Java Programming Lab Manual - NEP

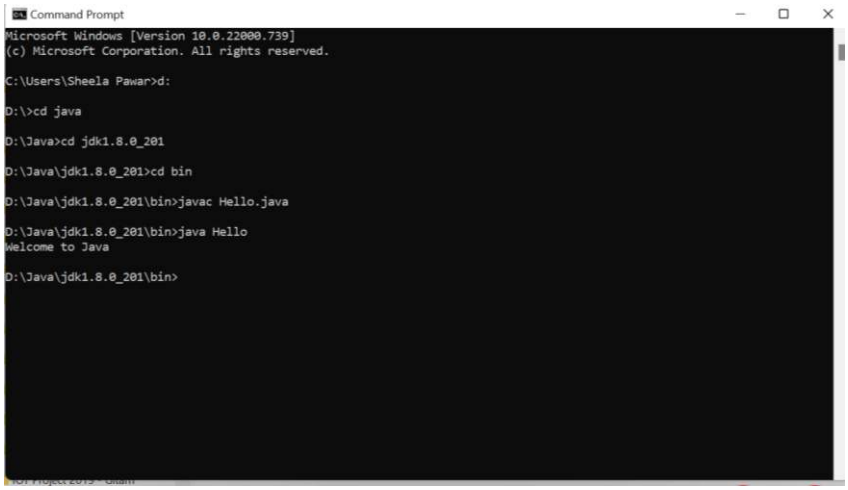
Bachelor of computer applications (Bangalore University)

Java Programming Lab Manual

1. **Write a simple java application to print the message “Welcome to java”.**

```
class Hello{
  public static void main(String args[]){
      System.out.println("Welcome to Java");
   }
}
```

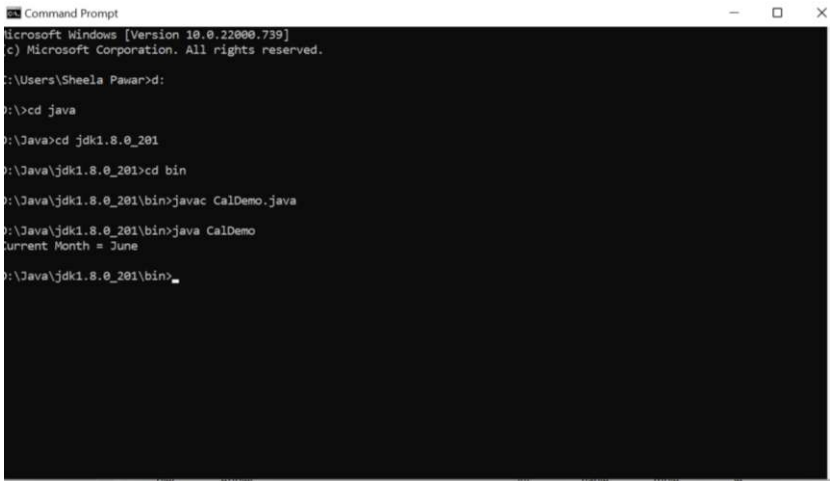
Output:



2. **Write a program to display the month of a year. Months of the year should be held in an array.**

```
import java.util.Calendar;
public class CalDemo {
  public static void main(String[] args) {
     Calendar calendar = Calendar.getInstance();
     String[] month = new String[] {"January", "February", "March", "April", "May", "June",
"July", "August","September", "October", "November", "December" };
            System.out.println("Current Month = " + month[calendar.get(Calendar.MONTH)]);
  }
}
```

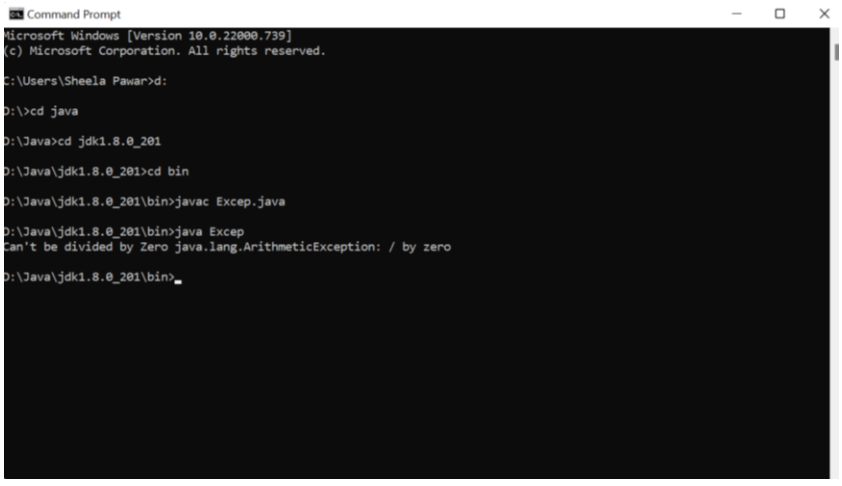
Output:



3. Write a program to demonstrate a division by zero exception

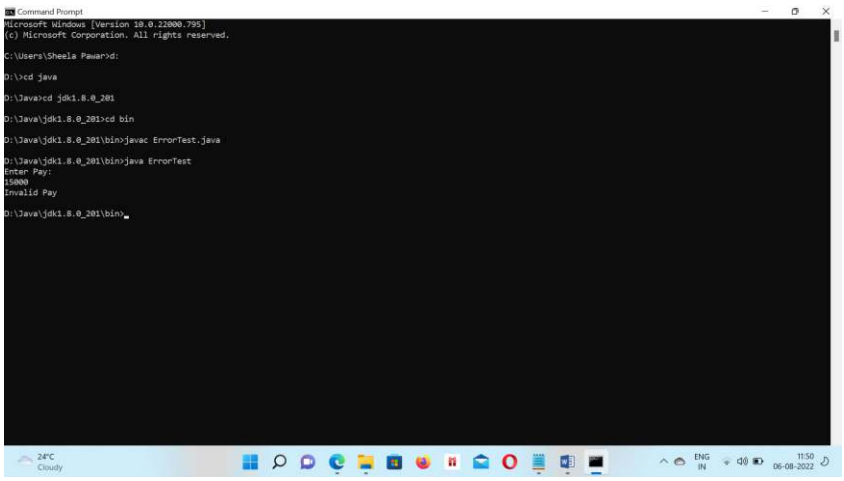
```
class Excep{
  public static void main (String args[]) {
    int num1 = 15, num2 = 0, result = 0;
    try{
       result = num1/num2;
       System.out.println("The result is" +result);
    }
    catch (ArithmeticException e) {
      System.out.println ("Can't be divided by Zero " + e);
    }
  }
}
```

Output:



