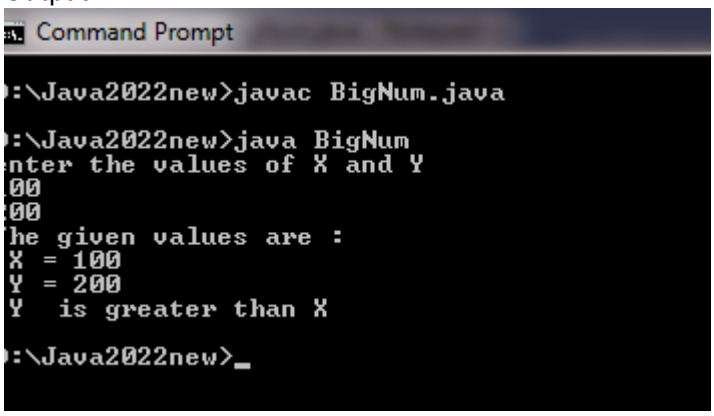
Java Lab Programs – Part A

1. Program to assign two integer values to X and Y. Using the 'if' statement the output of the program should display a message whether X is greater than Y

```
import java.io.*;

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

       BufferedReader br = new BufferedReader(new InputStreamReader
(System.in));

       System.out.println("enter the values of X and Y");
       int x=Integer.parseInt(br.readLine());
       int y=Integer.parseInt(br.readLine());

      System.out.println("The given values are :");
      System.out.println(" X = " + x);
      System.out.println(" Y = " + y);

     if(x==y)
           System.out.println("Both X and Y are Equal");
     else  if(x>y)
               System.out.println(" X  is greater than Y");
         else
                  System.out.println(" Y  is greater than X");
   }
}
```

Output

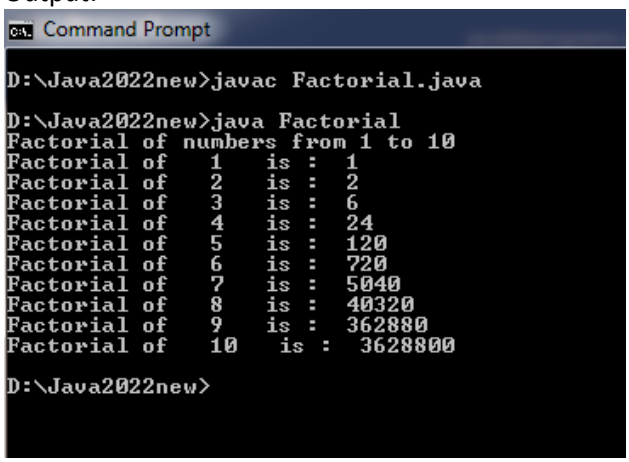


2. **Program to list the factorial of the numbers 1 to 10. To calculate the factorial value, use while loop. (Hint Fact of 4 = 4*3*2*1)**

Code :

```
import java.io.*;

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

          System.out.println("Factorial of numbers from 1 to 10");

           int i=1;

           while(i<=10)
           {
                int k=i;
                int fact=1;
                while(k>=1)
                 {
                      fact = fact*k;
                      k=k-1;
                 }
            System.out.println("Factorial of    " + i + "    is :   " +
                                    fact);

                i = i + 1;
           }
    }
}
```

Output:



3. **Program to add two integers and two float numbers. When no arguments are supplied, give a default value to calculate the sum. Use function overloading.**

Code:

```
class Overload
{

     int add( )
     {
         return 10+20;
   }

    int add(int x , int y)
    {
                return x + y;
    }

    float add(float x , float y)
    {
            return x + y;
    }
}

class AddNums
{

    public static  void  main (String args[])
     {
           int a,b;
           float p,q;

           Overload obj = new Overload();

          System.out.println("The addition of two numbers when no arguments passed
                                is :"+obj.add());

           a=100;
           b=200;
           System.out.println("The addition of two integers is :" + obj.add(a,b));

           p=10.34f;
           q=20.56f;
           System.out.println("The addition of two floating numbers  is :" +
                                obj.add(p,q));
     }
}
```

Output

```
Command Prompt

E:\Java2022>javac AddNums.java

E:\Java2022>java AddNums
The addition of two numbers when no arguments passed is :30
The addition of two integers is :300
The addition of two floating numbers  is :30.9

E:\Java2022>
```

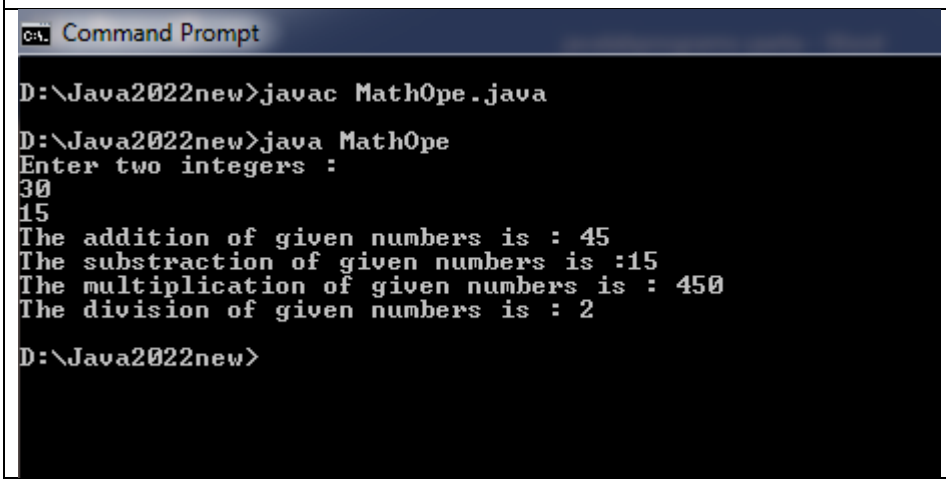
4.

Program to perform mathematical operations. Create a class called AddSub with methods to add and subtract. Create another class called MulDiv that extends from AddSub class to use the member data of the super class. MulDiv should have methods to multiply and divide A main function should access the methods and perform the mathematical operations.

Code :

```
Code :
import java.io.*;
import java.util.Scanner;

class AddSub
{
     int add(int x , int y)
      {
           return x+y;
     }

     int sub(int x , int y)
      {
           return x-y;
     }

}

class MulDiv extends AddSub
{
      int mul(int x , int y)
       {
              return x*y;
      }

     int div(int x , int y)
      {
             return x/y;
     }
}

class MathOpe
{
     public static void main(String args[]) throws IOException
     {
           Scanner sc = new Scanner(System.in);
           System.out.println("Enter two integers : ");

           int a = sc.nextInt();
           int b = sc.nextInt();

           MulDiv obj = new MulDiv();
           System.out.println("The addition of given numbers is : " +
                                  obj.add(a,b));
           System.out.println("The substraction of given numbers is
                                :" + obj.sub(a,b));
```

```
            System.out.println("The multiplication of given numbers is
                                : " + obj.mul(a,b));
            System.out.println("The division of given numbers is : " +
                                    obj.div(a,b));

      }


}
```



5. Program with class variable that is available for all instances of a class. Use static variable declaration. Observe the changes that occur in the object's member variable values

Code:

```
import java.io.*;

class Student
{
    static final String collegeCode="P20GTN0167";
     int regno;
     String name;

    Student(int regno, String name)
     {
          this.regno=regno;
           this.name=name;
     }

     void display()
     {
         System.out.println(collegeCode + "   :   " + regno + " : "
                    + name);

     }
}
public class StaticDemo
{
   public static void main(String args[])
   {
         Student s1 = new Student(1011 , "Anand");
         Student s2 = new Student(1012 , "Manjunath");
         Student s3 = new Student(1013,"Raghu");

         System.out.println("All the objects are sharing a static
                                variable collegeCode : P20GTN0167");
         System.out.println("CollegeCode"+"  :  "+"Reg No " + " : " +
                                "Name of the student");
         s1.display();
         s2.display();
         s3.display();
   }
}
```

