EX.NO: 01
DATE:

MOVING A CAR USING APPLET

AIM:

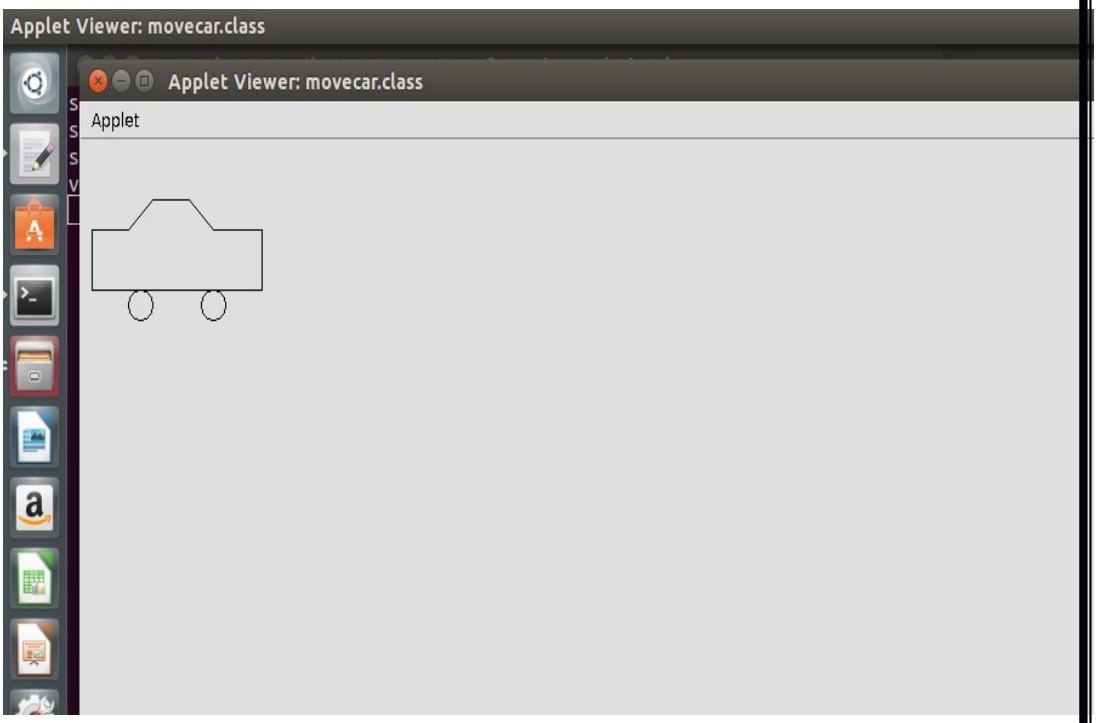
To write a program for moving a car in all the directions using applet.

PROGRAM:

```
/**<applet code=car.class width=1000 height=1000>
</applet>
*/
import java.applet.*;
import java.awt.*;
import java.awt.event.*;
import java.io.*;
import java.net.*;
public class movingcar extends Applet implements KeyListener
{
  int a1=40,a2=100,a3=100,a4=100;
  int x[]={10,10,40,60,90,110,150,150};
  int y[]={100,60,60,40,40,60,60,100};
  public void init()
  {
   addKeyListener(this);
  }
  public void paint(Graphics g)
  {
   g.drawPolygon(x,y,8);
   g.drawOval(a1,a2,20,20);
   g.drawOval(a3,a4,20,20);
  }
 public void keyPressed(KeyEvent ke)
 {
  int keycode=ke.getKeyCode();
  switch(keycode)
  {
    case KeyEvent.VK_UP:
     for(int i=0;i<8;i++)
     y[i]=y[i]-10;
     a2=a2-10;
     a4=a4-10;
    try
      {
       Thread.sleep(100);
      }catch(Exception e){}
```

```
        repaint();
        break;
      case KeyEvent.VK_DOWN:
       for(int i=0;i<8;i++)
       y[i]=y[i]+10;
       a2=a2+10;
       a4=a4+10;
      try
        {
          Thread.sleep(100);
        }catch(Exception e){}
        repaint();
        break;
     case KeyEvent.VK_LEFT:
       for(int i=0;i<8;i++)
       x[i]=x[i]-10;
       a1=a1-10;
       a3=a3-10;
      try
        {
          Thread.sleep(100);
        }catch(Exception e){}
        repaint();
        break;
   case KeyEvent.VK_RIGHT:
       for(int i=0;i<8;i++)
       x[i]=x[i]+10;
       a1=a1+10;
       a3=a3+10;
      try
        {
          Thread.sleep(100);
        }catch(Exception e){}
        repaint();
        break;



    }
  }

      public void keyReleased(KeyEvent ke)
          {
          }
          public void keyTyped(KeyEvent ke)
          {
          }
}
```

OUTPUT:



RESULT:

Thus the program has been successfully executed and the output is verified.

EX.NO: 02
DATE:

CAR WITH NAME USING APPLET

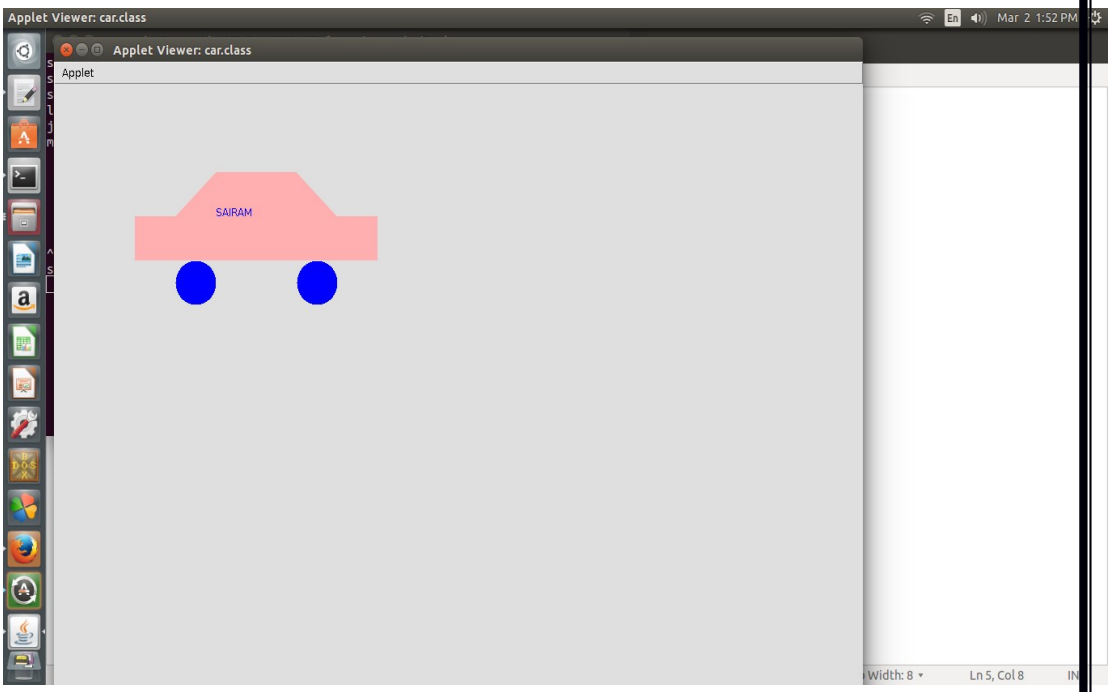
AIM:

To write a program to have the name in car using applet.

PROGRAM:

```
import java.applet.*;
import java.awt.*;
/**<applet code="car.class"width=1000 height=1000>
</applet>*/
public class car extends Applet
{
   int x[]={100,150,200,300,350,400,400,100};
   int y[]={150,150,100,100,150,150,200,200};
   public void paint(Graphics g)
  {
    g.setColor(Color.pink);
    g.fillPolygon(x,y,8);
    g.setColor(Color.blue);
    g.fillOval(150,200,50,50);
    g.fillOval(300,200,50,50);
    g.drawString("SAIRAM",200,150);
  }
}
```

OUTPUT:



RESULT:

Thus the program has been successfully executed and the output is verified.

EX.NO: 03
DATE:

Emp Form

AIM:

To design a emp form

PROGRAM:

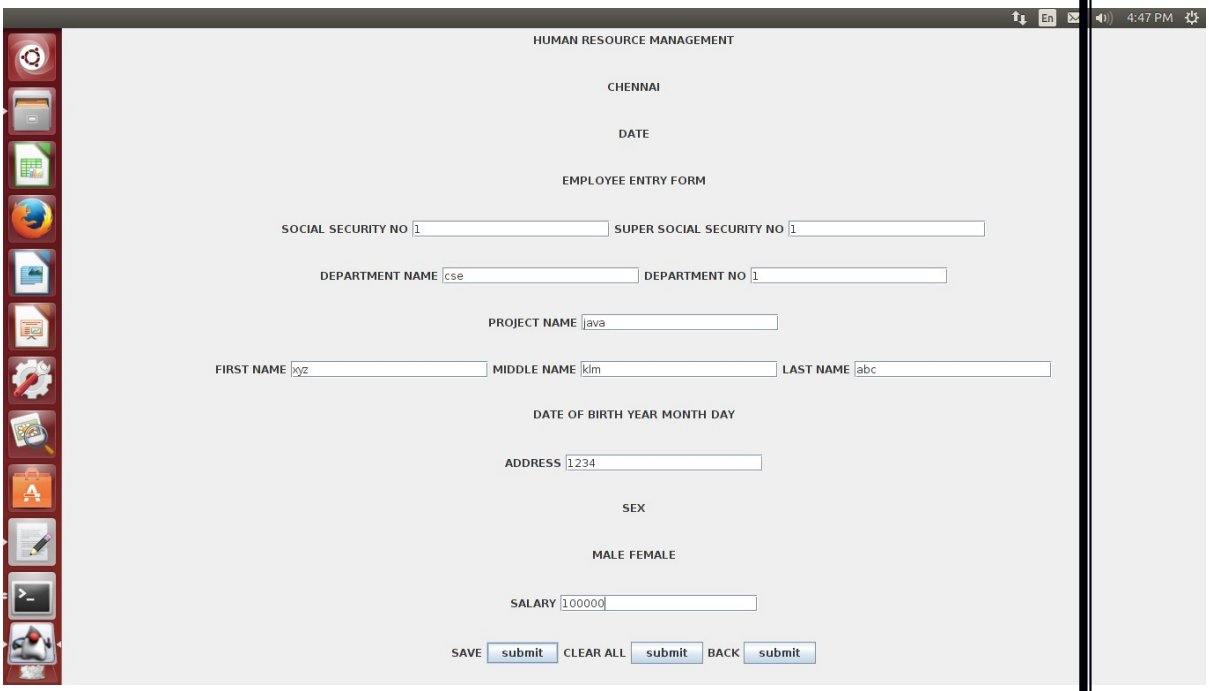
```
import java.awt.event.*;
import javax.swing.*;
import java.awt.*;
import java.sql.*;
class emp extends JFrame implements ActionListener
{
JTextField t1,t2,t3,t4,t5,t6,t7,t8,t9,t10;
emp()
{
Container c;
c=this.getContentPane();
c.setLayout(new GridLayout(14,2));
JPanel p1=new JPanel();
JPanel p2=new JPanel();
JPanel p3=new JPanel();
JPanel p4=new JPanel();
JPanel p5=new JPanel();
JPanel p6=new JPanel();
JPanel p7=new JPanel();
JPanel p8=new JPanel();
JPanel p9=new JPanel();
JPanel p10=new JPanel();JPanel p11=new JPanel();
JPanel p12=new JPanel();
JPanel p13=new JPanel();
JPanel p14=new JPanel();
JLabel l1=new JLabel("HUMAN RESOURCE MANAGEMENT");
JLabel l2=new JLabel("CHENNAI");
JLabel l3=new JLabel("DATE");
JLabel l4=new JLabel("EMPLOYEE ENTRY FORM");
JLabel l5=new JLabel("SOCIAL SECURITY NO");
JLabel l6=new JLabel("SUPER SOCIAL SECURITY NO");
JLabel l7=new JLabel("DEPARTMENT NAME");
JLabel l8=new JLabel("DEPARTMENT NO");
JLabel l9=new JLabel("PROJECT NAME");
JLabel l10=new JLabel("FIRST NAME");
JLabel l11=new JLabel("MIDDLE NAME");
```

```
JLabel l12=new JLabel("LAST NAME");
JLabel l13=new JLabel("DATE OF BIRTH");
JLabel l14=new JLabel("YEAR");
JLabel l15=new JLabel("MONTH");
JLabel l16=new JLabel("DAY");
JLabel l17=new JLabel("ADDRESS");
JLabel l18=new JLabel("SEX");
JLabel l19=new JLabel("MALE");
JLabel l20=new JLabel("FEMALE");
JLabel l21=new JLabel("SALARY");
JLabel l22=new JLabel("SAVE");
JLabel l23=new JLabel("CLEAR ALL");
JLabel l24=new JLabel("BACK");
JTextField t1=new JTextField(20);
JTextField t2=new JTextField(20);
JTextField t3=new JTextField(20);
JTextField t4=new JTextField(20);
JTextField t5=new JTextField(20);
JTextField t6=new JTextField(20);
JTextField t7=new JTextField(20);
JTextField t8=new JTextField(20);
JTextField t9=new JTextField(20);
JTextField t10=new JTextField(20);
JButton b1=new JButton("submit");
b1.addActionListener(this);
JButton b2=new JButton("submit");
b2.addActionListener(this);
JButton b3=new JButton("submit");
b3.addActionListener(this);
p1.add(l1);
p2.add(l2);
p3.add(l3);
p4.add(l4);
p5.add(l5);
p5.add(t1);
p5.add(l6);
p5.add(t2);
p6.add(l7);
p6.add(t9);
p6.add(l8);
p6.add(t10);
p7.add(l9);
p7.add(t3);
p8.add(l10);
p8.add(t4);
p8.add(l11);
p8.add(t5);
p8.add(l12);
```

```
p8.add(t6);
p9.add(l13);
p9.add(l14);
p9.add(l15);
p9.add(l16);
p10.add(l17);
p10.add(t7);
p11.add(l18);
p12.add(l19);
p12.add(l19);
p12.add(l20);
p13.add(l21);
p13.add(t8);
p14.add(l22);
p14.add(b1);
p14.add(l23);
p14.add(b2);
p14.add(l24);
p14.add(b3);
c.add(p1);
c.add(p2);
c.add(p3);
c.add(p4);
c.add(p5);
c.add(p6);
c.add(p7);
c.add(p8);
c.add(p9);
c.add(p10);
c.add(p11);
c.add(p12);
c.add(p13);
c.add(p14);
}
public void actionPerformed(ActionEvent ae)
{
    try
        {
            Class.forName("com.mysql.jdbc.Driver");
            String url="jdbc:mysql://172.18.61.33:3306/registration";
            Connection
con=DriverManager.getConnection(url,"registration","sse123");
            String query="insert into subjects(social security No,super social security
No,project name,first name,middle name,last
name,address,salary)values(?,?,?,?,?,?,?,?,?,?)";
            PreparedStatement st=con.prepareStatement(query);
            st.setString(1,t1.getText());
            st.setString(2,t2.getText());
```

```
            st.setString(3,t3.getText());
            st.setString(4,t4.getText());
            st.setString(5,t5.getText());
            st.setString(6,t6.getText());
            st.setString(7,t7.getText());
            st.setString(8,t8.getText());
             st.setString(9,t8.getText());
              st.setString(10,t8.getText());
            st.executeUpdate();
            System.out.println("success");
            st.close();
            con.close();
        }catch(Exception e){System.out.println(e);}

}
public static void main(String arg[])
{
emp e1=new emp();
e1.setSize(1200,1200);
e1.show();
}
}
```

OUTPUT:



RESULT:

Thus the program has been successfully executed and the output is verified

EX.NO: 04
DATE:

Add image

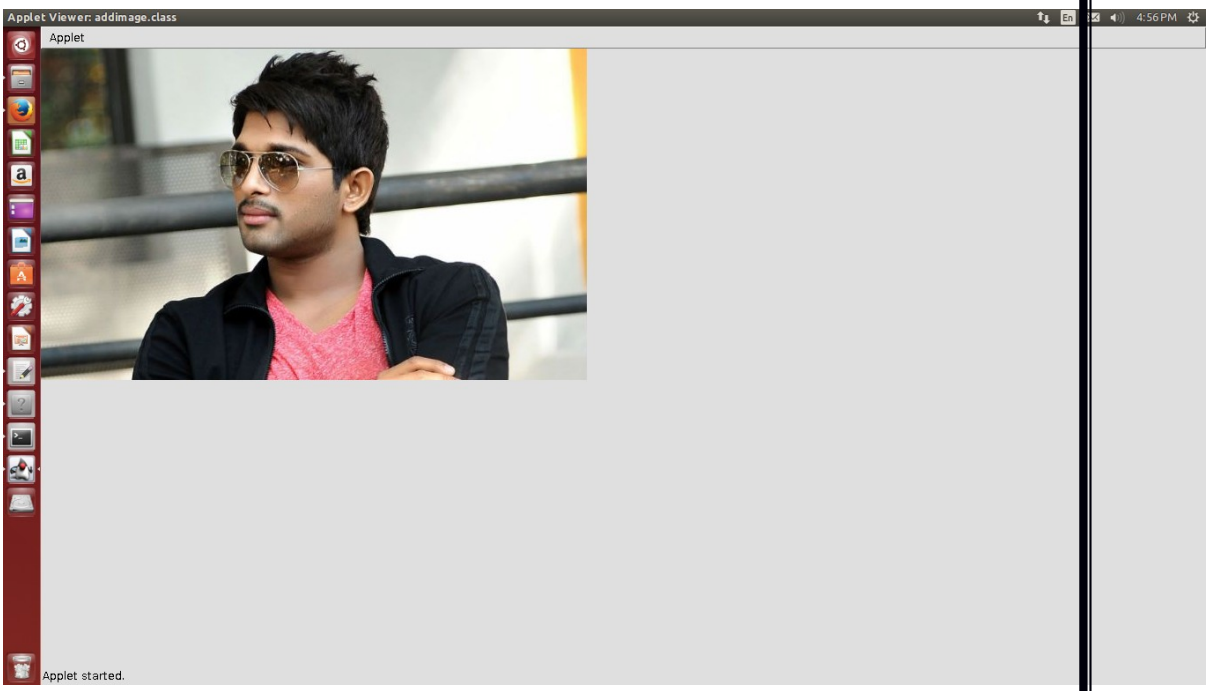
AIM:

To write a program to transfer the image by implementing client chat using TCP.

PROGRAM:

```
/**<applet code=addimage.class width=1000 height=1000>
</applet>
*/
import java.applet.*;
import java.applet.Applet;
import java.awt.Graphics2D;
import java.awt.*;
import java.net.URL;
public class addimage extends Applet
{
    Image img;
    URL base;
    public void init()
    {
           try
           {
             base=getDocumentBase();
           }catch(Exception e){System.out.println(e);}
           img=getImage(base,"1.jpg");

    }
    public void paint(Graphics g)
    {
      g.drawString("alluarjun",100,200);
      g.drawImage(img,0,0,this);
      Graphics2D gd=(Graphics2D)g;
    }
}
```

OUTPUT:



RESULT:

Thus the program has been successfully executed and the output is verify

EX.NO : 05
DATE :

Group Chat

EX.NO :05(A)

Multicast Send

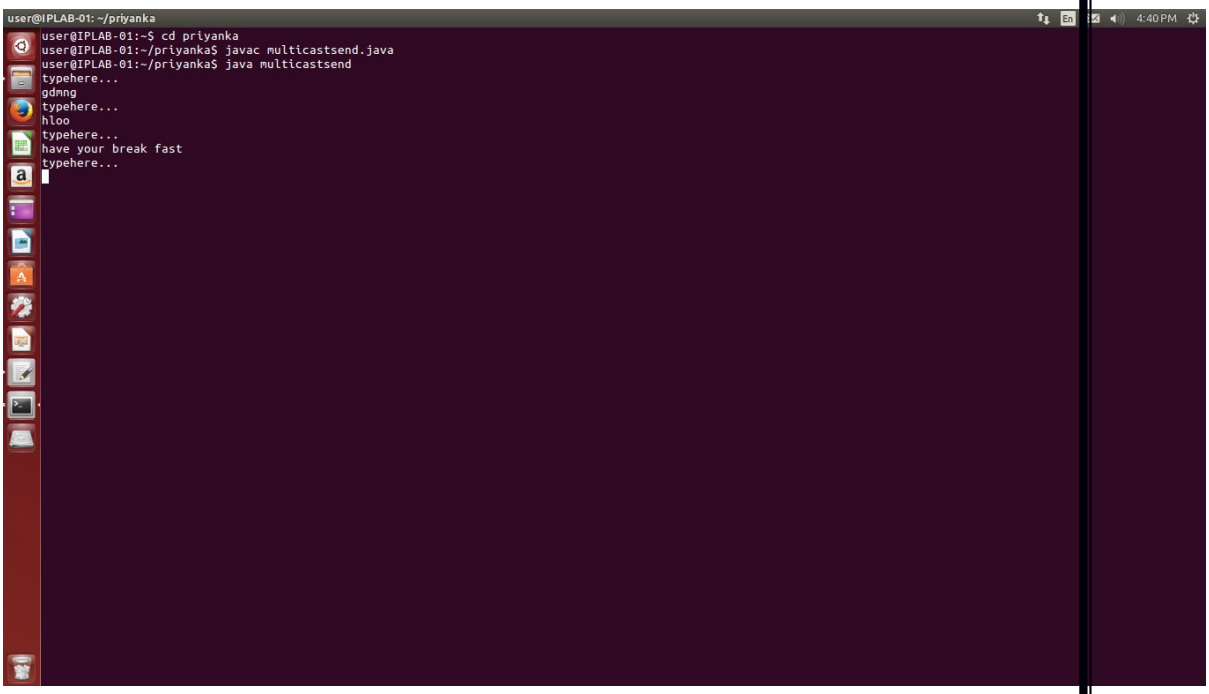
AIM:

To write a program to implement the multicast send

PROGRAM:

```
import java.net.*;
import java.io.*;
class multicastsend
{
  public static void main(String arg[])throws Exception
  {
    DatagramSocket Soc=new DatagramSocket();
    InetAddress Group=InetAddress.getByName("230.1.1.1");
    BufferedReader bufread=new BufferedReader(new
InputStreamReader(System.in));
    while (true)
    {
      System.out.println("typehere...");
      String Multicastmessage="priyanka:"+bufread.readLine();
      DatagramPacket Packet=new
DatagramPacket(Multicastmessage.getBytes(),Multicastmessage.length(),Group,1
2345);
      Soc.send(Packet);
    }
  }
}
```

OUTPUT:



RESULT:

Thus the program has been successfully executed and the output is verified

EX.NO :05(B)

Multicast Receive

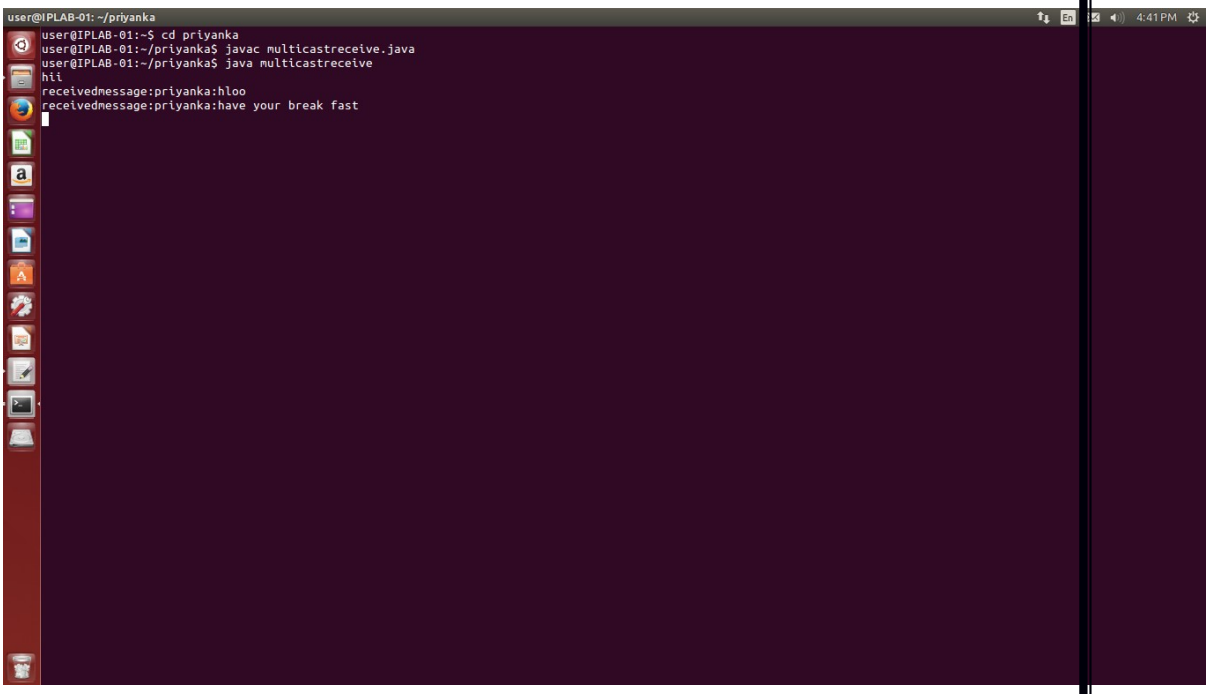
AIM:

To write a program to implement the multicast receive

PROGRAM:

```
import java.io.*;
import java.net.*;
class multicastreceive
{
   public static void main(String arg[])throws Exception
   {
    InetAddress maddress=InetAddress.getByName("230.1.1.1");
    MulticastSocket mSocket=new MulticastSocket(12345);
    mSocket.joinGroup(maddress);
    DatagramPacket packet=new DatagramPacket(new byte[1024],1024);
    while(true)
    {
      mSocket.receive(packet);
      String msg=new
String(packet.getData(),packet.getOffset(),packet.getLength());
      System.out.println("receivedmessage:"+msg);
    }
  }
}
```

OUTPUT:



RESULT : Thus the program has been successfully executed and the output is verified

EX.NO:06
DATE:

MACHINE TO MACHINE CHAT

EX.NO: 06(a)

MACHINE TO MACHINE CLIENT CHAT

AIM:

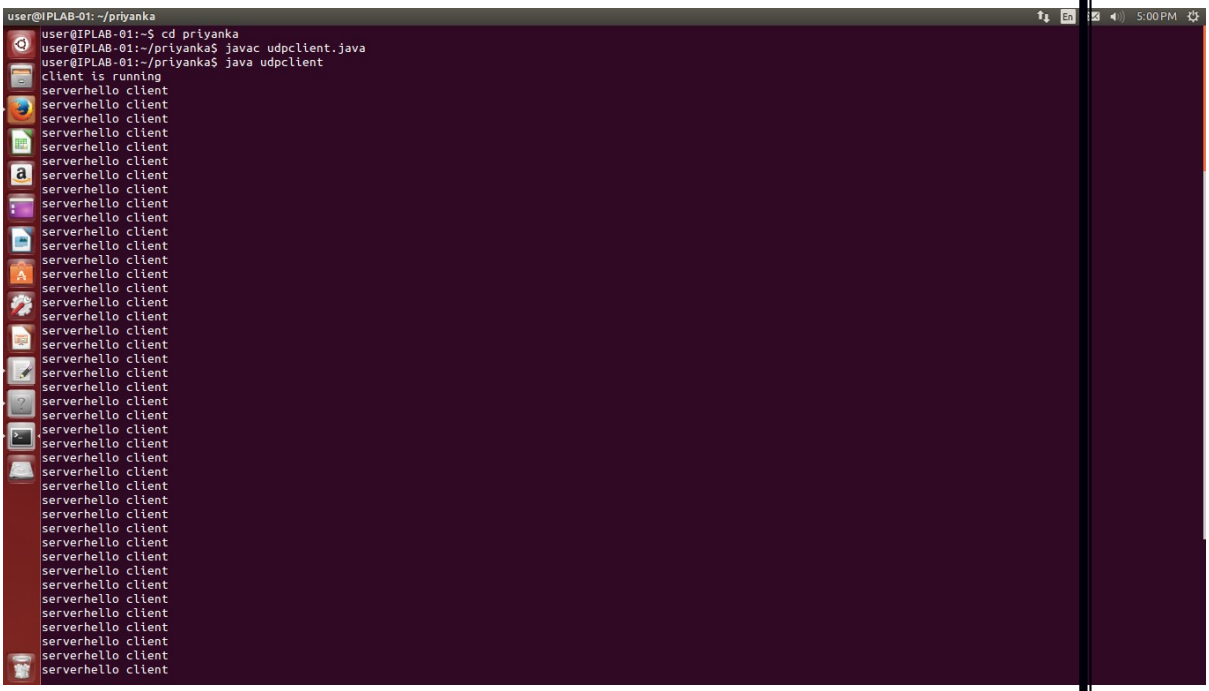
To write a program to implement the machine to machine using client chat.

PROGRAM:

```
import java.io.*;
import java.net.*;
class machineclient
{

        public static void main(String arg[])throws IOException
        {
           String str,str1="";
        int i;
           String words[]={"hi","what doing?","had break fast","studied?","what
home work on tomorrow?"};
        for(i=0;i<5;i++)
            System.out.println(words[i]);
        DatagramSocket serverSocket;
        DatagramPacket dp;
        BufferedReader dis;
        InetAddress ia;
        byte buff[]=new byte[1024];
        int cport=1789,sport=1790;
        serverSocket=new DatagramSocket(cport);
        dp=new DatagramPacket(buff,buff.length);
        dis=new BufferedReader(new InputStreamReader(System.in));
        ia=InetAddress.getByName("localhost");
        System.out.println("Client is Running");
           i=0;
        while(true)
         {
                  i++;
             if(i==5)
                i=0;
                  str1=words[i];

                  try
                  {
                           Thread.sleep(1000);
```

```
                }catch(Exception e){System.out.print(e);}
          buff=str1.getBytes();
          serverSocket.send(new DatagramPacket(buff,str1.length(),ia,sport));
            serverSocket.receive(dp);
             str=new String(dp.getData(),0,dp.getLength());
             if(str.equals("stop"))
             {
                   System.out.print("Terminated");break;
              }
       System.out.print("Server:"+str);

       }
   }
}
```

OUTPUT:



RESULT:

Thus the program has been successfully executed and the output is verified

EX.NO: 06(b)

MACHINE TO MACHINE SERVER CHAT

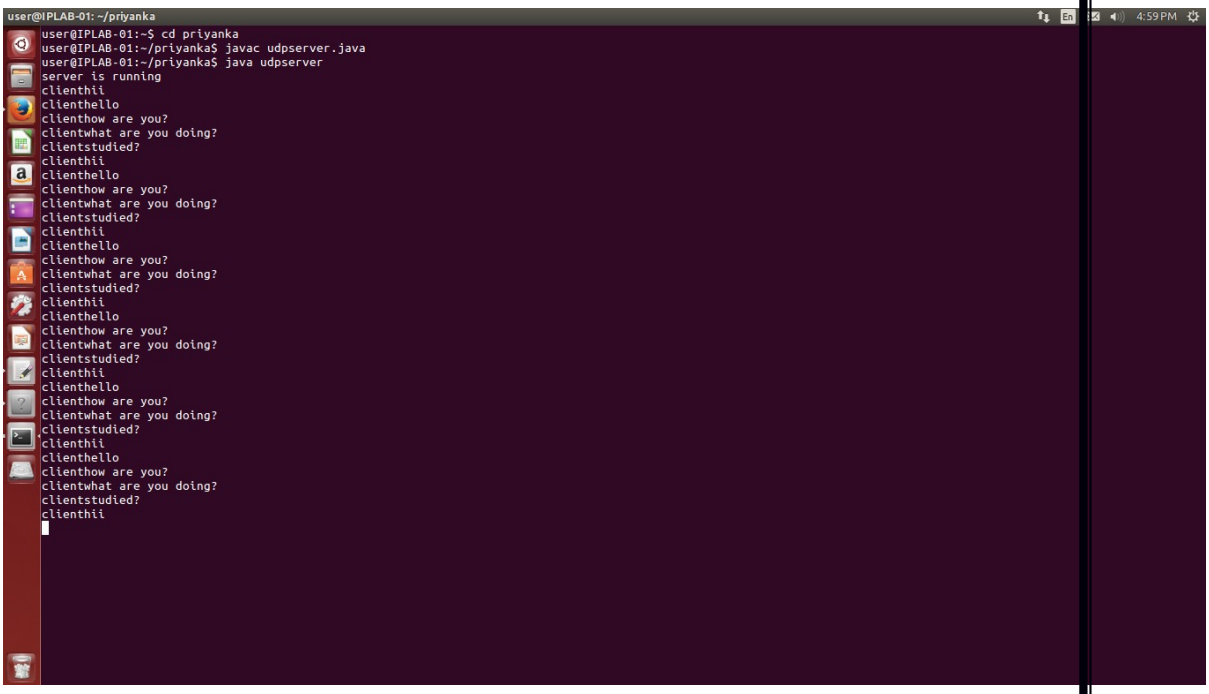
AIM:

To write a program to implement the machine to machine using server chat.