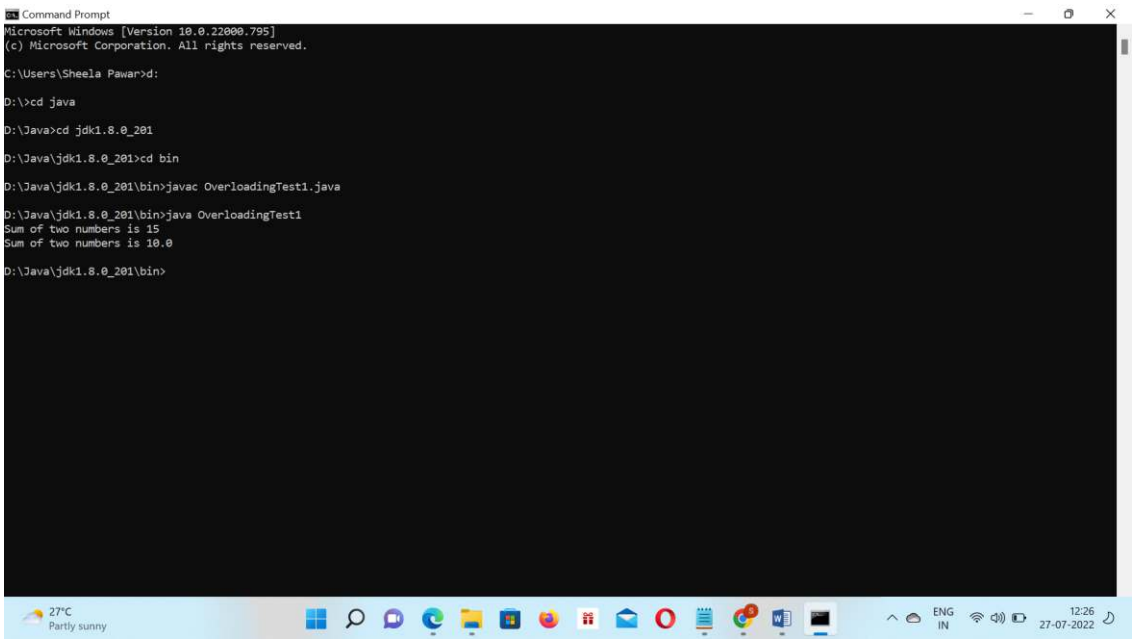
4. Write a program to create a user defined exception say Pay Out of Bounds

```
import java.io.*;
class PayOutOfBounds extends Exception
{
public void showError()
{
System.out.println("Invalid Pay");
}
}
class ErrorTest
{
public static void main(String []args) throws Exception
{
InputStreamReader isr=new InputStreamReader(System.in);
BufferedReader br=new BufferedReader(isr);
int m=0;
try
{
System.out.println("Enter Pay:");
m=Integer.parseInt(br.readLine());
if(m>10000)
throw new PayOutOfBounds();
System.out.println("Your Pay:"+m);
}
catch(PayOutOfBounds e)
{
e.showError();
}
}
}
```



5. **Write a java 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.**

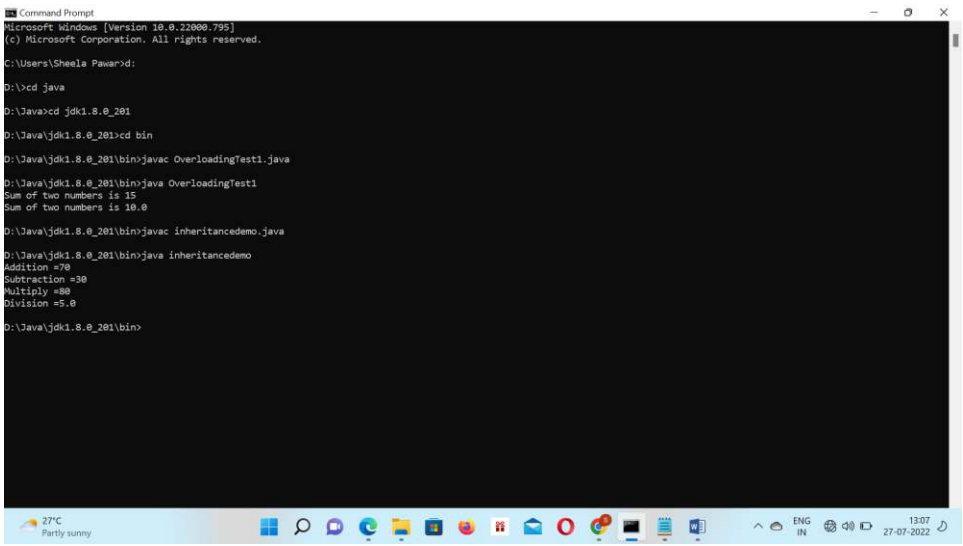
```
public class OverloadingTest1
{

void addition(int a, int b)
{
int sum=a+b;
System.out.println("Sum of two numbers is "+sum);
}

                void addition(double a ,double b)

{
double sum=a+b;
System.out.println("Sum of two numbers is "+sum);
}
public static void main(String args[])
{
 OverloadingTest1 ovl=new OverloadingTest1();
 ovl.addition(5,10);
 ovl.addition(1.5,8.5);
}
}
```



6. **Write a 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.**

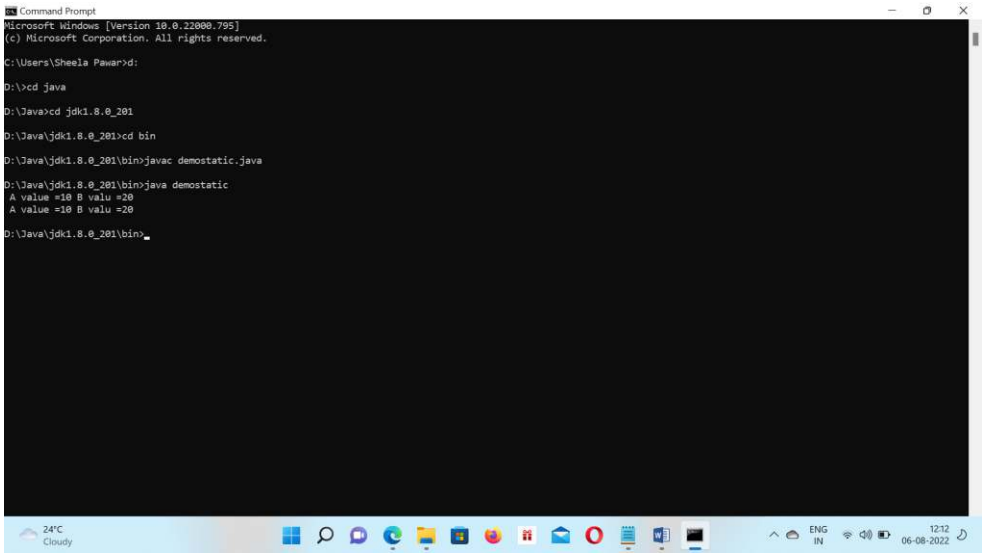
```
class addsub
{
int num1,num2;
addsub(int n1, int n2)
{
num1 = n1;
num2 = n2;
}
int add()
{
return num1+num2;
}
int sub()
{
return num1-num2;
}
}
class multdiv extends addsub
{
public multdiv(int n1, int n2)
{
super(n1, n2);
}
int mul()
{
return num1*num2;
}
float div()
{
return num2/num1;
}
}
public class inheritancedemo
{
public static void main(String arg[])
{
addsub r1=new addsub(50,20);
int ad = r1.add();
int sb = r1.sub();
System.out.println("Addition =" +ad);
System.out.println("Subtraction =" +sb);
```

```
multdiv r2 =new multdiv(4,20);
int ml = r2.mul();
float dv =r2.div();
System.out.println("Multiply =" +ml);
System.out.println("Division =" +dv);
}
}
```