Output

```
E:\Java2022>javac StaticDemo.java

E:\Java2022>java StaticDemo
All the objects are sharing a static variable collegeCode : P20GTN0167
CollegeCode  :  Reg No  : Name of the student
P20GTN0167   :   1011 : Anand
P20GTN0167   :   1012 : Manjunath
P20GTN0167   :   1013 : Raghu
```

6. Write a Program

a. To find the area and circumference of the circle by accepting the radius from the user.

b. To accept a number and find whether the number is Prime or not

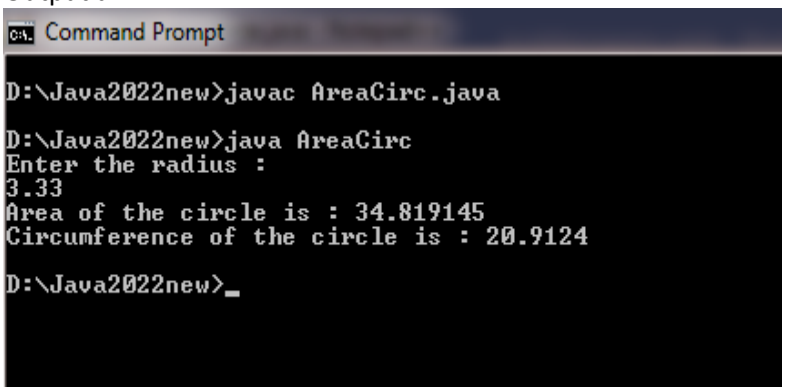
Code

```
/* To find the area and circumference of the circle by accepting the
radius from the user */

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

class AreaCirc
{
   public static void main(String args[]) throws IOException
    {
        Scanner sc = new Scanner(System.in);
        final float pi = 3.14f;

        System.out.println("Enter the radius :");
        float radius = sc.nextFloat();


        float area=pi*radius*radius;
        float circ = 2*pi*radius;

        System.out.println("Area of the circle is : " + area);
        System.out.println("Circumference of the circle is : " +
                                    circ);

   }

}
```

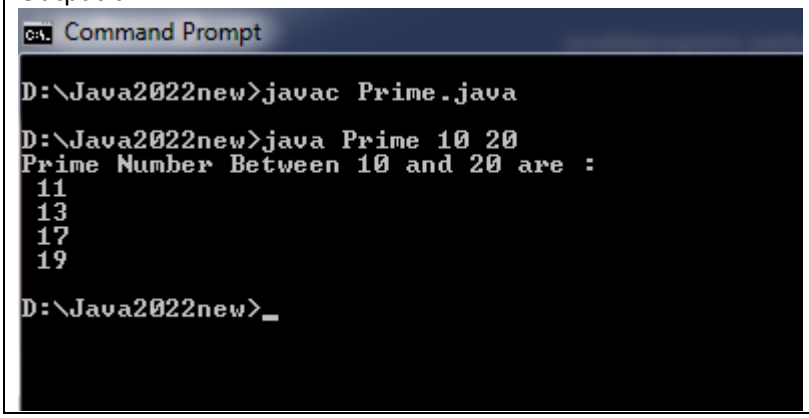
```
/* To accept a number and find whether the number is Prime or not */
class Prime
{
   public static void main(String args[])
   {
     int i,j;
     if(args.length<2)
        {
          System.out.println("No Command line arguments ");
           return;
        }
        int num1=Integer.parseInt(args[0]);
       int num2=Integer.parseInt (args[1]);
       System.out.println("Prime Number Between "+num1+" and "+num2+"
                                    are : ");
       for(i=num1; i<=num2; i++)
        {
            for(j=2; j<i; j++)
              {
```

```
                    int n=i%j;
                    if(n==0)
                    {
                          break;
                    }
                 }
            if(i==j)
              {
                     System.out.println(" " +i);
               }
            }
     }
}
```

Output a :



Output b:



7. Program to create a student class with following attributes;
Enrollment No: Name, Mark of sub1, Mark of sub2, mark of sub3, Total Marks. Total of the three marks must be calculated only when the student passes in all three subjects. The pass mark for each subject is 50. If a candidate fails in any one of the subjects his total mark must be declared as zero. Using this condition write a constructor for this class. Write separate functions for accepting and displaying student details. In the main method create an array of three student objects and display the details.

Code:

```
import java.io.*;

class Student
{
    int enrolno;
    int sub1, sub2, sub3, tot;

    void getdata()
     {
           try
           {
             BufferedReader br = new BufferedReader(new
                                 InputStreamReader(System.in));
             System.out.println("Enter the enrollment number:");
             enrolno = Integer.parseInt(br.readLine());
             System.out.println("Enter the marks in Subject  - 1");
             sub1=Integer.parseInt(br.readLine());
             System.out.println("Enter the marks in Subject  - 2");
             sub2=Integer.parseInt(br.readLine());
             System.out.println("Enter the marks in Subject  - 3");
             sub3=Integer.parseInt(br.readLine());
          }
          catch(Exception e)
           {}
     }

    void putdata()
     {
        System.out.println( enrolno + "           " + sub1 + "
                           " + sub2 + "     " + sub3
+ "     " + tot);

    }
}

class StudentDemo
{
      public static void main(String args[])
      {
             Student[] st = new Student[3];

              System.out.println("Enter marks for three Students");
              for(int i=0;i<st.length;i++)
               {
                       st[i]=new Student();
                       st[i].getdata();
               }
```

```
                System.out.println(" EnrollNo      SUB1      SUB2      SUB3
                                              Total ");
                System.out.println("-----------------------------------------
                                             ----------------------");
                  for(int i=0;i<st.length;i++)
                  {
                       if(st[i].sub1>=50 & st[i].sub2>=50&st[i].sub3>=50)
                              {
                                   st[i].tot=st[i].sub1+st[i].sub2+st[i].sub3;
                               }
                       else
                                st[i].tot=0;

                    st[i].putdata();
                   }
                System.out.println("-----------------------------------------
                                             --------------------------");

        }
}
```

Output

```
Command Prompt

D:\Java2022new>javac StudentDemo.java

D:\Java2022new>java StudentDemo
Enter marks for three Students
Enter the enrollment number:
1011
Enter the marks in Subject  - 1
90
Enter the marks in Subject  - 2
80
Enter the marks in Subject  - 3
70
Enter the enrollment number:
1012
Enter the marks in Subject  - 1
80
Enter the marks in Subject  - 2
90
Enter the marks in Subject  - 3
70
Enter the enrollment number:
1013
Enter the marks in Subject  - 1
70
Enter the marks in Subject  - 2
90
Enter the marks in Subject  - 3
90
 EnrollNo       SUB1       SUB2      SUB3      Total
-----------------------------------------------------
1011            90         80      70      240
1012            80         90      70      240
1013            70         90      90      250
-----------------------------------------------------

D:\Java2022new>_
```

8. Program to define a class called employee with the name and date of appointment. Create ten employee objects as an array and sort them as per their date of appointment. ie, print them as per their seniority.