PROGRAM:

```
import java.io.*;
import java.net.*;
class machineserver
{

        public static void main(String arg[])throws IOException
        {
                String str,str1;
                DatagramSocket serverSocket;
                DatagramPacket dp;
                BufferedReader dis;
                InetAddress ia;
                byte buff[]=new byte[1024];
                int cport=1789,sport=1790;
                serverSocket=new DatagramSocket(sport);
                dp=new DatagramPacket(buff,buff.length);
                dis=new BufferedReader(new InputStreamReader(System.in));
                ia=InetAddress.getByName("localhost");
                System.out.println("Server is Running");
                while(true)
                {
                        serverSocket.receive(dp);
                        str=new String(dp.getData(),0,dp.getLength());
                        if(str.equals("stop"))
                        {
                                System.out.print("Terminated");break;
                        }
                        System.out.print("client:"+str);
                        //str1=new String(dis.readLine());
                        str1="hello client";
                        buff=str1.getBytes();
                        serverSocket.send(new
DatagramPacket(buff,str1.length(),ia,cport));
                }
        }
}
```

OUTPUT:



RESULT:

Thus the program has been successfully executed and the output is verified

EX.NO:07
DATE:

REVERSE THE INTEGER

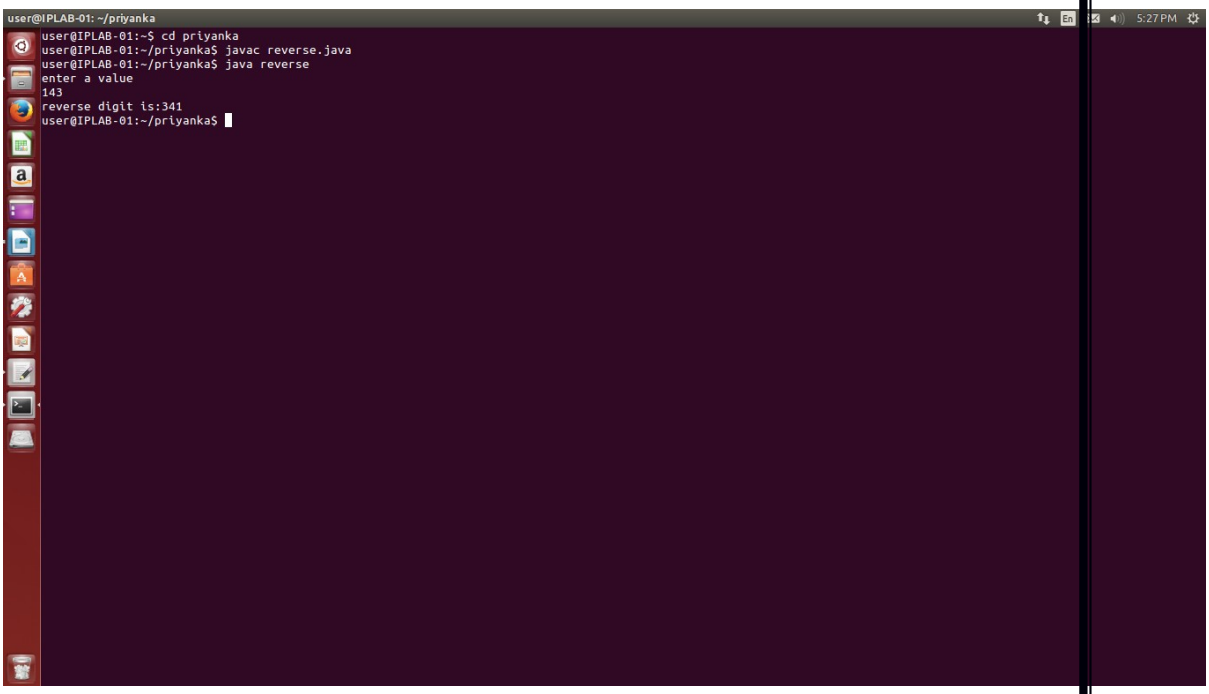
AIM:

To write a program to reverse the integer values.

PROGRAM:

```
import java.io.*;
import java.util.*;
class reverse
{
    public static void main(String arg[])
    {
        int rem,a,rev=0;
        System.out.println("enter a value");
        Scanner s=new Scanner(System.in);
        a=s.nextInt();
        while(a!=0)
        {
                rem=a%10;
                rev=(rev*10)+rem;
                a=a/10;
        }
         System.out.println("reverse digit is:"+rev);
     }

}
```

OUTPUT:



RESULT:

Thus the program has been successfully executed and the output is verified.

EX.NO:08
DATE:

Word is present or not

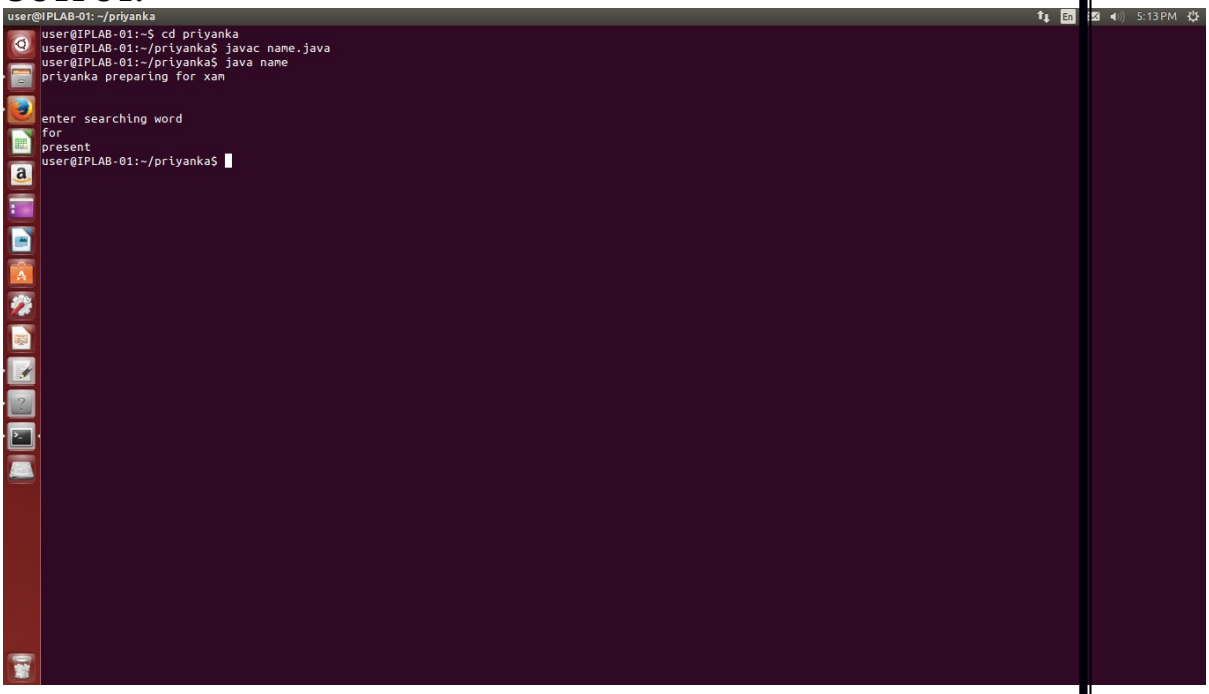
AIM:

To write a program to find the word is present or not..

PROGRAM:

```
import java.io.*;
import java.util.*;
class name
{
   public static void main(String arg[])throws Exception
   {
    int flag=0;
    byte[] b=new byte[1024];
    String str=" ";
    BufferedInputStream buf=new BufferedInputStream(new
FileInputStream("file1.txt"));
    int i=0;
    while((i=buf.read(b))!=-1)
     {
        str=new String(b,0,i);
        System.out.println(str);
     }
    //String str="hello i am priya";
    System.out.println("enter searching word");
    Scanner s=new Scanner(System.in);
    String searchword=s.nextLine();
    String[] words=str.split(" ");
    for(i=0;i<words.length;i++)
    {
      if(words[i].equals(searchword))
      flag=1;
    }
     if(flag==1)
        System.out.println("present");
     else
        System.out.println("not present");

   }
}
```

OUTPUT:



RESULT:

Thus the program has been successfully executed and the output is verified.

EX.NO: 09
DATE:

PALINDROME

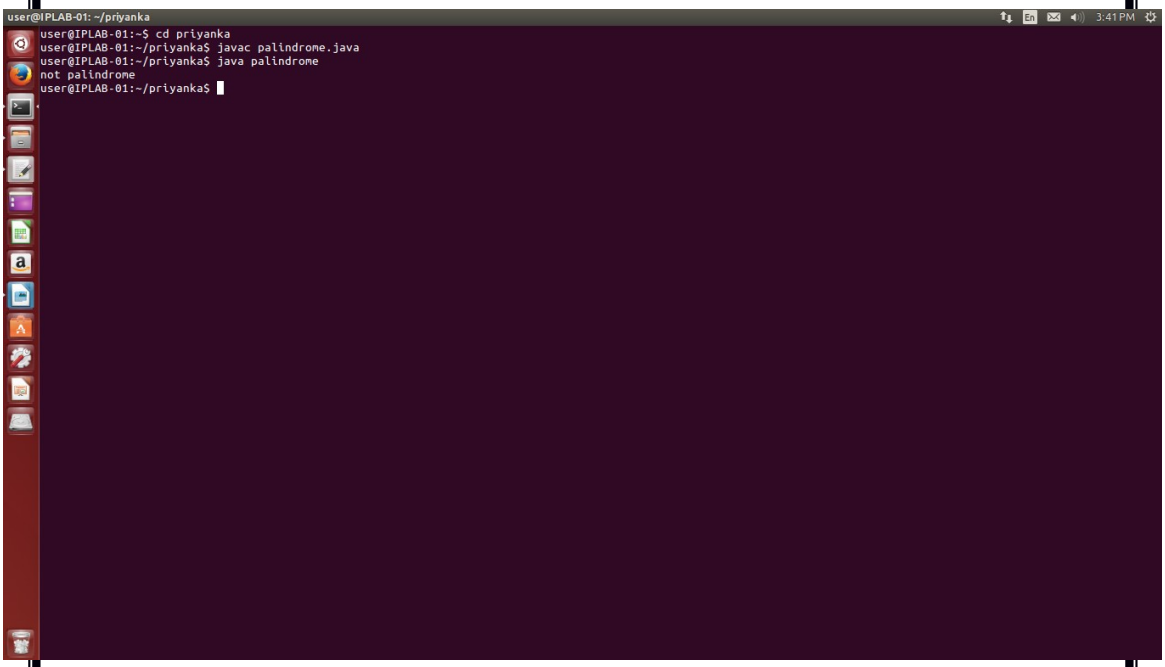
AIM:

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

PROGRAM:

```
import java.io.*;
class palindrome
{
public static void main(String args[])
{
        int r,sum=0,temp;
        int n=453;
        temp=n;
        while(n>0)
        {
                r=n%10;
                sum=(sum*10)+r;
                n=n/10;
        }
        if(temp==sum)
        System.out.println("palindrome number ");
        else
        System.out.println("not palindrome");
  }
}
```

OUTPUT:



RESULT:

Thus the program has been successfully executed and the output is verified.