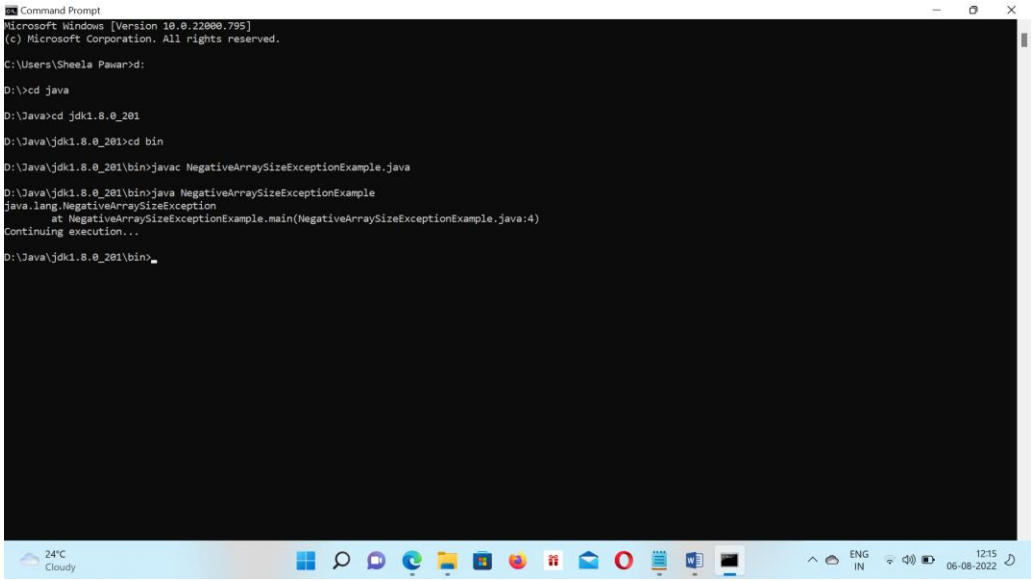
7. Write a program with class variable that is available for all instances of a class. Use static variable declaration. Observe the changes that occur in the object's member variable values.

```
class Staticvar
{
 public static int a,b;
public void display()
{
 System.out.println(" A value ="+a+" B valu ="+b);
}
}
class Demo
{
 public static void main(String args[])

{
  Staticvar sv=new Staticvar();
  sv.a=10;
  sv.b=20;
  sv.display();
  Staticvar sv1=new Staticvar();
sv.display();
}
}
```



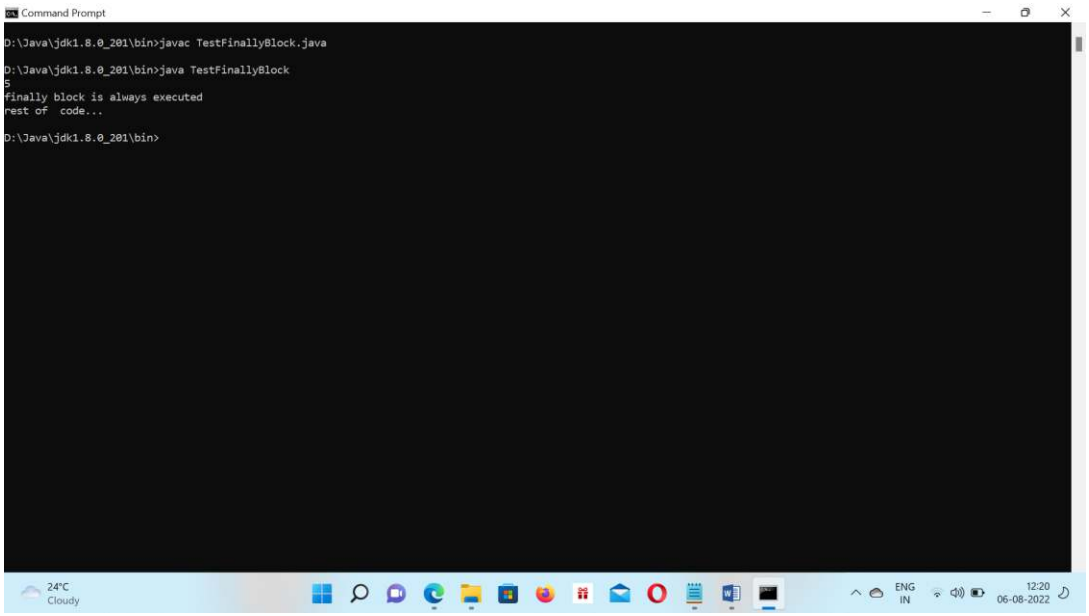
8. Write a small program to catch Negative Array Size Exception. This exception is caused when the array is initialized to negative values.

```
public class NegativeArraySizeExceptionExample {
    public static void main(String[] args) {
        try {
            int[] array = new int[-5];
        } catch (NegativeArraySizeException nase) {
            nase.printStackTrace();
        }
        System.out.println("Continuing execution...");
    }
}
```



9. Write a program to handle Null Pointer Exception and use the "finally" method to display a message to the user.

```
class TestFinallyBlock {
  public static void main(String args[]){
  try{
   int data=25/5;
   System.out.println(data);
  }
  catch(NullPointerException e){
System.out.println(e);
}
 finally {
System.out.println("finally block is always executed");
}

System.out.println("rest of  code...");
  }
}
```