Code :

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

class Employee
{
    String name;
    Date doj;

    public Employee(String name, Date doj)
    {
        this.name=name;
        this.doj=doj;
    }

    public void display()
    {
      System.out.println("Employee : " + name +"\t"+ " DOJ : " +
                doj.getDate() + ":" + doj.getMonth()+":"+doj.getYear());
    }

}

public class EmpSeniority
{
     public  static void  main(String args[])
     {
         Employee [] emp = new Employee[10];
         emp[0]= new Employee("Manjunath B", new Date(2010,01,01));
         emp[1]= new Employee("Raghu DB", new Date(2011,02,01));
         emp[2]= new Employee("Kishore K", new Date(2009,05,01));
         emp[3]= new Employee("Madhu", new Date(2014,03,01));
         emp[4]= new Employee("Shafi", new Date(2015,05,01));
         emp[5]= new Employee("Krishna", new Date(2016,04,01));
         emp[6]= new Employee("Anand", new Date(2020,01,01));
         emp[7]= new Employee("Kavitha", new Date(2017,04,01));
         emp[8]= new Employee("Shilpa", new Date(2017,03,01));
         emp[9]= new Employee("Vasudha", new Date(2015,02,01));

         System.out.println("List of the Given Employees : ");
         for(int i=0 ; i<emp.length;i++)
                       emp[i].display();

         //finding senior most employee
          for(int i=0; i<emp.length;i++)
           {
               for(int j=i+1;j<emp.length;j++)
                     {

                         if(emp[i].doj.after(emp[j].doj))
                              {
                                  Employee t = emp[i];
```

```
                                    emp[i]=emp[j];
                                    emp[j]=t;
                                }

                            }
                        }
                System.out.println("******** Employee Seniority List *************");
                 for(int i=0 ; i<emp.length;i++)
                        emp[i].display();
}
}
```

Output:

```
E:\Java2022>java EmpSeniority
List of the Given Employees :
Employee : Manjunath B   DOJ : 1:1:2010
Employee : Raghu DB      DOJ : 1:2:2011
Employee : Kishore K     DOJ : 1:5:2009
Employee : Madhu         DOJ : 1:3:2014
Employee : Shafi         DOJ : 1:5:2015
Employee : Krishna       DOJ : 1:4:2016
Employee : Anand         DOJ : 1:1:2020
Employee : Kavitha       DOJ : 1:4:2017
Employee : Shilpa        DOJ : 1:3:2017
Employee : Vasudha       DOJ : 1:2:2015
******** Employee Seniority List *************
Employee : Kishore K     DOJ : 1:5:2009
Employee : Manjunath B   DOJ : 1:1:2010
Employee : Raghu DB      DOJ : 1:2:2011
Employee : Madhu         DOJ : 1:3:2014
Employee : Vasudha       DOJ : 1:2:2015
Employee : Shafi         DOJ : 1:5:2015
Employee : Krishna       DOJ : 1:4:2016
Employee : Shilpa        DOJ : 1:3:2017
Employee : Kavitha       DOJ : 1:4:2017
Employee : Anand         DOJ : 1:1:2020

E:\Java2022>
```

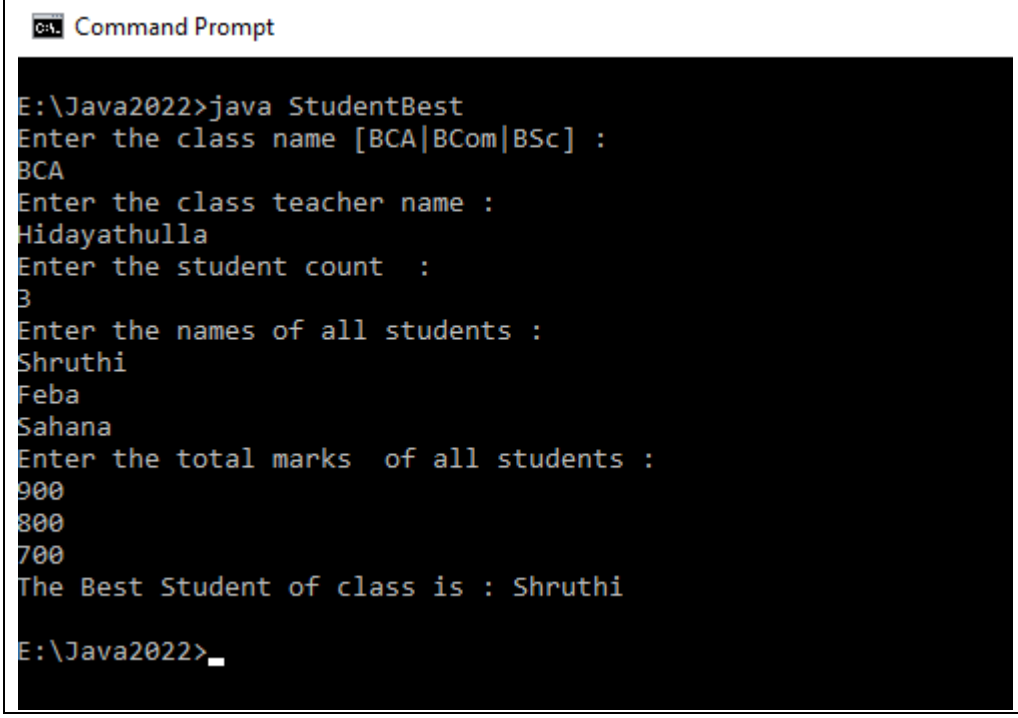
9. In a college first year class are having the following attributes Name of the class (BCA, BCom, BSc), Name of the staff No of the students in the class, Array of students in the class
Define a class called first year with above attributes and define a suitable constructor. Also write a method called best Student () which process a first-year object and return the student with the highest total mark. In the main method define a first-year object and find the best student of this class

Code:

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

class FirstYear
{
   String className;
   String classTeacher;

   int stdCount;
   String[] stdNames = new String[20];
   int [] stdMarks = new int[20];

   Scanner sc = new Scanner(System.in);

   public FirstYear()
   {
      getData();

   }

   public void getData()
   {
        System.out.println("Enter the class name [BCA|BCom|BSc] :");
        className=sc.next();

        System.out.println("Enter the class teacher name :");
        classTeacher=sc.next();

        System.out.println("Enter the student count  :");
        stdCount=sc.nextInt();

        System.out.println("Enter the names of all students :");
        for(int i=0;i<stdCount;i++)
            stdNames[i]=sc.next();

        System.out.println("Enter the total marks  of all students :");
        for(int i=0;i<stdCount;i++)
             stdMarks[i]=sc.nextInt();

  }

  public void bestStudent()
  {
        int best=0, k=-1;
        for(int i=0;i<stdCount;i++)
         {
            if(stdMarks[i]>best)
             {
               best=stdMarks[i];
               k=i;
             }
```

```
        }
         System.out.println("The Best Student of class is : "+stdNames[k]);

  }


}//end of class FirstYear

public class StudentBest
{
     public static void main(String args[])
     {
          FirstYear obj = new FirstYear();
            obj.bestStudent();

     }
}
```

Output



10. Create a package 'student. Fulltime. BCA 'in your current working directory
a. Create a default class student in the above package with the following attributes: Name, age, sex.
b. Have methods for storing as well as displaying

Hint:
This code should be saved as BCAStudent.java and compile it as follows
C:\> javac –d . BCAStudent.java
It creates a directory called student with sub directory Fulltime and in that subdirectory BCA and in that BCAStudent class exists

Code

```
package student.Fulltime.BCA;
import java.util.*;

public class BCAStudent
{
    String name;
    int age;
    String sex;
    Scanner sc = new Scanner(System.in);

    public void getData()
    {
       System.out.println("Enter student name :");
       name = sc.next();
       System.out.println("Enter student Gender :");
       sex=sc.next();
       System.out.println("Enter student Age :");
       age=sc.nextInt();
     }

    public void display()
    {
       System.out.println("The Given Student Details are :");
       System.out.println("Name :"+name);
       System.out.println("Age :"+age);
       System.out.println("Sex :"+sex);
    }
}
```