EX.NO: 10
DATE:

ENCODE AND DECODE A STRING

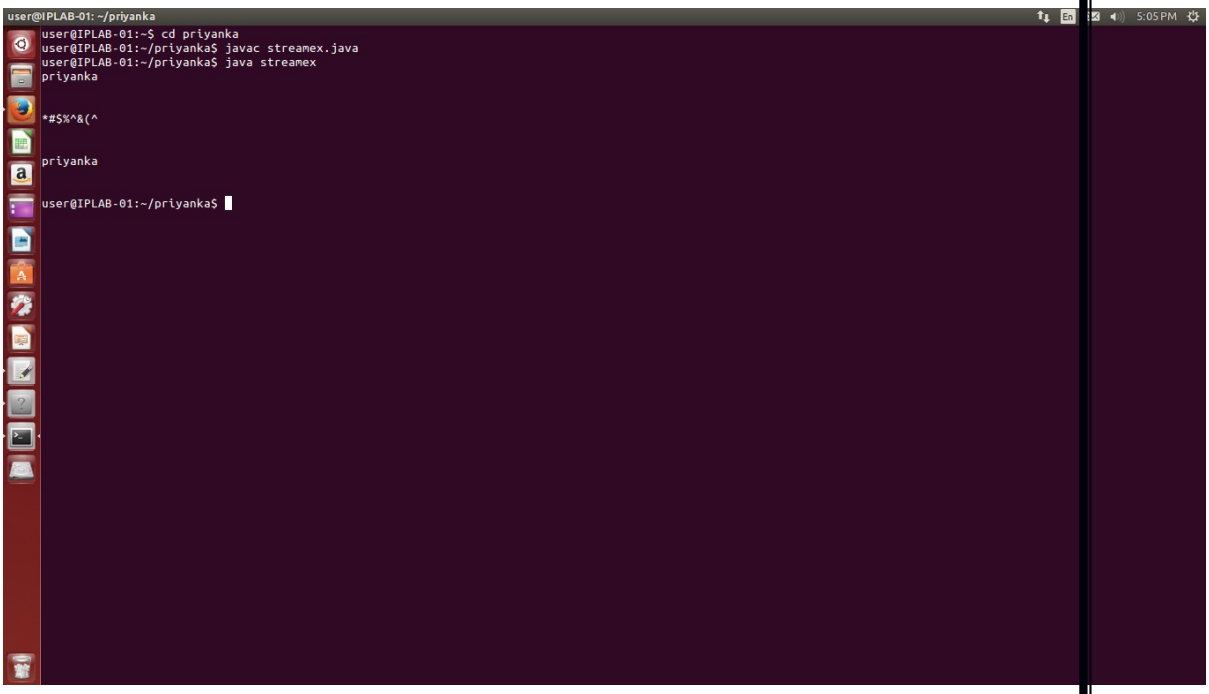
AIM:

To write a program to encode and decode the string.

PROGRAM:

```
import java.io.*;
class streamex
{
	public static void main(String arg[])throws Exception
	{
		byte[] b=new byte[1024];
		String Str=" ";
		BufferedInputStream buf=new BufferedInputStream(new
FileInputStream("file1.txt"));
		int i=0;
		while((i=buf.read(b))!=-1)
		{
			Str=new String(b,0,i);
		}
		System.out.println(Str);
		 buf.close();
		String encode=Str.replace('e','*');
		encode=encode.replace('m','#');
		encode=encode.replace('a','$');
		encode=encode.replace('n','%');
		encode=encode.replace('i','@');
		String decode=str.replace('*','e');
		decode=decode.replace('#','m');
		decode=decode.replace('$','a');
		decode=decode.replace('%','n');
		decode=decode.replace('@','i');
		System.out.println(encode);
		System.out.println(decode);
		BufferedOutputStream buf1=new BufferedOutputStream(new
FileOutputStream("file2.txt"));
		buf1.write(encode.getBytes());
		buf1.close();
	}
}
```

OUTPUT:



RESULT:

Thus the program has been successfully executed and the output is verified.

EX.NO: 11
DATE:

COPY A TEXT FROM ONE FILE TO ANOTHER FILE

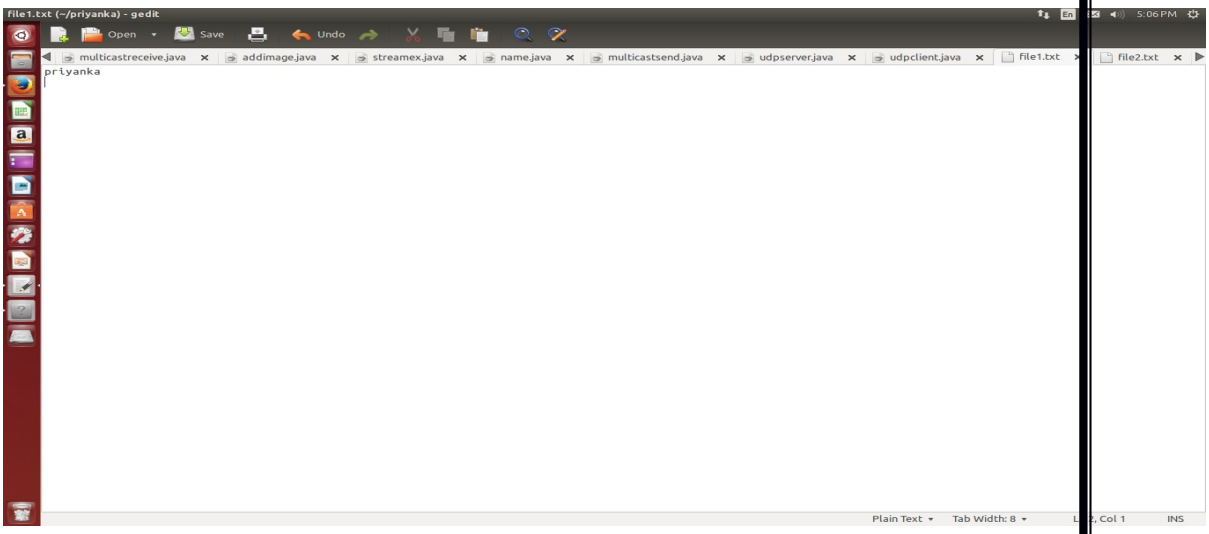
AIM:

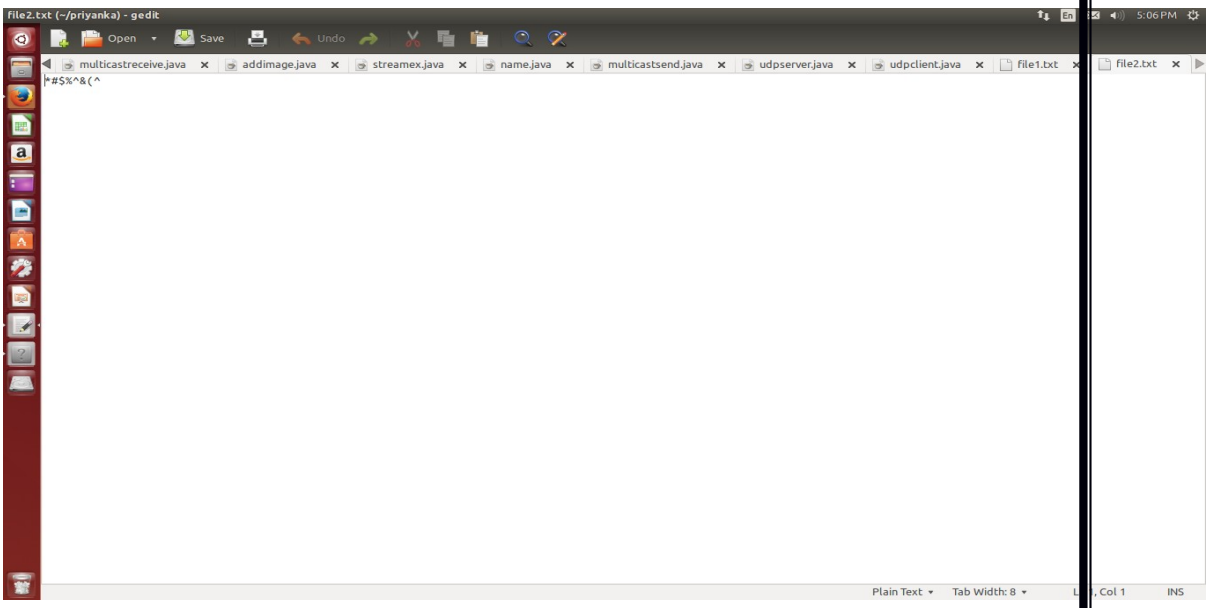
To write a program to copy the text from one file to another file.

PROGRAM:

```
import java.io.*;
public class stream
{
	public static void main(String args[])throws Exception
	{
		byte[] b=new byte[1024];
		String str="";
		BufferedInputStream buf=new BufferedInputStream(new
FileInputStream("file.txt"));
		int i=0;
		while((i=buf.read(b))!=-1)
		{
			str=new String(b,0,i);
		}
		System.out.println(str);
		buf.close();
		BufferedOutputStream buf1=new BufferedOutputStream(new
FileOutputStream("file1.txt"));
		buf1.write(str.getBytes());
		buf1.close();

	}
}
```

OUTPUT:





RESULT:

Thus the program has been successfully executed and the output is verified.

EX.NO: 12
DATE:

FILESTREAM

AIM:

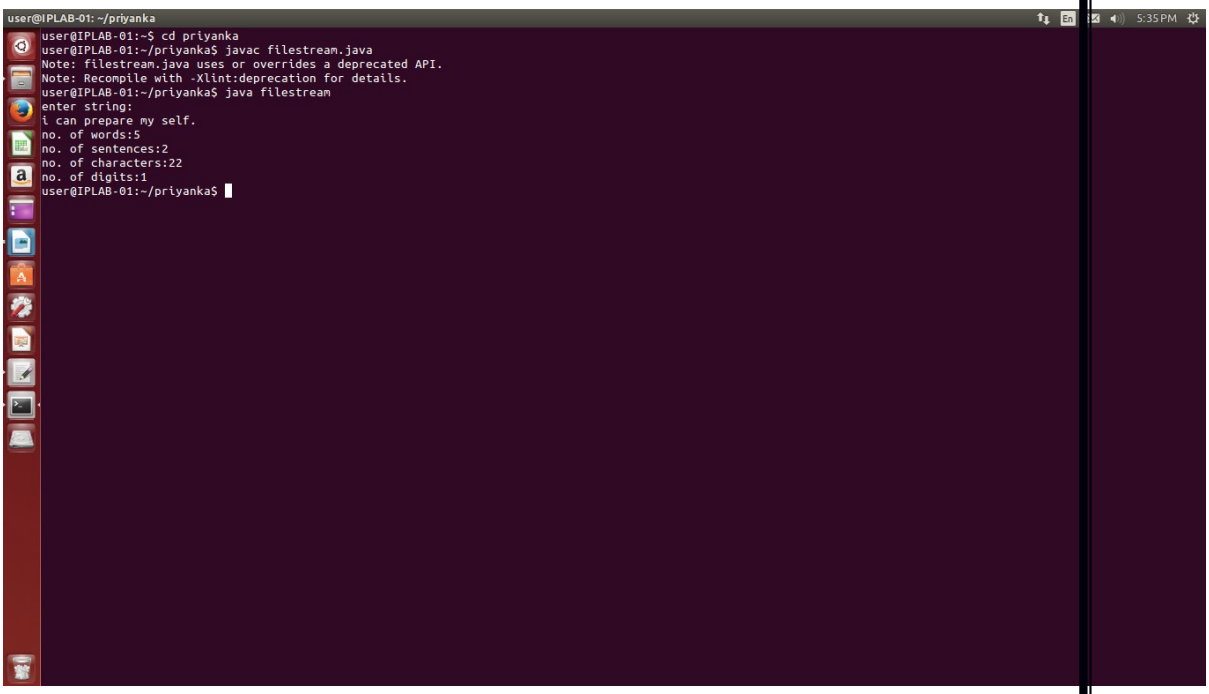
To write a program to display the words, sentences, digits.