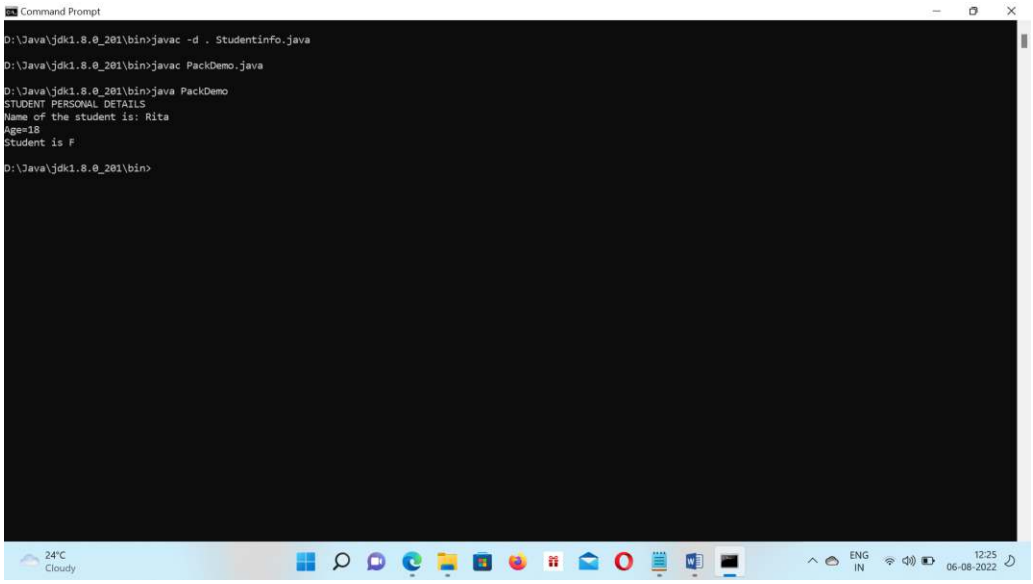


10. Create a package‘ student.Fulltime.BCA‘ in your current working directory a. Create a default class student in the above package with the following attributes: Name, age, sex. b. Have methods for storing as well as displaying

a.
```
package student.Fulltime.BCA;
  public class Studentinfo
  {
   int age;
   char gender;
   String name;
  public Studentinfo(String a,int e,char f)
  {
```

```
 name=a;
 age=e;
 gender=f;
 }
public void display()
 {
  System.out.println("STUDENT PERSONAL DETAILS");
  System.out.println("Name of the student is: "+name);
  System.out.println("Age="+age);
  System.out.println("Student is "+gender);
  }
}
```

b.
```
import student.Fulltime.BCA.*;
class PackDemo
{
 public static void main(String args[])
 {
student.Fulltime.BCA.Studentinfo a
= new student.Fulltime.BCA.Studentinfo("Rita",18,'F');
  a.display();
}
}
```

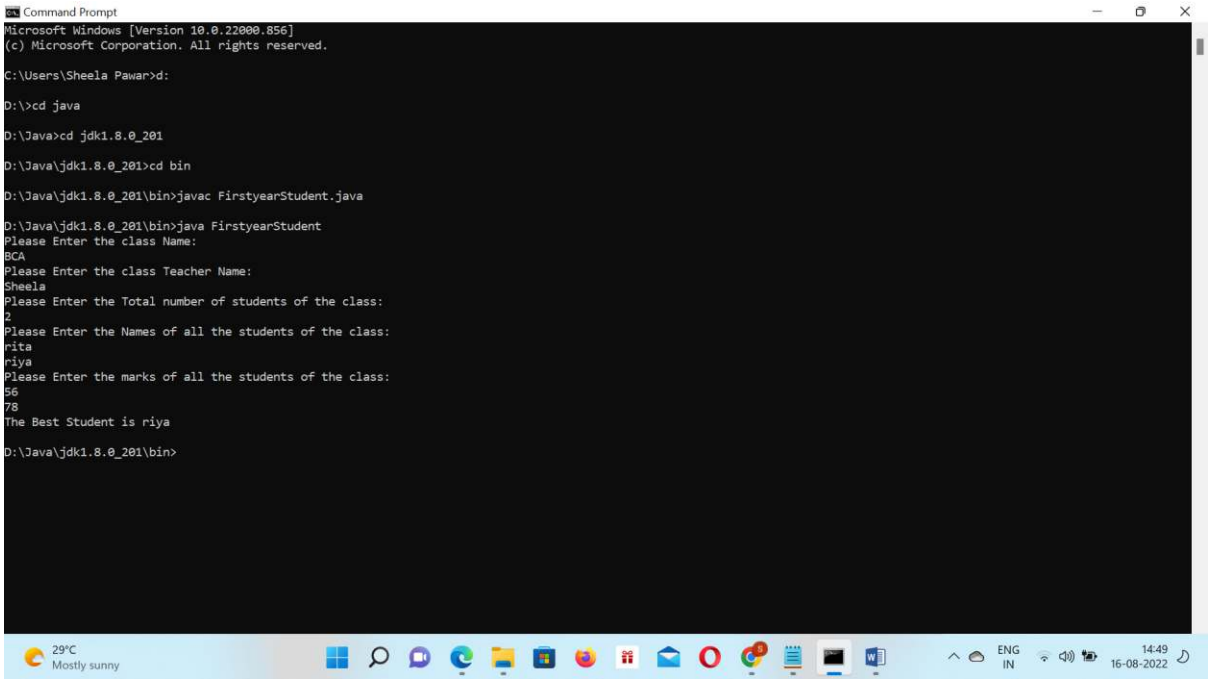


11. **In a college first year class are having the following attributesName of the class (BCA, BCom, BSc), Name of the staff No of the students in the class, Array of students in the class 10. Define a class called first year with above attributes and define a suitable constructor. Also write a method 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**

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

```
 class FirstYear{
String classname;
String classteacher;
int stdcount;
int stdmarks[]= new int[50];
String stdnames[] = new String[50];
Scanner sc = new Scanner(System.in);

public FirstYear() {
getinfo();
}

public void getinfo() {
System.out.println("Please Enter the class Name:");
classname = sc.nextLine();
System.out.println("Please Enter the class Teacher Name:");
classteacher = sc.nextLine();
System.out.println("Please Enter the Total number of students of the class:");
stdcount = Integer.parseInt(sc.nextLine());

System.out.println("Please Enter the Names of all the students of the class:");
for(int i=0;i<stdcount;i++)
stdnames[i]=sc.nextLine();

System.out.println("Please Enter the marks of all the students of the class:");
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 is "+stdnames[k]);
}
}
public class FirstyearStudent{
public static void main(String args[]){
FirstYear fy = new FirstYear();
fy.bestStudent();
}
}
```