Hint : This file should be named as MypackageDemo.java and compiled and executed as a normal java program

Code:

```
import student.Fulltime.BCA.BCAStudent;

public class MypackageDemo
{
    public static void main(String args[])
                              {
                                   BCAStudent stud = new BCAStudent();
                                stud.getData();
                                stud.display();
                              }

}
```

Output

Command Prompt

```
E:\Java2022>java MypackageDemo
Enter student name :
Anand
Enter student Gender :
Male
Enter student Age :
19
The Given Student Details are :
Name :Anand
Age :19
Sex :Male

E:\Java2022>
```

****

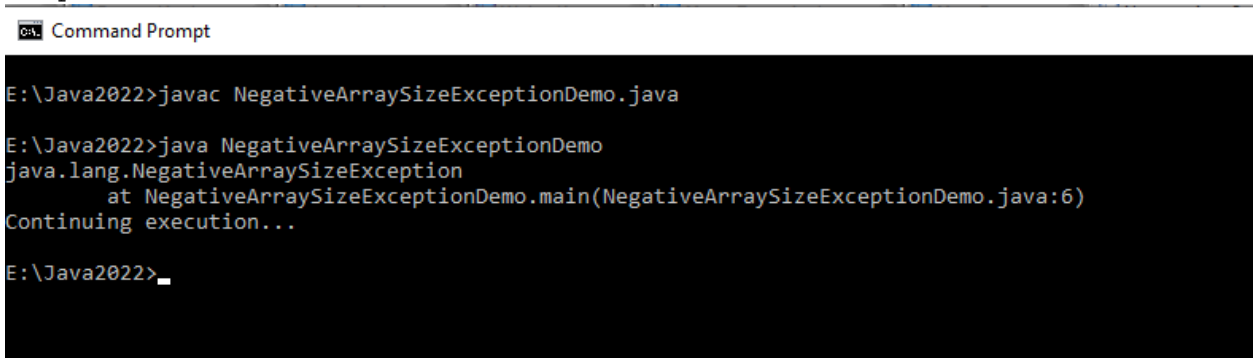
Java Lab Programs – Part B

1. Program to catch Negative Array Size Exception. This exception is caused when the array is initialized to negative values.

Code :

```
public class NegativeArraySizeExceptionExample
{
    public static void main(String[] args)
   {
        try {
                  int[] array = new int[-10];
              } catch (NegativeArraySizeException e)
             {
                       e.printStackTrace();
                      //handle the exception
              }
        System.out.println("Continuing execution...");
    }
}
```

Output:

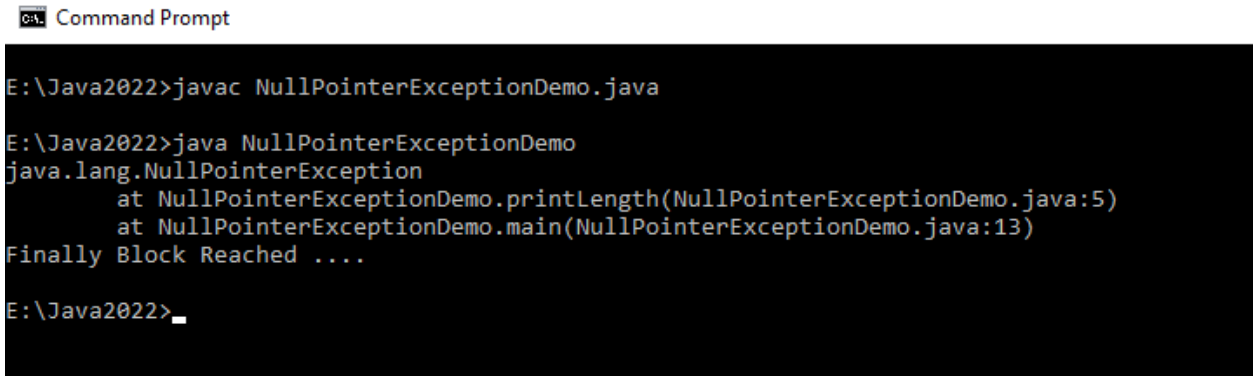


2. Program to handle Null Pointer Exception and use the "finally" method to display a message to the user.

Code :

```
public class NullPointerExceptionExample
{
    private static void printLength(String str)
    {
            System.out.println(str.length());
     }

    public static void main(String args[])
    {
        String myString = null;
         try
         {
             printLength(myString);
         }catch(NullPointerException e)
                 {
                      e.printStackTrace();
                 }
         finally
         {
              System.out.println("Finally Block Reached ....");

       }
    }
}
```

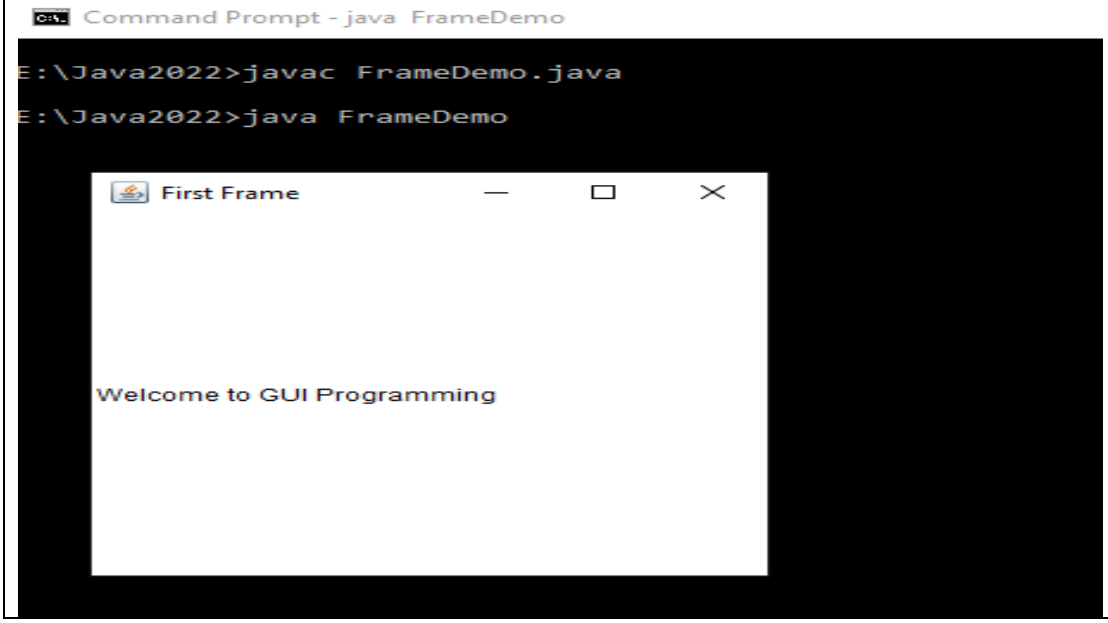
Output:



3. Program which create and displays a message on the window

```
import java.awt.*;

public class FrameDemo
{
    FrameDemo()
    {
            Frame fm = new Frame();
            fm.setTitle("First Frame");
            Label lb = new Label("Welcome to GUI Programming");
            fm.add(lb);
            fm.setSize(300,300);
            fm.setVisible(true);

    }
    public static void main(String args[])
     {
            FrameDemo ta = new FrameDemo();


     }

}
```