PROGRAM:

```
import java.io.*;
class filestream
{
    public static void main(String arg[])throws IOException
    {
        String str;int wordcount=1,sentcount=1,charcount=1,digitcount=1;
        System.out.println("enter string:");
        try
        {
            DataInputStream d=new DataInputStream(System.in);
            str=d.readLine();
            OutputStream os=new FileOutputStream("Input.txt");
            os.write(str.getBytes());
            os.close();
            InputStream in=new FileInputStream("Input.txt");
            int size=in.available();

            for(int i=0;i<size;i++)
            {
                  char ch=(char)in.read();
                  if(ch==' ')
                  wordcount++;
                  if(ch=='.')
                  sentcount++;
                  if(Character.isDigit(ch))
                  digitcount++;

            }
            in.close();
             System.out.println("no. of words:"+wordcount);
             System.out.println("no. of sentences:"+sentcount);
             System.out.println("no. of characters:"+size);
             System.out.println("no. of digits:"+digitcount);
```

```
        }catch(Exception e){}
    }
}
```

OUTPUT:



RESULT:

Thus the program has been successfully executed and the output is verified

EX.NO: 13
DATE:

ODD AND EVEN THREADS

AIM:

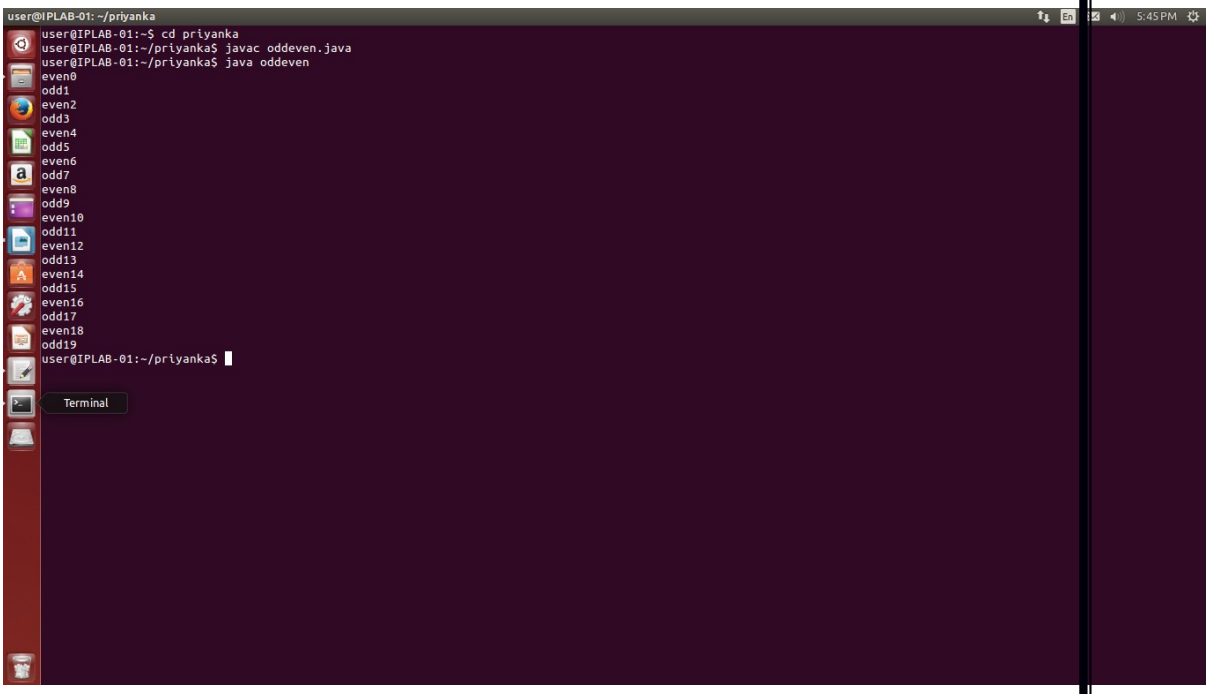
To write a program to display the odd and even threads.

PROGRAM:

```
import java.io.*;
class oddeven
{
      public static void main(String arg[])
      {
            eventhread et=new eventhread();
            et.start();
            oddthread ot=new oddthread();
            ot.start();
      }
}
class eventhread extends Thread
{
       int i;
       public void run()
       {
            for(i=0;i<20;i+=2)
            {
                  System.out.println("even"+i);
                   try
                   {
                         Thread.sleep(1000);
                   }catch(Exception e){}

            }
       }
}
class oddthread extends Thread
{
         int i;
         public void run()
         {
                  for(i=1;i<20;i+=2)
```

```
                {
                        System.out.println("odd"+i);
                        try
                         {
                                Thread.sleep(1000);
                           }catch(Exception e){}

                }
        }
}
```

OUTPUT:



RESULT:

Thus the program has been successfully executed and the output is verified

EX.NO: 14
DATE:

POSITIVE AND NEGATIVE THREAD

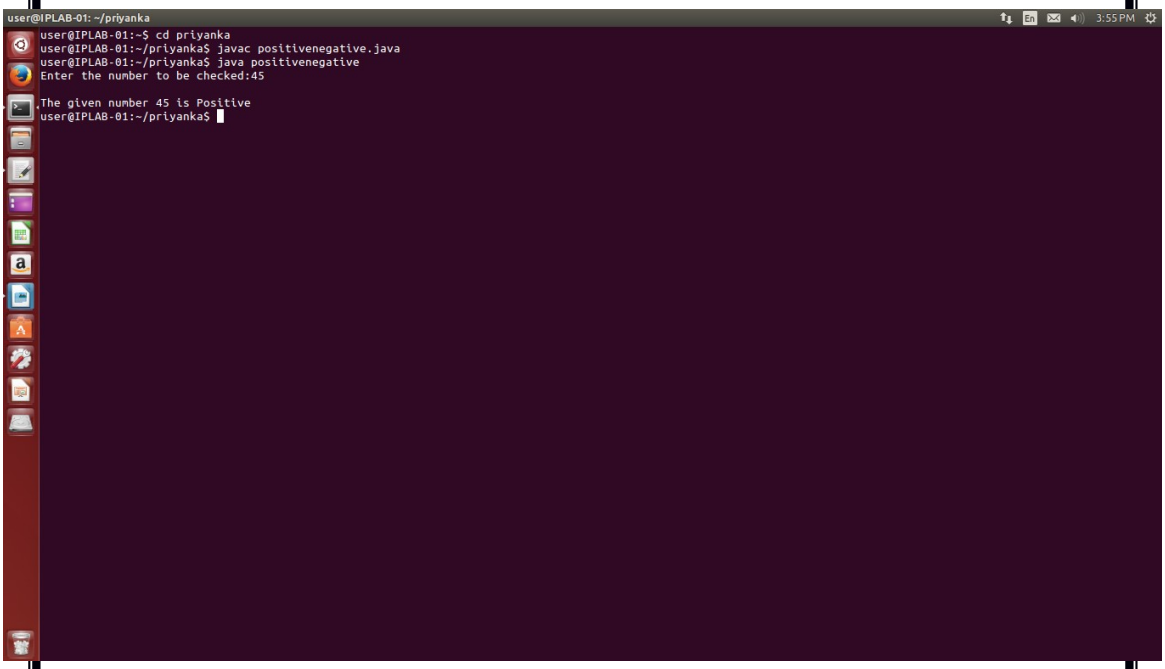
AIM:

To write a program to display the positive and negative threads.

PROGRAM:

```
import java.util.*;
public class postivenegative
{
	public static void main(String arg[])
	{
		int n;

		Scanner s = new Scanner(System.in);
		System.out.print("Enter the number to be checked:");
		n = s.nextInt();
		if(n > 0)
		{
			System.out.println("\nThe given number "+n+" is Positive");
		}
		else if(n < 0)
		{
			System.out.println(\nThe given number "+n+" is Negative");
		}
		else
		{
			System.out.println("\nThe given number "+n+" is neither
Positive nor Negative ");
		}
	}
}
```

OUTPUT:



RESULT:

Thus the program has been successfully executed and the output is verified

EX.NO: 15
DATE

Design Student Form Using HTML

AIM:

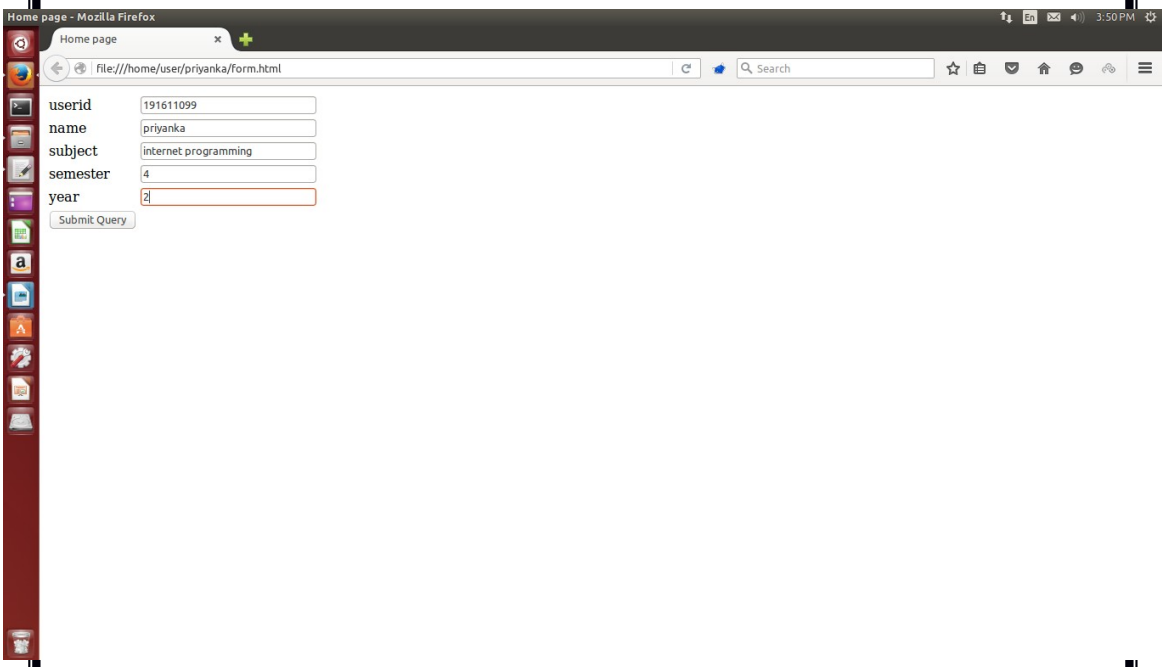
To design a html form

PROGRAM:

FORM.HTML:

```
<HTML>
<TITLE>Home page</TITLE>
<Body>
<form id="form1"  name="form1" method="post" action="submit.php">
<table>
<tr>
<td>userid</td>
<td><input type="text"   name="userid"></td>
</tr>
<tr>
<td>name</td>
<td><input type="text"  name="name"></td>
</tr>
<tr>
<td>subject</td>
<td><input type="text"  name="subject"></td>
</tr>
<tr>
<td>semester</td>
<td><input type="text"  name="semester"></td>
</tr>
<tr>
<td>year</td>
<td><input type="text"  name="year"></td>
</tr>
<tr>
<td><input type="submit" name="submit"values="submit"></td>
</tr>
</table>
</form>
</Body>
</HTML>
```

OUTPUT:



RESULT:

Thus the program has been successfully executed and the output is verified

EX.NO: 16
DATE:

Login Demo

AIM:

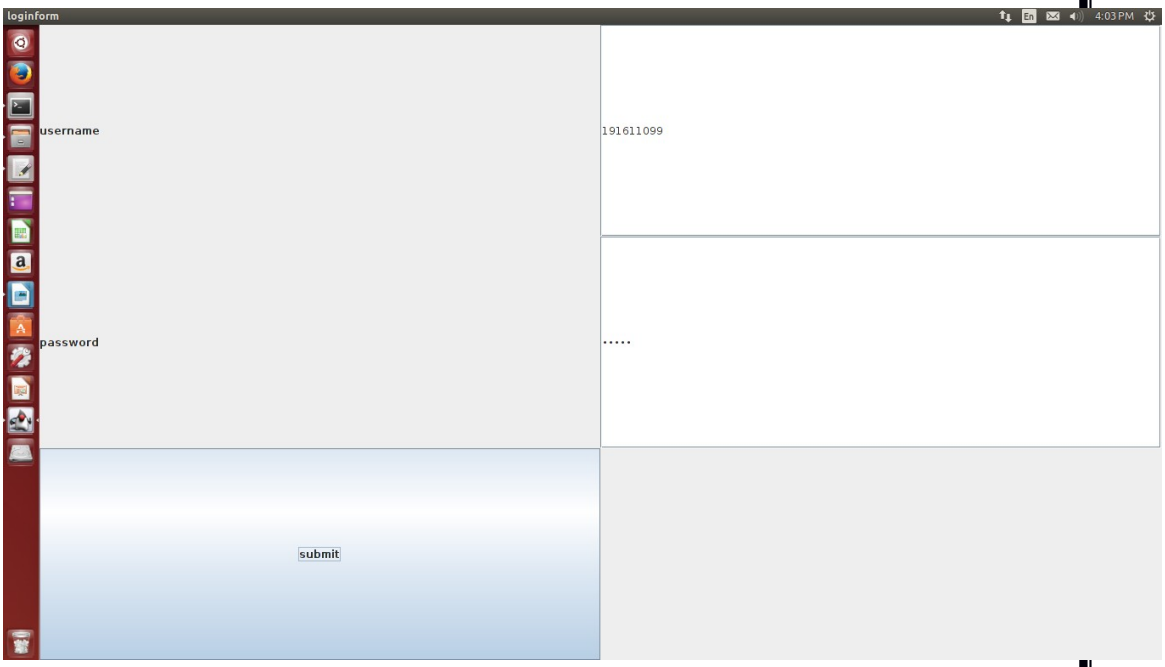
To write a program for login demo

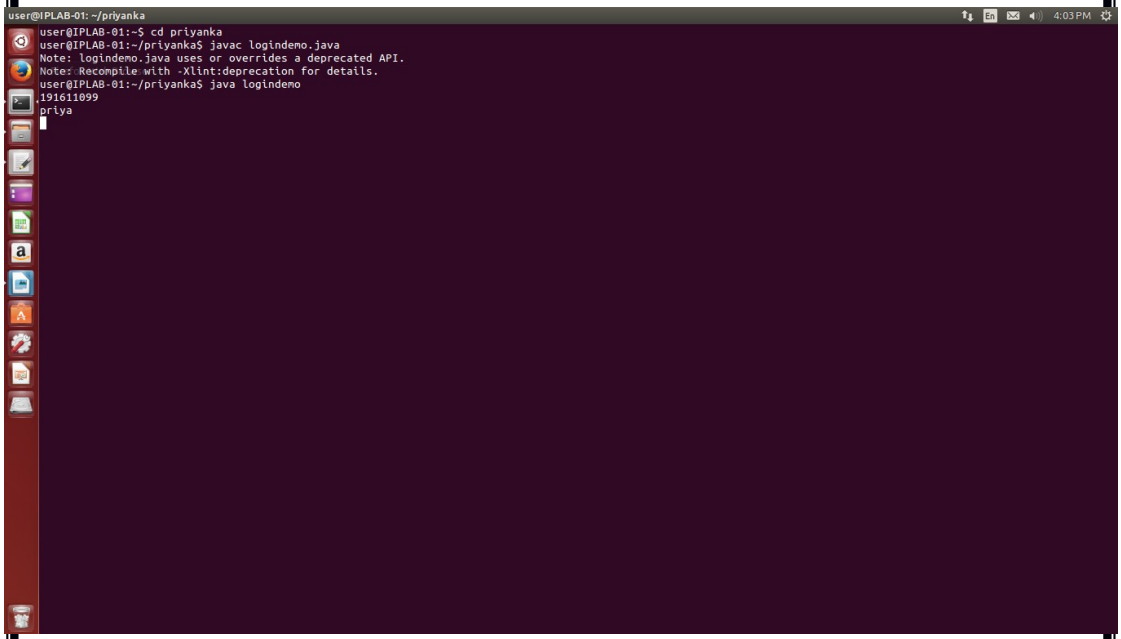
PROGRAM:

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
class login extends JFrame implements ActionListener
{
        JButton submit;
        JPanel panel;
        JLabel label1,label2;
        final JTextField text1,text2;
        login()
        {
            label1=new JLabel("username");
            text1=new JTextField(15);
            label2=new JLabel("password");
        text2=new JPasswordField(15);
            submit=new JButton("submit");
            panel=new JPanel(new GridLayout(3,2));
            panel.add(label1);
            panel.add(text1);
            panel.add(label2);
            panel.add(text2);
            panel.add(submit);
            add(panel,BorderLayout.CENTER);
            submit.addActionListener(this);
            setTitle("loginform");
        }
        public void actionPerformed(ActionEvent e)
        {
            String value1=text1.getText();
            String value2=text2.getText();
            System.out.println(value1);
            System.out.println(value2);
            if(value1.equals(value2))
        {
            emp e1=new emp();
```

```
                e1.setSize(1200,1200);
                 e1.show();
                }

        }
}
class logindemo
{
        public static void main(String arg[])
        {
                try
                {
                        login l=new login();
                        l.setSize(1000,1000);
                        l.setVisible(true);
                }catch(Exception e)
                {
                        JOptionPane.showMessageDialog(null,e.getMessage());
                }
        }
}
```

OUTPUT:





RESULT:

Thus the program has been successfully executed and the output is verified

EX.NO: 17
DATE:

DESIGN DATE FORMAT USING PHP PROGRAM

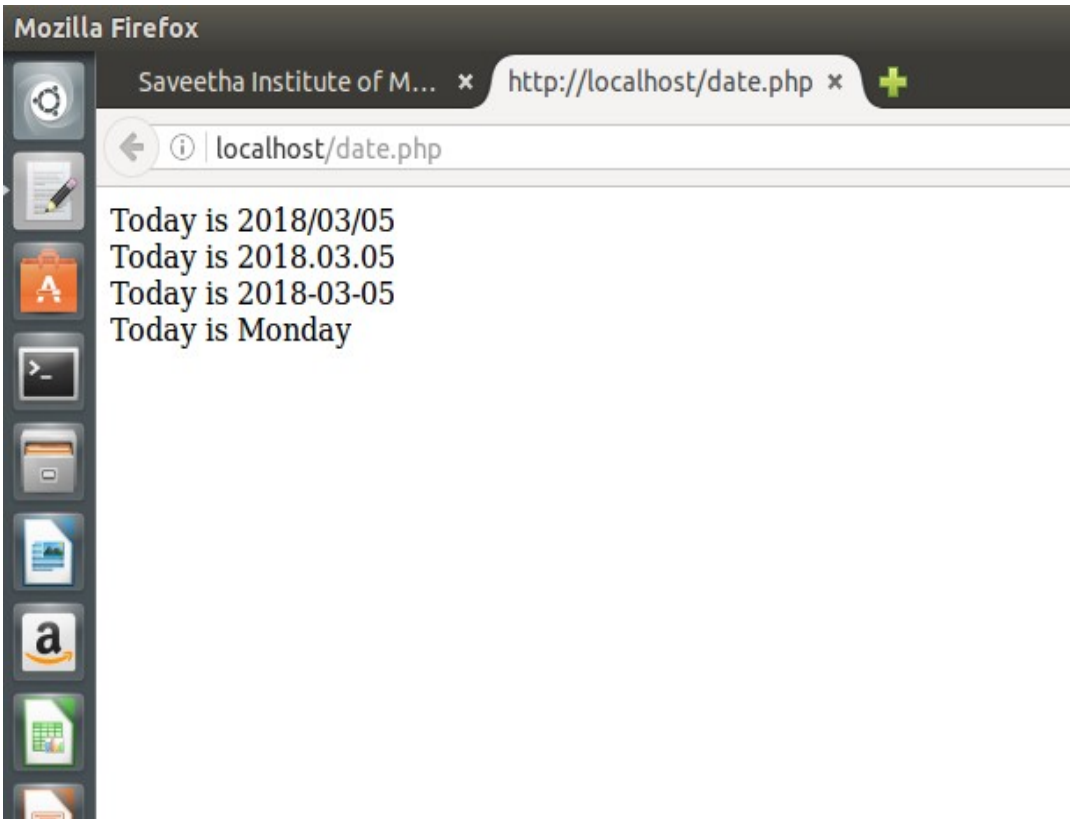
AIM:

To design the date format using php program.

PROGRAM:

```
<!DOCTYPE html>
<html>
<body>

<?php
echo "Today is " . date("Y/m/d") . "<br>";
echo "Today is " . date("Y.m.d") . "<br>";
echo "Today is " . date("Y-m-d") . "<br>";
echo "Today is " . date("l");
?>