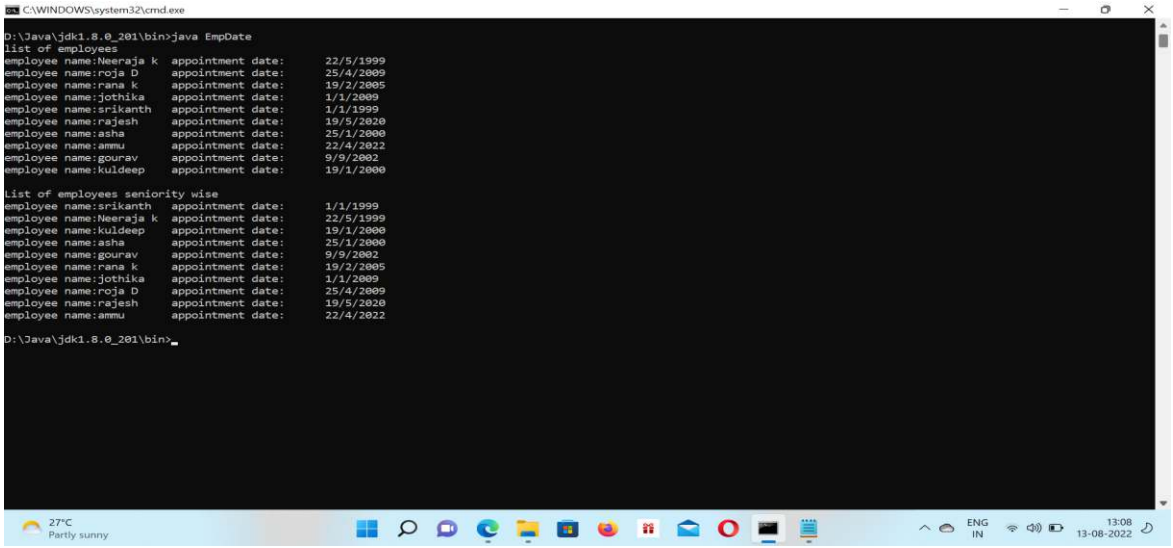


12. **Write a Java program to define a class called employee with the name and date of appointment. Create ten employee objects as an array and sort them as per their date of appointment. ie, print them as per their seniority.**

```
import java.util.Date;
class Employee {
    String name;
```

```
    Date appdate;
     public Employee(String nm, Date apdt){
name=nm;
appdate=apdt;
}
public void display(){
System.out.println("employee name:"+name+"\t appointment date:\t"+appdate.getDate()+
"/"+ appdate.getMonth()+"/"+appdate.getYear());
}}
public class EmpDate{
public static void main(String args[]){
Employee emp[]=new Employee[10];
emp[0]=new Employee("Neeraja k", new Date(1999,05,22));
emp[1]=new Employee("roja D",new Date(2009,04,25));
emp[2]=new Employee("rana k",new Date (2005,02,19));
emp[3]=new Employee("jothika", new Date(2009,01,01));
emp[4]=new Employee("srikanth",new Date(1999,01,01));
emp[5]=new Employee("rajesh",new Date(2020,05,19));
emp[6]=new Employee("asha",new Date(2000,01,25));
emp[7]=new Employee("ammu",new Date(2022,04,22));
emp[8]=new Employee("gourav",new Date(2002,9,9));
emp[9]=new Employee("kuldeep",new Date(2000,01,19));
System.out.println("list of employees");
for(int i=0;i<emp.length;i++)
emp[i].display();
for(int i=0;i<emp.length;i++){
for(int j=i+1;j<emp.length;j++){
if(emp[i].appdate.after(emp[j].appdate)){
Employee t= emp[i];
emp[i]=emp[j];
emp[j]=t;
```

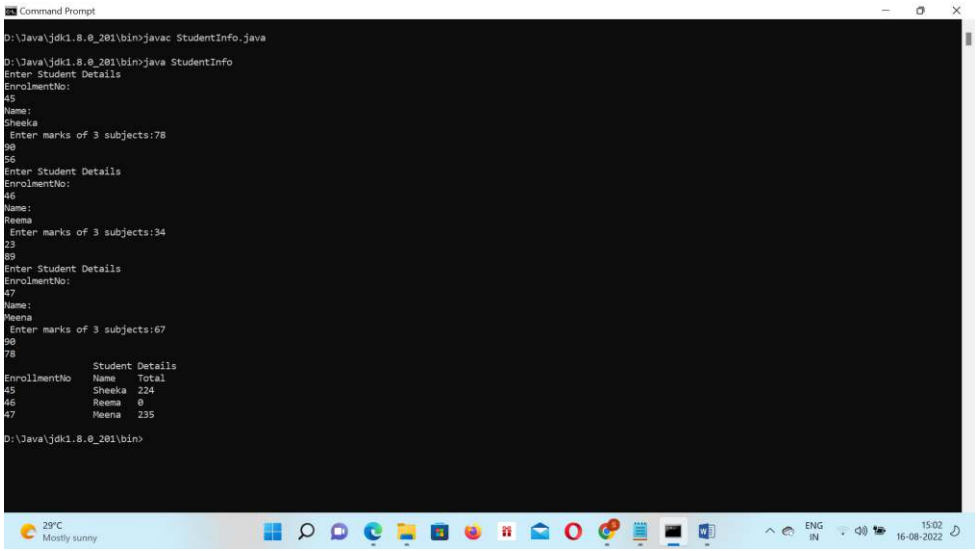
```
}}}
System.out.println("\nList of employees seniority wise");
for(int i=0;i<emp.length;i++)
emp[i].display();
}}
```



13. Write a java program to create a student class with following attributes: Enrollment_id: 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 declaredas 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.