Output

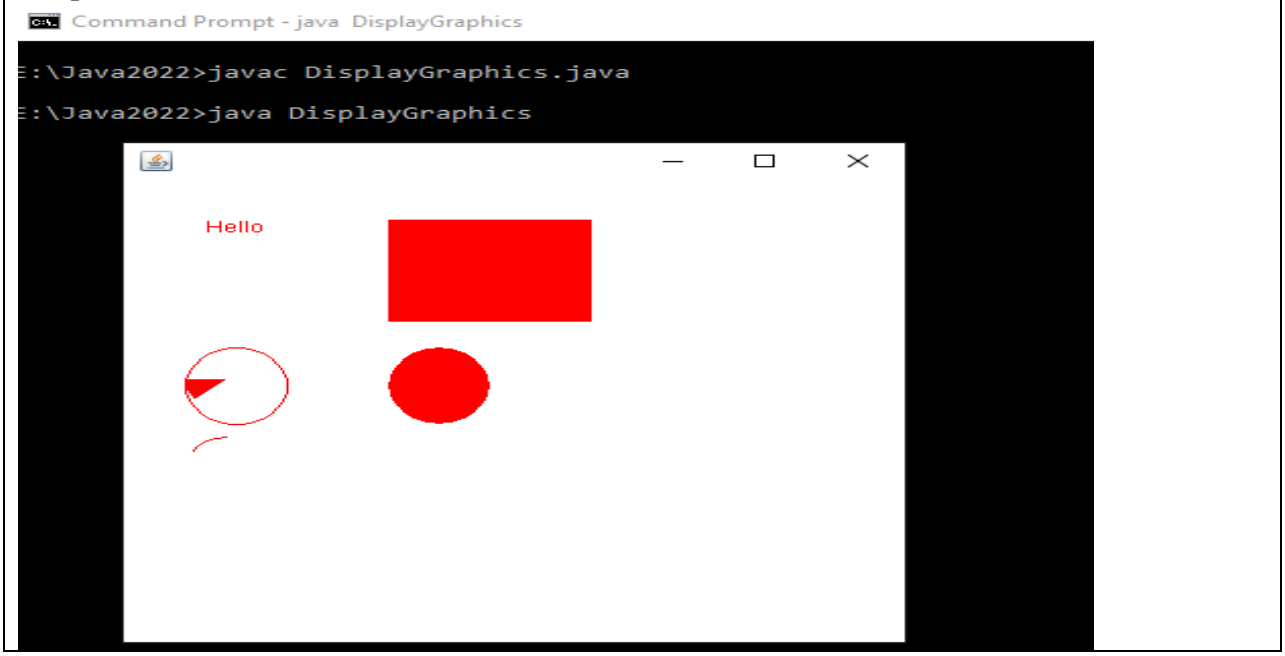


4. Program to draw several shapes in the created window

Code :

```
import java.awt.*;
import javax.swing.JFrame;

public class DisplayGraphics extends Canvas{

    public void paint(Graphics g) {
        g.drawString("Hello",40,40);
        setBackground(Color.WHITE);
        g.fillRect(130, 30,100, 80);
        g.drawOval(30,130,50, 60);
        setForeground(Color.RED);
        g.fillOval(130,130,50, 60);
        g.drawArc(30, 200, 40,50,90,60);
        g.fillArc(30, 130, 40,50,180,40);

    }
        public static void main(String[] args) {
        DisplayGraphics m=new DisplayGraphics();
        JFrame f=new JFrame();
        f.add(m);
        f.setSize(400,400);
        //f.setLayout(null);
        f.setVisible(true);
    }

}
```

Output

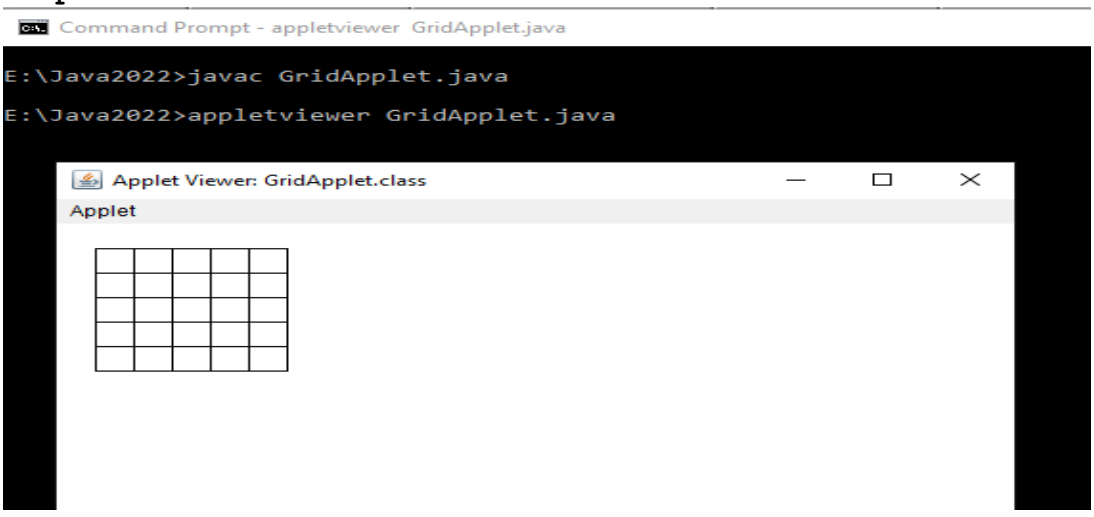

Command Prompt - java DisplayGraphics
E:\Java2022>javac DisplayGraphics.java
E:\Java2022>java DisplayGraphics
Hello

5. Program to create an applet and draw grid lines

Code :

```
import java.awt.*;
import java.applet.*;
/*
  <applet code="GridApplet.class" height=500 width=500></applet>

  */
public class GridApplet extends Applet
{
      public void paint(Graphics g)
      {
              int row, col, x, y = 20;

              for(row=1;row<5;row++)
               {
                        x=20;
                        for(col=1;col<5;col++)
                        {
                                g.drawRect(x,y,40,40);
                                x=x+20;
                       }
                       y=y+20;
                }
     }
}
```

Output:



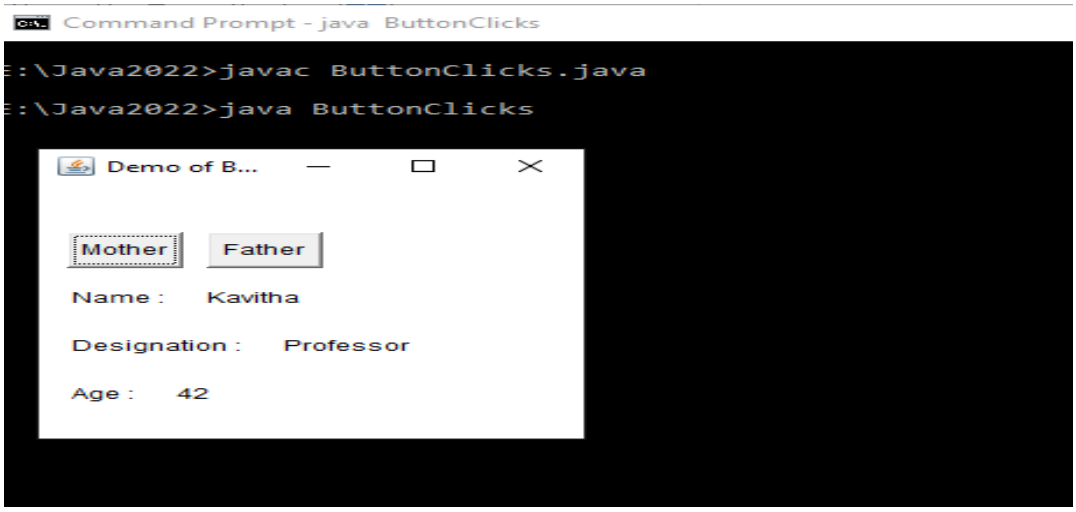
6. Program which creates a frame with two buttons father and mother. When we click the father button the name of the father, his age and designation must appear. When we click mother similar details of mother also appear.

