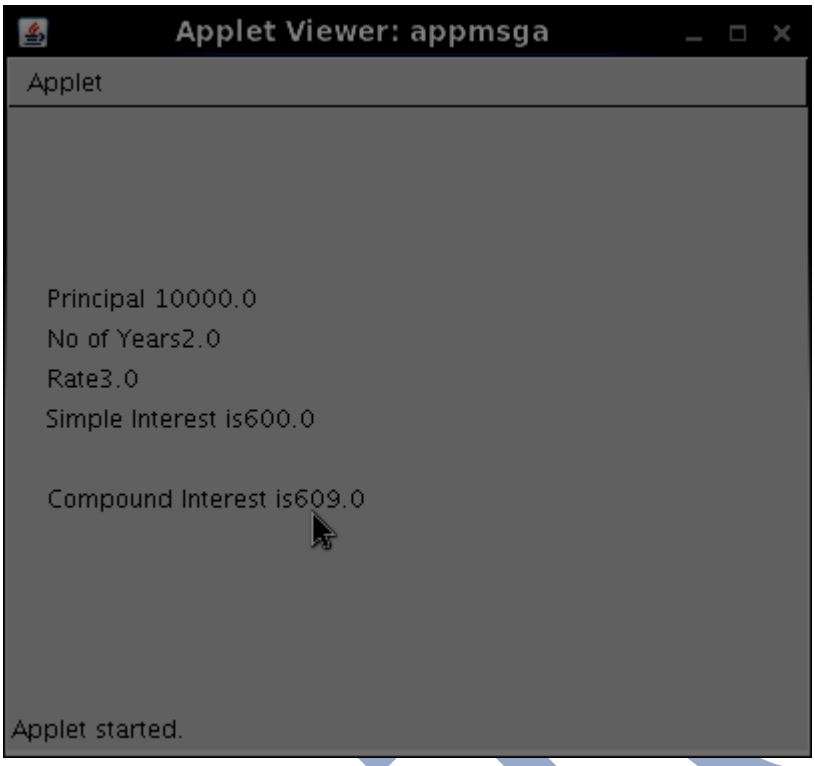
Program

```
import java.applet.Applet;
import java.awt.*;
import java.applet.*;
import java.lang.*;
import java.lang.Math.*;
import java.util.*;
public class appmsga extends Applet {
double p, n,r,si,ci;
String prin,year,rates;
public void init() {
p=Double.parseDouble(getParameter("principal"));
n=Double.parseDouble(getParameter("years"));
r=Double.parseDouble(getParameter("rate"));
prin=Double.toString(p);
year=Double.toString(n);
rates=Double.toString(r);
si=p*n*r/100;
ci=p*Math.pow((1+r/100),n)-p;
}
public void paint(Graphics g)
{
g.drawString("Principal "+ prin,20,100);
g.drawString("No of Years"+ year,20,120);
g.drawString("Rate"+rates,20,140);
g.drawString("Simple Interest is"+si,20,160);
g.drawString("Compound Interest is"+ci,20,200);
}
}
```

msga.html

```
<html>
<body>
<applet code="appmsga" width="400" height="300">
<param name="principal" value="10000">
<param name="years" value="2">
<param name="rate" value="3">
</applet>
</body>
</html>
```

Output



Result

Thus the program simple and compound interest is compiled, executed and the required output is obtained.

DRAW BAR CHART USING APPLET

Ex.No :15
Date :

AIM :

To write a java program to draw a bar chart for the marks scored in 5 subjects by a student using
Graphics object.

REQUIREMENTS:

- Computer with Pentium/Dual Core Processors
- NETBEANS 8.2

ALGORITHM

1.Start the program
2.Create an applet tag assign the class name width and height
3.Close an applet tag
4.Create a class bar which extends applet class

5.Declare the array label[] and value[]
6.Define init () method to instantiate the array using label and value and assign name and give values
7.Call paint() method to draw a bar chart.
8.Stop the Program.

bar.html

```
<html>
<applet code="bar.class" width=500 height=500>
</applet>
</html>
```

Program

```
import java.applet.*;
import java.awt.*;

public class bar extends Applet
{

int n=0;
String label[ ]=new String[5];
int value[ ]=new int[5];


   public void init()
   {
try
{

label[0]="tamil";
label[1]="english";
label[2]="maths";
label[3]="science";
label[4]="social";
value[0]=75;
value[1]=57;
value[2]=92;
value[3]=66;
value[4]=80;
}
catch(Exception e)
{}

 }
   public void paint(Graphics g)
   {


      for(int i=0;i<5;i++)
      {

         g.setColor(Color.red);
         g.drawString(label[i],5,(i*50)+30);
         g.setColor(Color.blue);
```

```
        g.fillRect(60,i*50+10,value[i],30);

            }

}
}
```

Output

Result

Thus the program for draw bar chart using applet is compiled,executed and the required output is obtained.