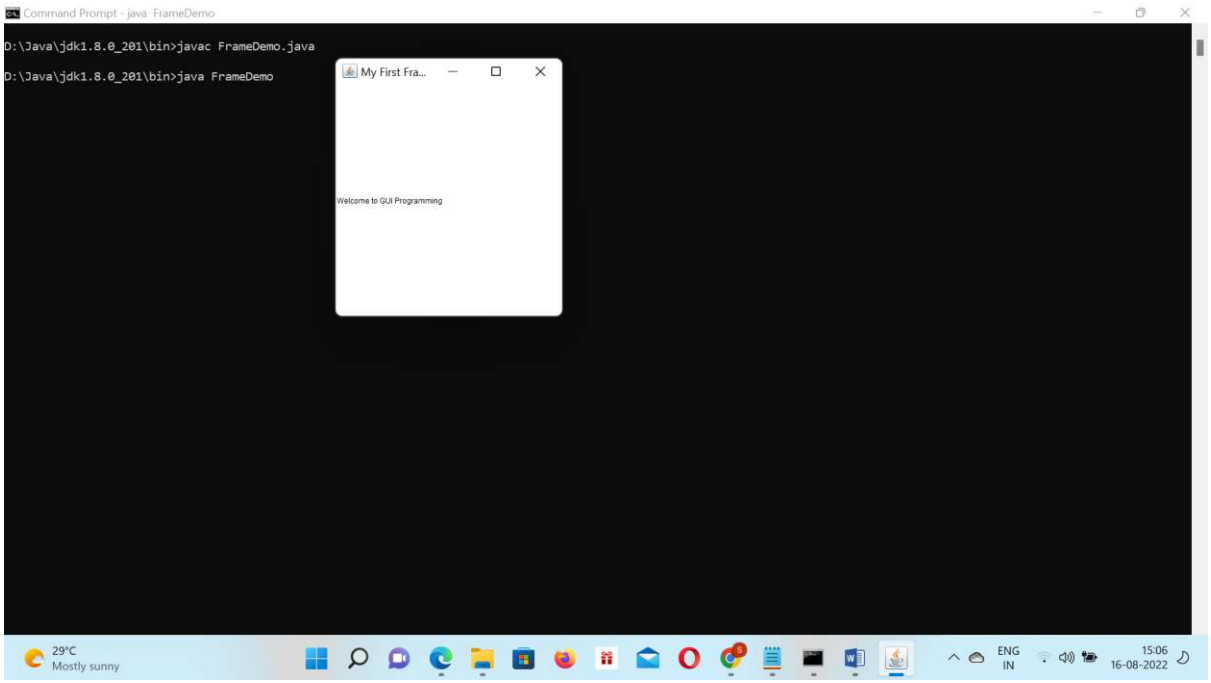
```
import java.util.*;

class Student {
  Scanner sc = new Scanner(System.in);
  String Enrollment_id;
  String Name;
  int sub1, sub2, sub3, total;

 Student(){
 readStudentInfo();
 }

 public void readStudentInfo(){
```

```
        System.out.println("Enter Student Details");
        System.out.println("EnrolmentNo: ");
        Enrollment_id = sc.next();
        System.out.println("Name: ");
        Name = sc.next();
        System.out.print(" Enter marks of 3 subjects:");
        sub1 = sc.nextInt();
        sub2 = sc.nextInt();
        sub3 = sc.nextInt();
        if (sub1 >= 50 && sub2 >= 50 && sub3 >= 50)
         total = sub1 + sub2 + sub3;
        else
         total = 0;
       }
       public void displayInfo() {
         System.out.println(Enrollment_id+"\t\t"+Name+"\t"+total);
        }
      }
      public class StudentInfo {
           public static void main(String[] args) {
       Student s[] = new Student[3];
        for (int i = 0; i <  3; i++) {s
            s[i] = new Student();
      }
       System.out.println("\t\tStudent Details");
       System.out.println("EnrollmentNo\tName\tTotal");
       for(int i = 0;i < 3; i++) {
          s[i].displayInfo();
       }
      }
}
```



14. Write a program which create and displays a message on the window

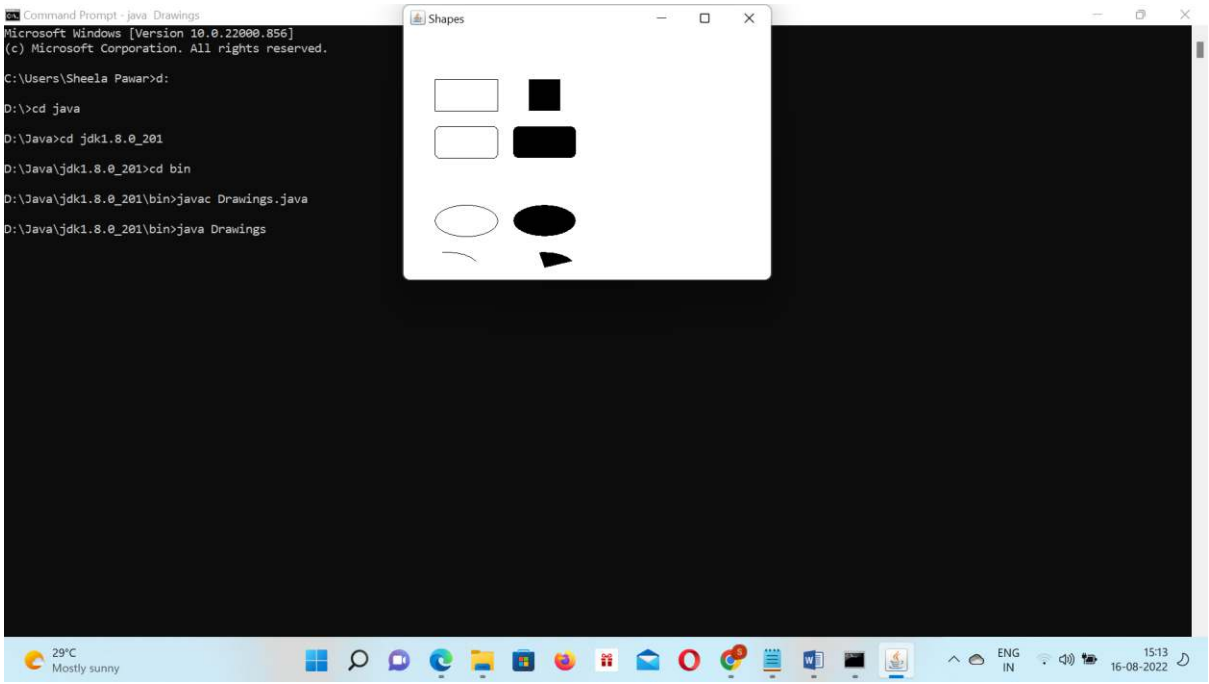
```
import java.awt.*;
public class FrameDemo{
FrameDemo(){
Frame fm = new Frame();
  fm.setTitle("My First Frame");
Label lb = new Label("Welcome to GUI Programming");
fm.add(lb);
fm.setSize(300,300);
fm.setVisible(true);
}

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



15. Write a program to draw several shapes in the created window

```
import java.awt.*;
public class Drawings extends Canvas {
    public void paint(Graphics g)  {
      g.drawRect(50,75,100,50);
      g.fillRect(200,75,50,50);
      g.drawRoundRect(50,150,100,50,15,15);
      g.fillRoundRect(175,150,100,50,15,15);
      g.drawOval(50,275,100,50);
      g.fillOval(175,275,100,50);
      g.drawArc(20,350,100,50,25,75);
```

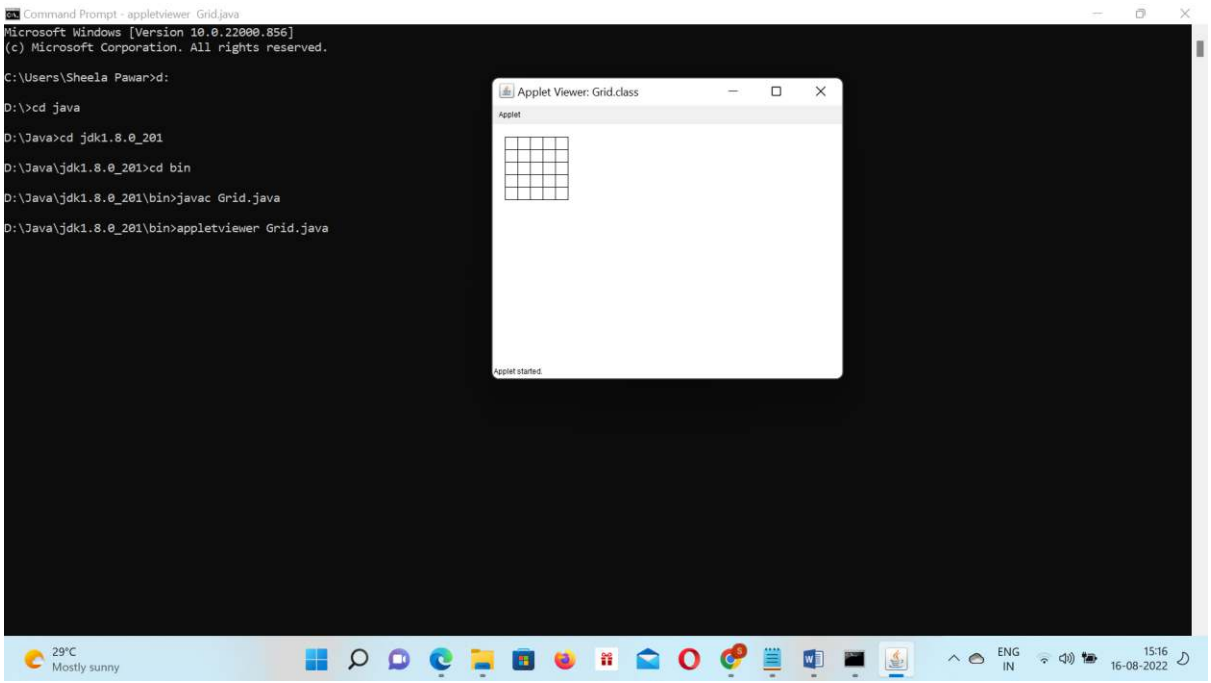
```
        g.fillArc(175,350,100,50,25,75);
       }