Code :

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

public class ButtonClicks
{
    public static void main(String args[])
      {
          Frame f = new Frame("Demo of Button");
           Label l = new Label("Welcome to My Family Page");
           l.setFont(new Font("Times Roman", Font.BOLD,18));
           final Label lbl1 = new Label();
           final Label lbl2 = new Label();
           final Label lbl3 = new Label();


           l.setBounds(250,30,600,50);
           lbl1.setBounds(20,110,500,30);
           lbl2.setBounds(20,150,500,30);
           lbl3.setBounds(20,190,500,30);

           Button mb = new Button("Mother");
           mb.setBounds(20,70,50,30);
           mb.addActionListener(new ActionListener()
                                { public void actionPerformed(ActionEvent e)
                                                {
               lbl1.setText("Name :" + "     " + " Kavitha ");
              lbl2.setText("Designation :" + "     " + " Professor ");
              lbl3.setText("Age :" + "     " + " 42 ");
                                                  }
                                        }
                           );

          Button fb = new Button("Father");
           fb.setBounds(80,70,50,30);
           fb.addActionListener(new ActionListener()
                                { public void actionPerformed(ActionEvent e)
                                                {
                   lbl1.setText("Name :" + "     " + "Rajesh");
                   lbl2.setText("Designation :" + "     " + "Bank Manager");
                   lbl3.setText("Age :" + "     " + " 45 ");
                                                  }
                                        }
                           );

      // adding elements to frame
      f.add(mb);
```

```
        f.add(fb);
        f.add(l);
        f.add(lbl1);
        f.add(lbl2);
        f.add(lbl3);

        //set size , visibility and layout
        f.setSize(250,250);
        f.setLayout(null);
        f.setVisible(true);
    }
}
```

Output:

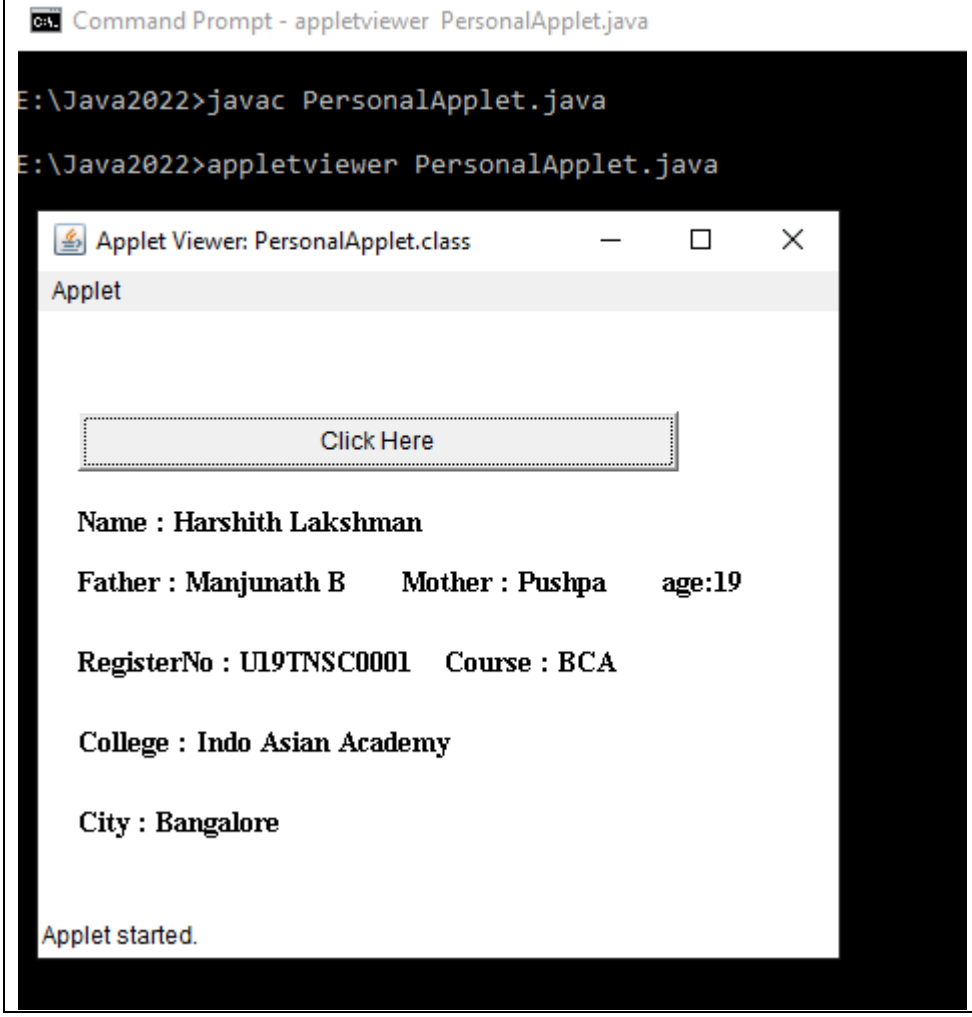


7. Program which creates a frame with two buttons father and mother. When we click the father button the name of the father, his age and designation must appear. When we click mother similar details of mother also appear.

Code :

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

/* <Applet code ="PersonalApplet.class" width=400 height=400> </Applet> */
public class PersonalApplet extends Applet implements ActionListener
{
            String s1="" , s2=" ", s3=" ", s4=" ", s5=" ";

            public void init()
            {
                setLayout(null);
                setSize(400,300);
                Button btn = new Button("Click Here");
                add(btn);
                btn.setBounds(20,50,300,30);
                btn.addActionListener(this);
            }

            public void actionPerformed(ActionEvent e)
            {
                s1="Name : Harshith Lakshman";
                s2= "Father : Manjunath B       Mother : Pushpa       age:19";
                s3="RegisterNo : U19TNSC0001    Course : BCA";
                s4="College : Indo Asian Academy";
                s5="City : Bangalore";
                repaint();

            }

            public void paint(Graphics g)
            {
                g.setFont(new Font("TimesRoman",Font.BOLD,14));
                g.drawString(s1,20,110);
                g.drawString(s2,20,140);
                g.drawString(s3,20,180);
                g.drawString(s4,20,220);
                g.drawString(s5,20,260);

            }


}
```

Output :


Command Prompt - appletviewer PersonalApplet.java
E:\Java2022>javac PersonalApplet.java
E:\Java2022>appletviewer PersonalApplet.java
Applet Viewer: PersonalApplet.class
Applet
Click Here
Name : Harshith Lakshman
Father : Manjunath B Mother : Pushpa age:19
RegisterNo : U19TNSC0001 Course : BCA
College : Indo Asian Academy
City : Bangalore
Applet started.