</body>
</html>
```

OUTPUT:



RESULT:

Thus the program has been successfully executed and the output is verified

EX.NO: 18
DATE:

READ AND DISPLAY FROM FILE

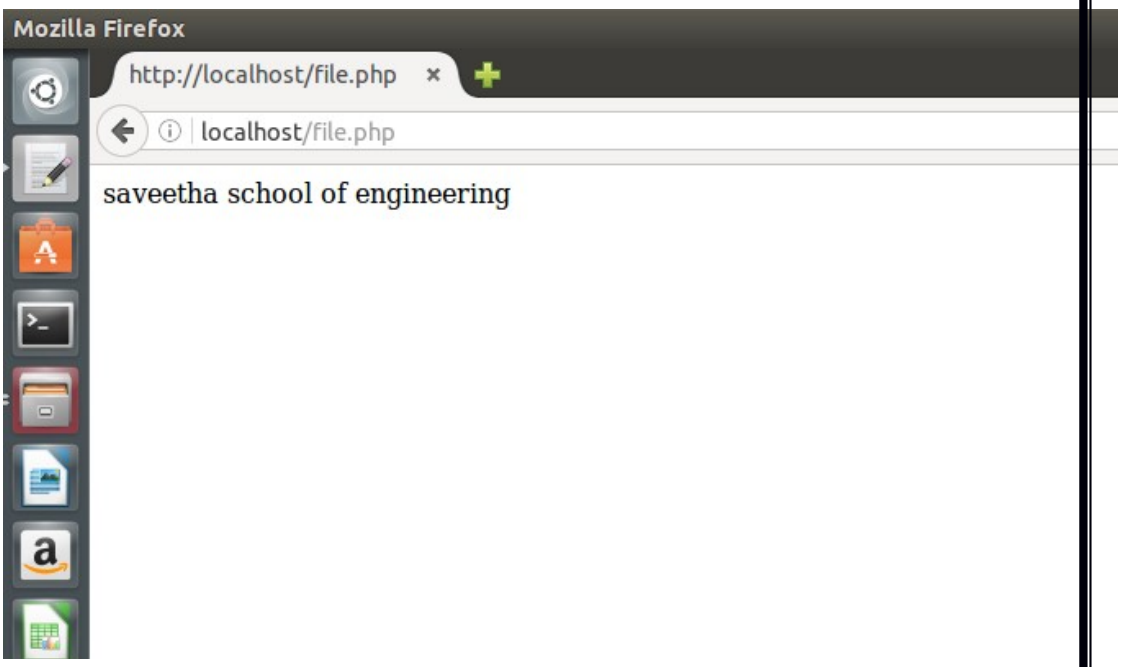
AIM:

To write a php program to read and display from file.

PROGRAM:

```
<?php
$myfile = fopen("webdictionary.txt", "r") or die("Unable to open file!");
echo fread($myfile,filesize("webdictionary.txt"));
fclose($myfile);
?>
```

OUTPUT:



RESULT:

Thus the program has been successfully executed and the output is verified

EX.NO: 19
DATE:

FUNCTIONS IN PHP

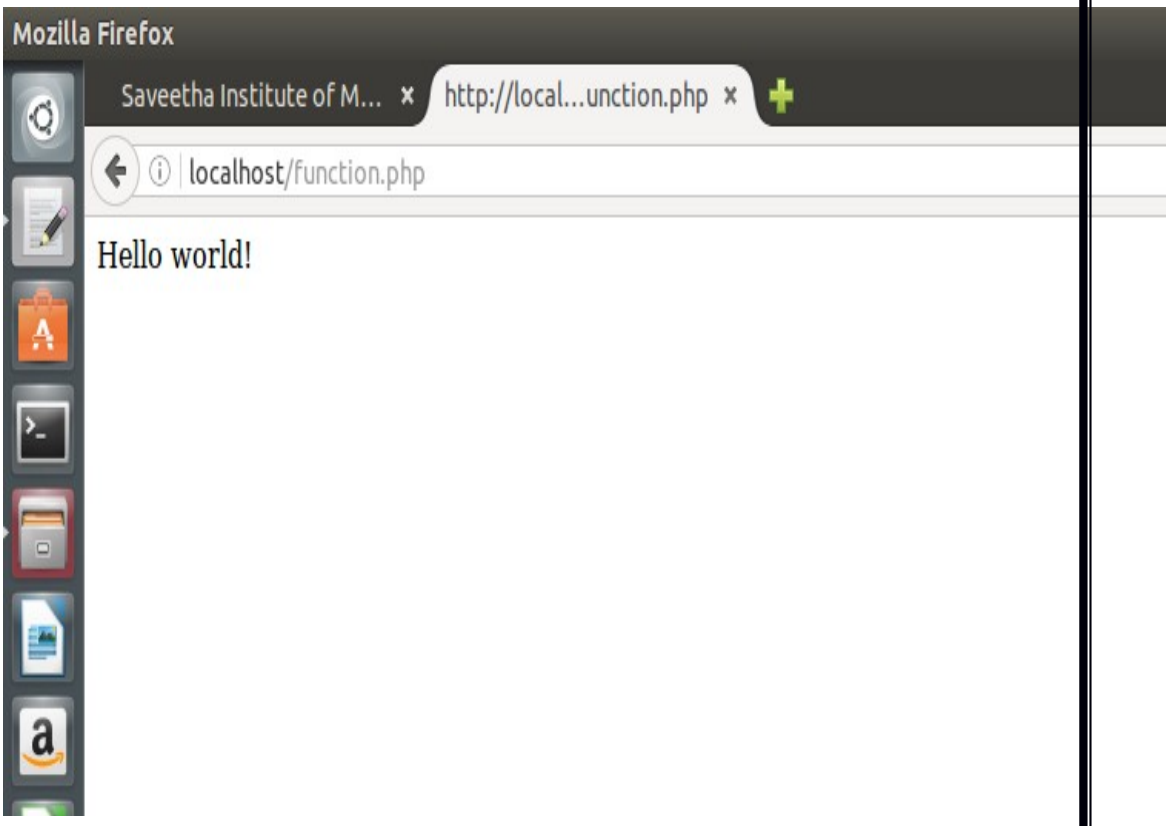
AIM:

To write a php program for functions.

PROGRAM:

```
<?php function writeMsg() { echo "Hello world!";} writeMsg();?>
```

OUTPUT:



RESULT:

Thus the program has been successfully executed and the output is verified

EX.NO: 20
DATE:

ARRAYS IN PHP

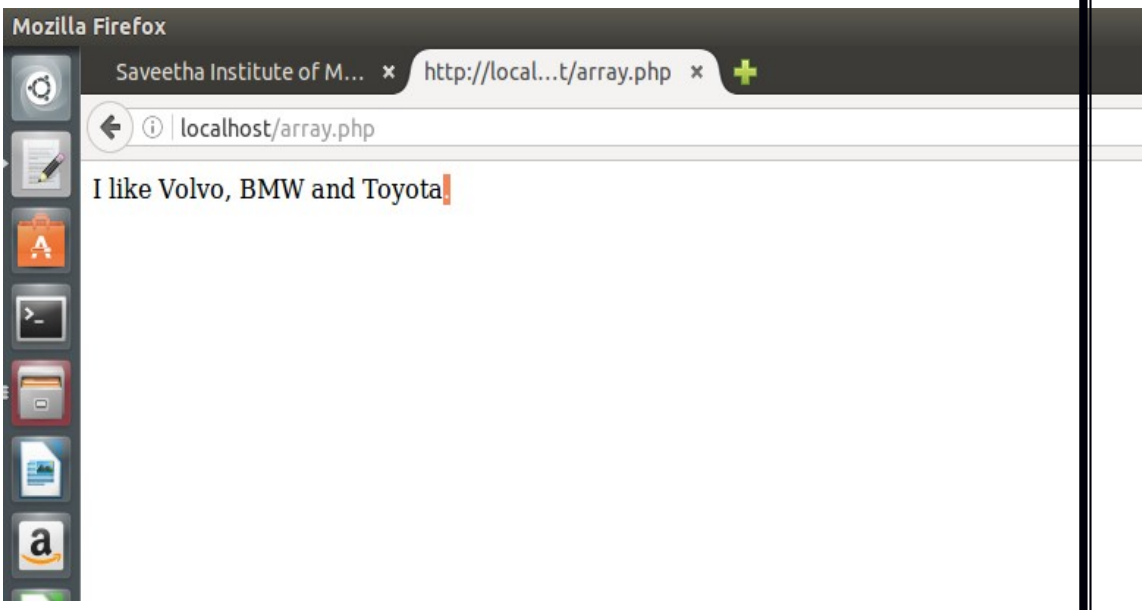
AIM:

To write a php program for arrays.

PROGRAM:

```
<?php $cars = array("Volvo", "BMW", "Toyota");  echo "I like " . $cars[0] . ", " . $cars[1] . " and " . $cars[2] . "."; ?>
```

OUTPUT:



RESULT:

Thus the program has been successfully executed and the output is verified

EX.NO: 21
DATE:

FACTORIAL IN PHP

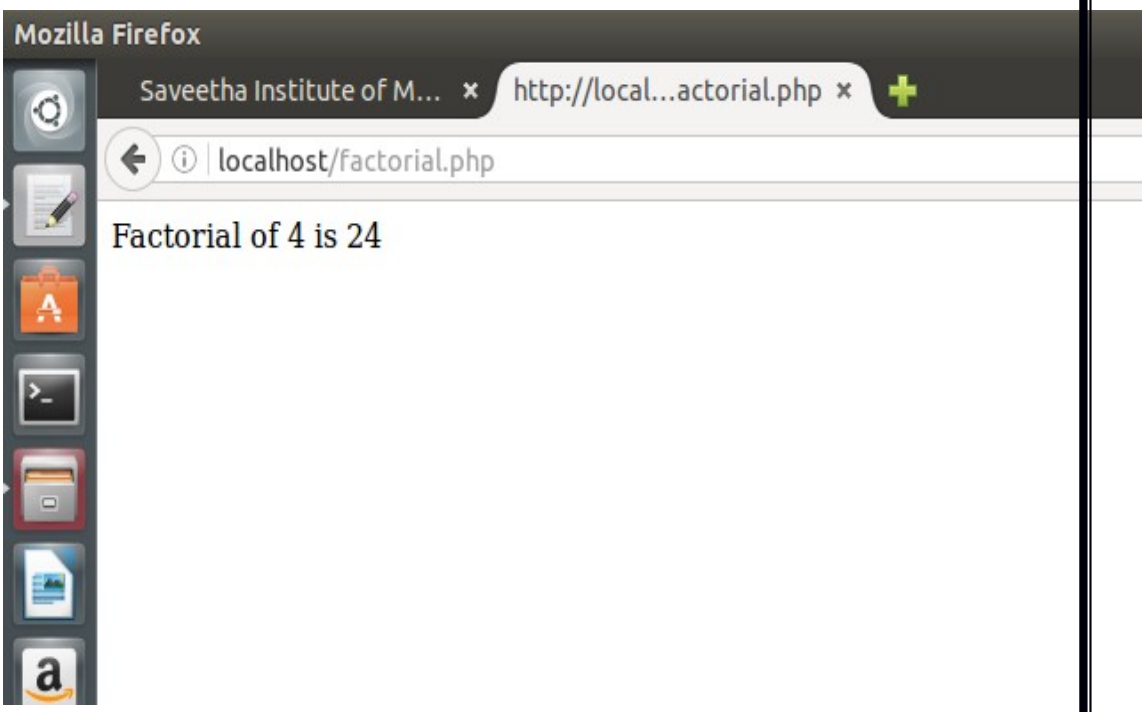
AIM:

To write the php program for factorial of a number.

PROGRAM:

```
<?php
 $num = 4;
 $factorial = 1;
 for ($x=$num; $x>=1; $x--)
{
 $factorial = $factorial * $x;
}
 echo "Factorial of $num is $factorial";
 ?>
```

OUTPUT:



RESULT:

Thus the program has been successfully executed and the output is verified