       public static void main(String args[]) {
        Drawings m=new Drawings();
        Frame f=new Frame("Shapes");
        f.add(m);
        f.setSize(300,450);
        f.setVisible(true);
       }
}
```



16. Write a program to create an applet and draw grid lines

```
import java.awt.*;
import java.applet.*;
public class Grid extends Applet {
  public void paint(Graphics g) {
    int row, column,x,y=20;
  //for every row
  for(row=1;row<5;row++) {
      x=20;
  //for ever column
  for(column=1;column<5;column++) {
      g.drawRect(x,y,40,40);
      x=x+20;
  }
```

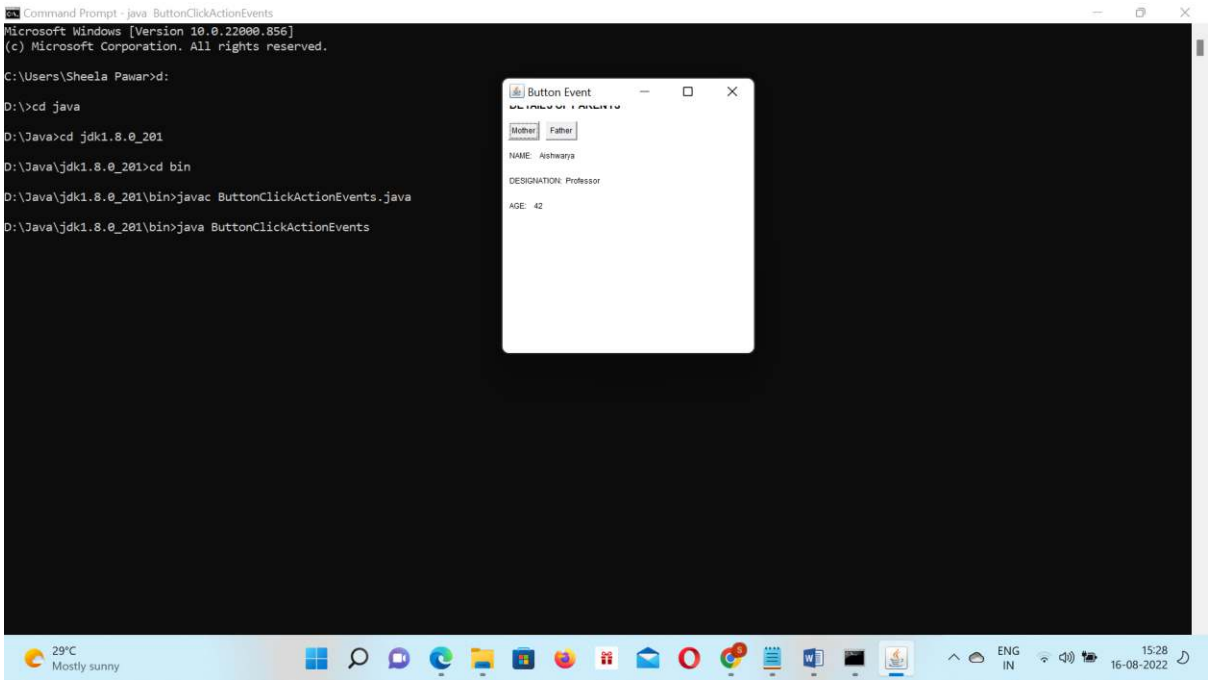
```
        y=y+20;
        }
       }
      }

      /*
       *<applet code="Grid.class" height =500 width =500> </applet>
*/
```



17. Write a 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.

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

public class ButtonClickActionEvents {

  public static void main(String args[]) {
    Frame f=new Frame("Button Event");
    Label l=new Label("DETAILS OF PARENTS");
    l.setFont(new Font("Calibri",Font.BOLD, 16));
    Label nl=new Label();
    Label dl=new Label();
    Label al=new Label();
    l.setBounds(20,20,500,50);
    nl.setBounds(20,110,500,30);
    dl.setBounds(20,150,500,30);
```