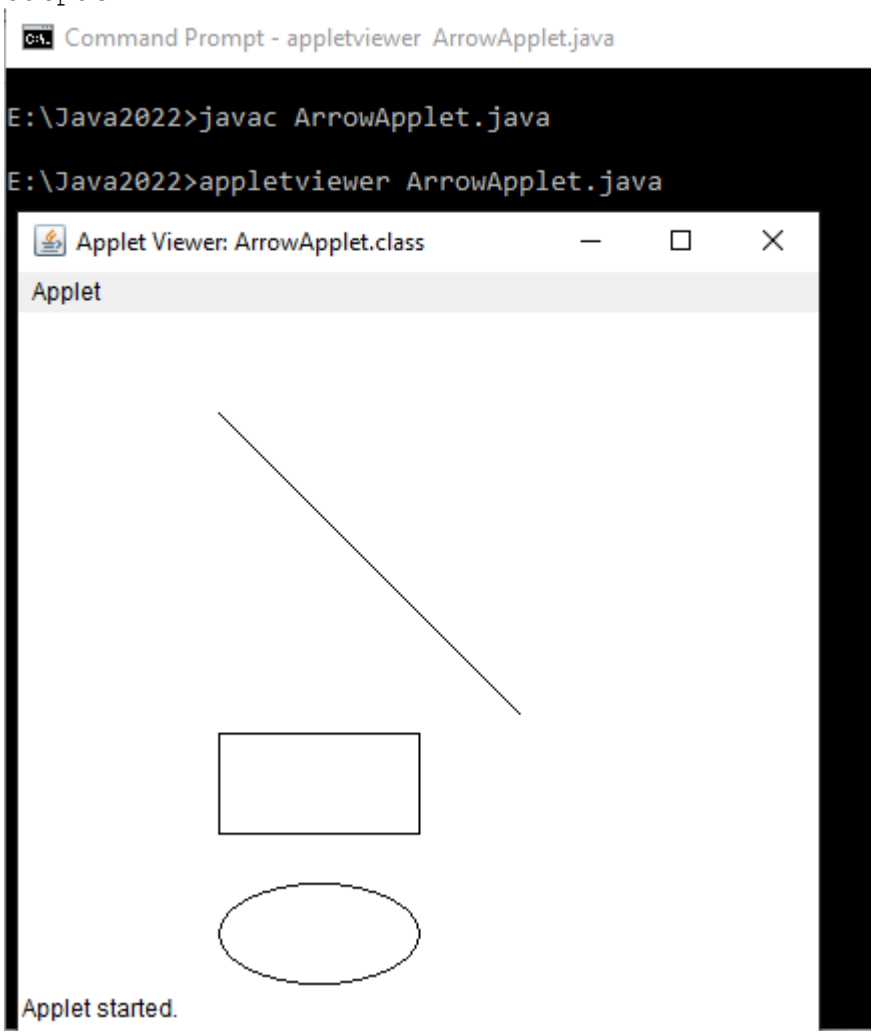
8 : Program to move different shapes according to the arrow key pressed.

Code :

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

/* <applet code="ArrowApplet.class" height=400 width=400> </applet> */

public class ArrowApplet extends Applet implements KeyListener
{
            int x1=100, y1=50, x2=250, y2=200;
            public void init()
            {
                addKeyListener(this);

            }

            public void keyPressed(KeyEvent ke)
            {
                showStatus("keydown");
                int key = ke.getKeyCode();

                switch(key)
                {
                      case KeyEvent.VK_LEFT:x1=x1-10; x2=x2-10;
                                            break;
                      case KeyEvent.VK_RIGHT:x1=x1+10; x2=x2+10;
                                            break;
                      case KeyEvent.VK_UP:y1=y1-10; y2=y2-10;
                                            break;
                      case KeyEvent.VK_DOWN:y1=y1+10; y2=y2+10;
                                            break;


                }

                repaint();
            }

            public void keyReleased(KeyEvent ke)
            {

            }
    public void keyTyped(KeyEvent ke)
            {
                repaint();
            }

    public void paint(Graphics g)
            {
                g.drawLine(x1,y1,x2,y2);
                g.drawRect(x1,y1+160,100,50);
```

```
                g.drawOval(x1,y1+235,100,50);


            }

}
```

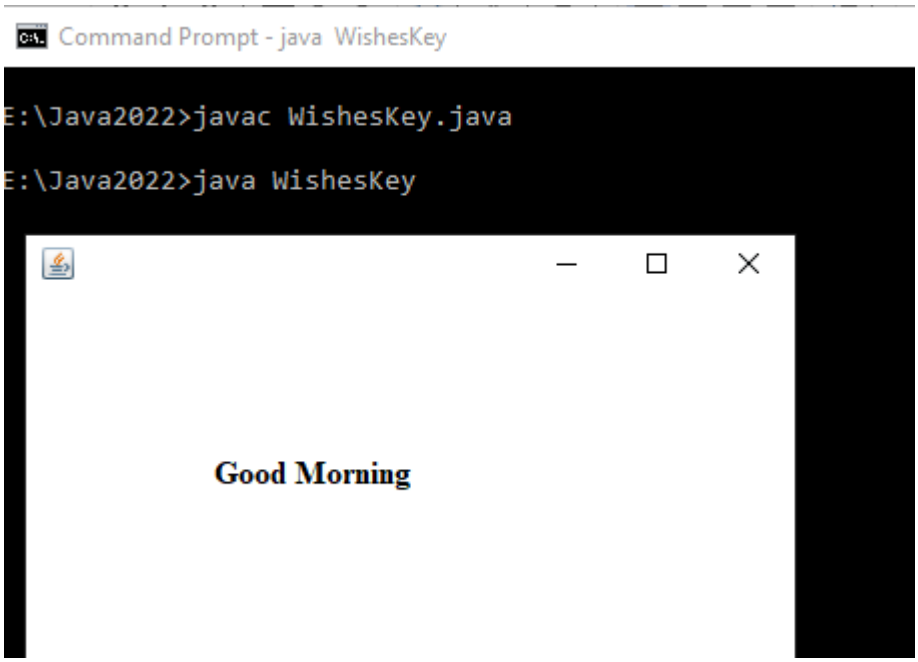
Output



9. Program to create a window when we press M or m the window displays Good Morning, A or a the window displays Good After Noon E or e the window displays Good Evening, N or n the window displays Good Night

```
Code :
import java.awt.*;
import java.awt.event.*;

public class WishesKey extends Frame implements KeyListener
{
   Label lbl;

   WishesKey()
   {
       addKeyListener(this);
                requestFocus();
                lbl = new Label();
                lbl.setBounds(100,100,200,40);
                lbl.setFont(new Font("TimesRoman", Font.BOLD, 16));
                add(lbl);
                setSize(400,400);
                setLayout(null);
                setVisible(true);

   }

   public void keyPressed(KeyEvent e)
   {
                if(e.getKeyChar()=='M' || e.getKeyChar()=='m' )
                      lbl.setText("Good Morning");
                else if(e.getKeyChar()=='A' || e.getKeyChar()=='a' )
                   lbl.setText("Good Afternoon");
                   else if(e.getKeyChar()=='E' || e.getKeyChar()=='e' )
                                     lbl.setText("Good Evening");
                      else  if(e.getKeyChar()=='N' || e.getKeyChar()=='e' )
                                lbl.setText("Good Night");


   }
   public void keyReleased(KeyEvent e)
   {}
   public void keyTyped(KeyEvent e)
   {}

   public static void main(String args[])
   {
                new WishesKey();

   }

}
```

Output


Command Prompt - java WishesKey
E:\Java2022>javac WishesKey.java
E:\Java2022>java WishesKey
Good Morning

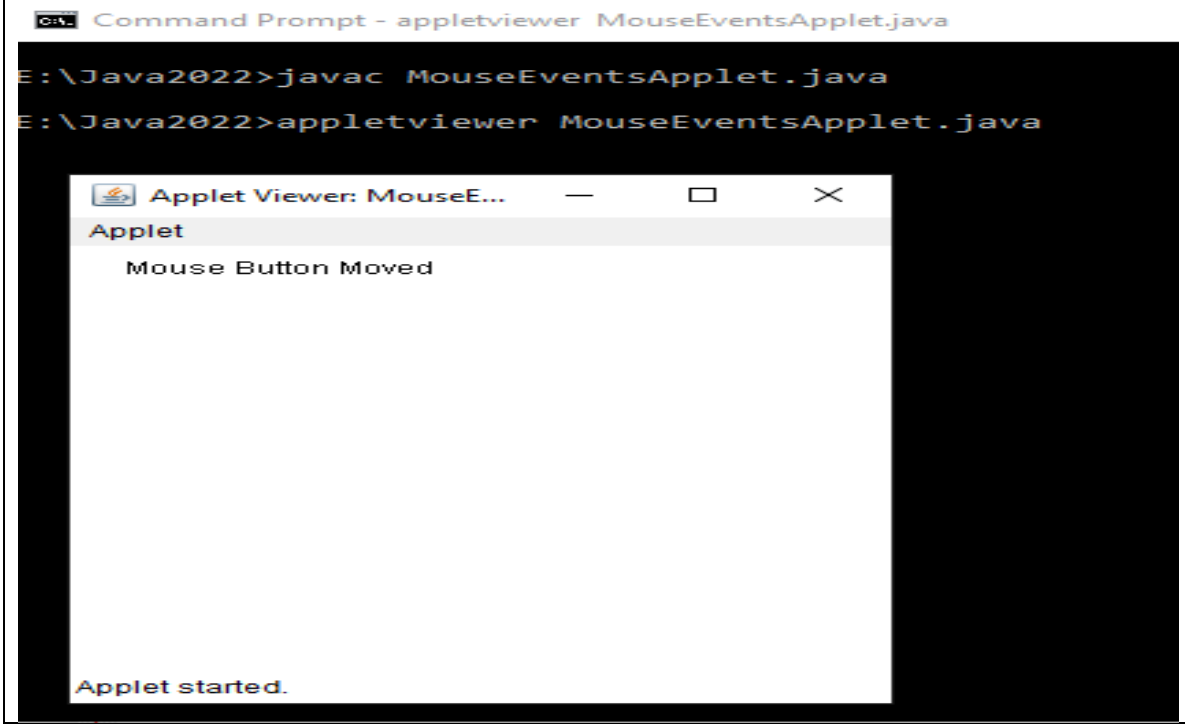
10. Demonstrate the various mouse handling events using suitable example

Code:

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

/* <applet code="MouseEventsApplet.class" height=300 width=300></applet> */

public class MouseEventsApplet extends Applet implements
MouseListener,MouseMotionListener

{
    String str=" ";

    public void init()
    {
       addMouseListener(this);
       addMouseMotionListener(this);
    }

    public void paint(Graphics g)
    {
       g.drawString(str,20,20);

   }

   public void mousePressed(MouseEvent me)
   {
       str="Mouse Button Pressed";
       repaint();
   }

 public void mouseClicked(MouseEvent me)
   {
       str="Mouse Button Clicked";
       repaint();
   }

 public void mouseReleased(MouseEvent me)
   {
       str="Mouse Button Released";
       repaint();
   }

 public void mouseEntered(MouseEvent me)
   {
       str="Mouse Button Entered";
       repaint();
   }
```

```
 public void mouseExited(MouseEvent me)
   {
        str="Mouse Button Exited";
        repaint();
   }

 public void mouseMoved(MouseEvent me)
   {
        str="Mouse Button Moved";
        repaint();
   }
  public void mouseDropped(MouseEvent me)
   {
        str="Mouse Button Dropped";
        repaint();
   }

  public void mouseDragged(MouseEvent me)
   {
        str="Mouse Button Dragged";
        repaint();
   }
}
```

Output:



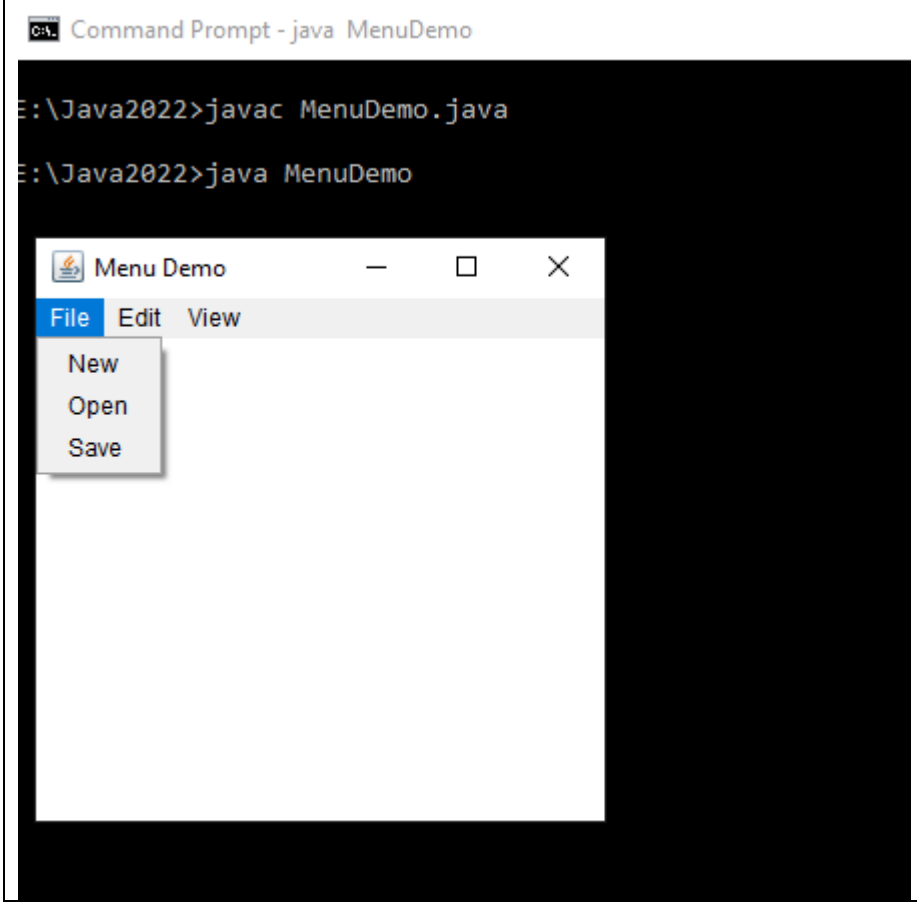
11. Program to create menu bar and pull-down menus

Code :

```
import java.awt.*;
public class MenuDemo
{
    MenuDemo()
                {
                    Frame fr = new Frame("Menu Demo");
                    MenuBar mb = new MenuBar();

                    Menu filemenu = new Menu("File");
                    Menu editmenu = new Menu("Edit");
                    Menu viewmenu = new Menu("View");

                    mb.add(filemenu);
                    mb.add(editmenu);
                    mb.add(viewmenu);

                    MenuItem a1 = new MenuItem("New");
                    MenuItem a2 = new MenuItem("Open");
                    MenuItem a3 = new MenuItem("Save");

                    MenuItem b1 = new MenuItem("Cut");
                    MenuItem b2 = new MenuItem("Copy");
                    MenuItem b3 = new MenuItem("Paste");

                    MenuItem c1 = new MenuItem("Zoom");
                    MenuItem c2 = new MenuItem("Full Screen");

                    filemenu.add(a1);
                    filemenu.add(a2);
                    filemenu.add(a3);

                    editmenu.add(b1);
                    editmenu.add(b2);
                    editmenu.add(b3);

                    viewmenu.add(c1);
                    viewmenu.add(c2);

                    fr.setMenuBar(mb);
                    fr.setSize(300,300);
                    fr.setLayout(null);
                    fr.setVisible(true);


                }

    public static void main(String args[])
    {
                    new MenuDemo();
                }
```

```
}
```

Output


Command Prompt - java MenuDemo
E:\Java2022>javac MenuDemo.java
E:\Java2022>java MenuDemo
Menu Demo
File
Edit
View
New
Open
Save

****