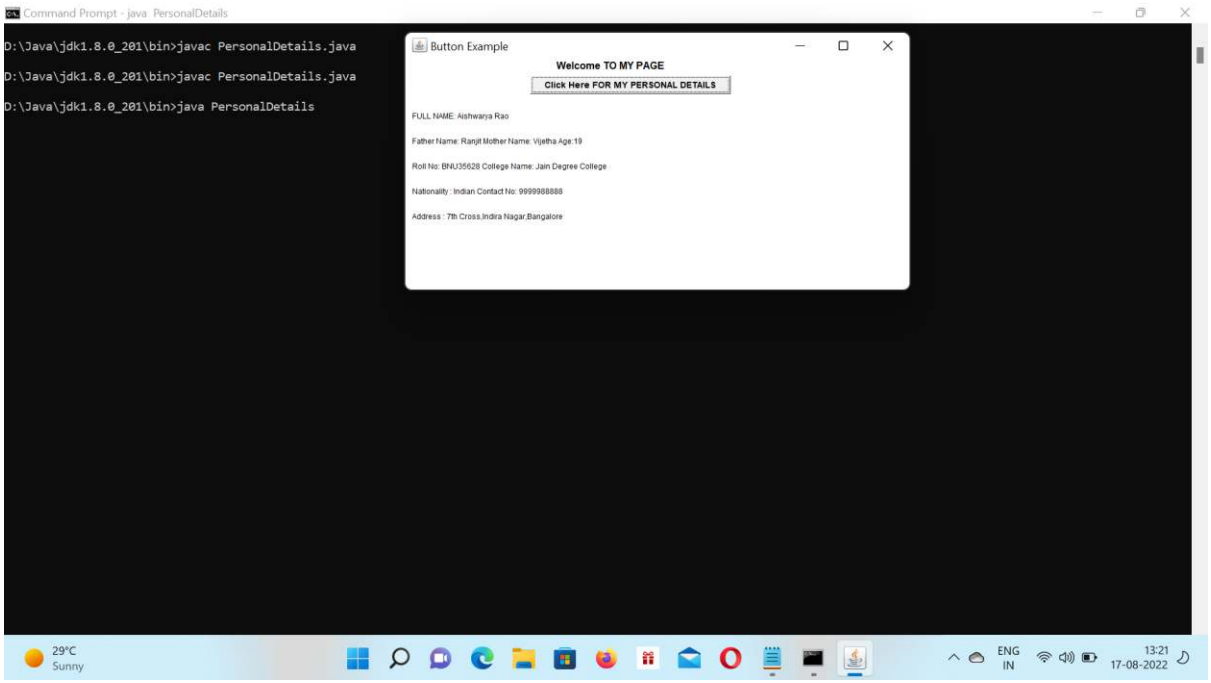
```
        al.setBounds(20,190,500,30);
        Button mb=new Button("Mother");
        mb.setBounds(20,70,50,30);
        mb.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                nl.setText("NAME:"+"    "+"Aishwarya");
                dl.setText("DESIGNATION:"+"  "+"Professor");
                al.setText("AGE:"+"    "+"42");
             }
        });
        Button fb=new Button("Father");
        fb.setBounds(80,70,50,30);
        fb.addActionListener(new ActionListener() {
           public void actionPerformed(ActionEvent e) {
             nl.setText("NAME:"+"   "+"Ram");
             dl.setText("DESIGNATION:"+"    "+"Manager");
             al.setText("AGE:"+"    "+"44");
              }
        });
        //adding elements to the frame
        f.add(mb);
        f.add(fb);
        f.add(l);
        f.add(nl);
        f.add(dl);
        f.add(al);
        // setting size,layout, and visibility
        f.setSize(250,250);
        f.setLayout(null);
        f.setVisible(true);
        }
}
```



18. Create a frame which displays your personal details with respect to a button click

```
import java.awt.*;
import java.awt.event.*;
 public class PersonalDetails
{
  public static void main(String args[])
   {
     Frame f = new Frame("Button Example");
     Label l = new Label("Welcome TO MY PAGE");
     l.setFont(new Font("Calibri",Font.BOLD,16));
      Label fnl= new Label();
      Label mnl = new Label();
      Label lnl = new Label();
      Label rl = new Label();
      Label al = new Label();
   l.setBounds(250,30,600,50);
   fnl.setBounds(20,120,600,30);
   mnl.setBounds(20,160,600,30);
   lnl.setBounds(20,200,600,30);
   rl.setBounds(20,240,600,30);
   al.setBounds(20,280,600,30);
Button mb = new Button ("Click Here FOR MY PERSONAL DETAILS");
mb.setFont(new Font ("Calibri",Font.BOLD,14));
mb.setBounds(210,70,320,30);
mb.addActionListener(new ActionListener(){
    public void actionPerformed(ActionEvent e){
      fnl.setText("FULL NAME: Aishwarya Rao");
```

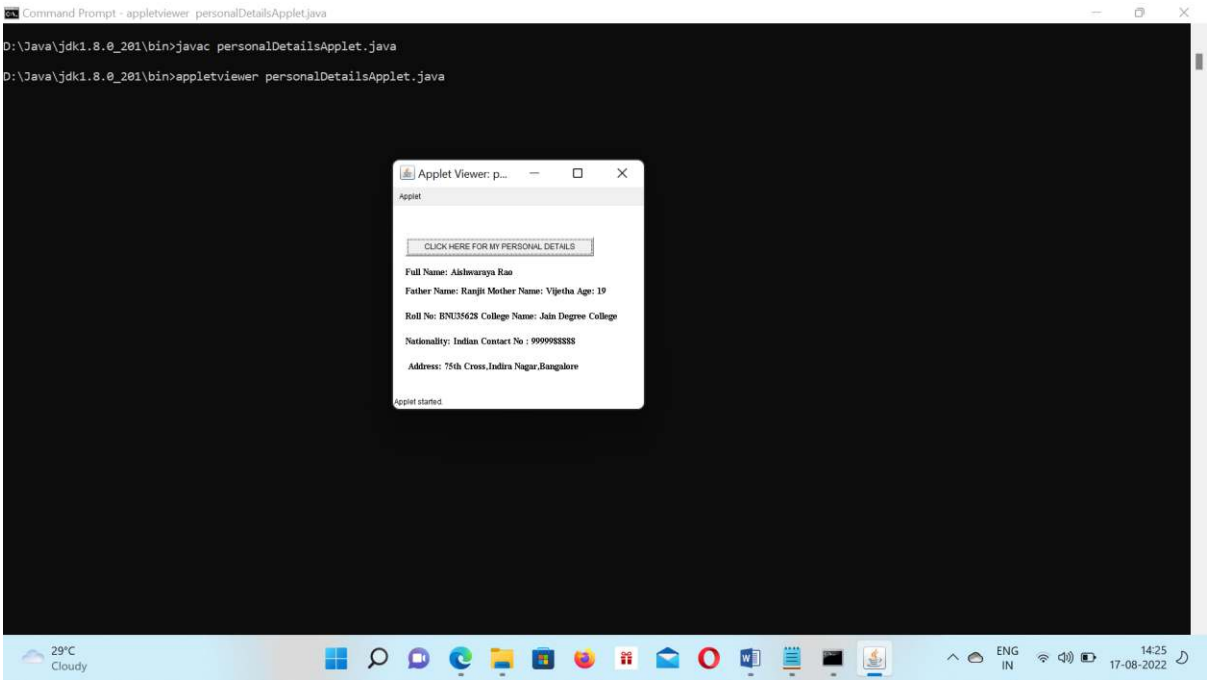
```
        mnl.setText("Father Name: Ranjit Mother Name: Vijetha Age:19");
        lnl.setText("Roll No: BNU35628 College Name: Jain Degree College");
        rl.setText("Nationality : Indian Contact No: 9999988888");
        al.setText("Address : 7th Cross,Indira Nagar,Bangalore");
    }
});
//adding elements to the frame
f.add(mb);
f.add(l);
f.add(fnl);
f.add(mnl);
f.add(lnl);
f.add(rl);
f.add(al);
f.setSize(400, 400);
f.setLayout(null);
f.setVisible(true);
 }
}
```



19. Create a simple applet which reveals the personal information of yours.

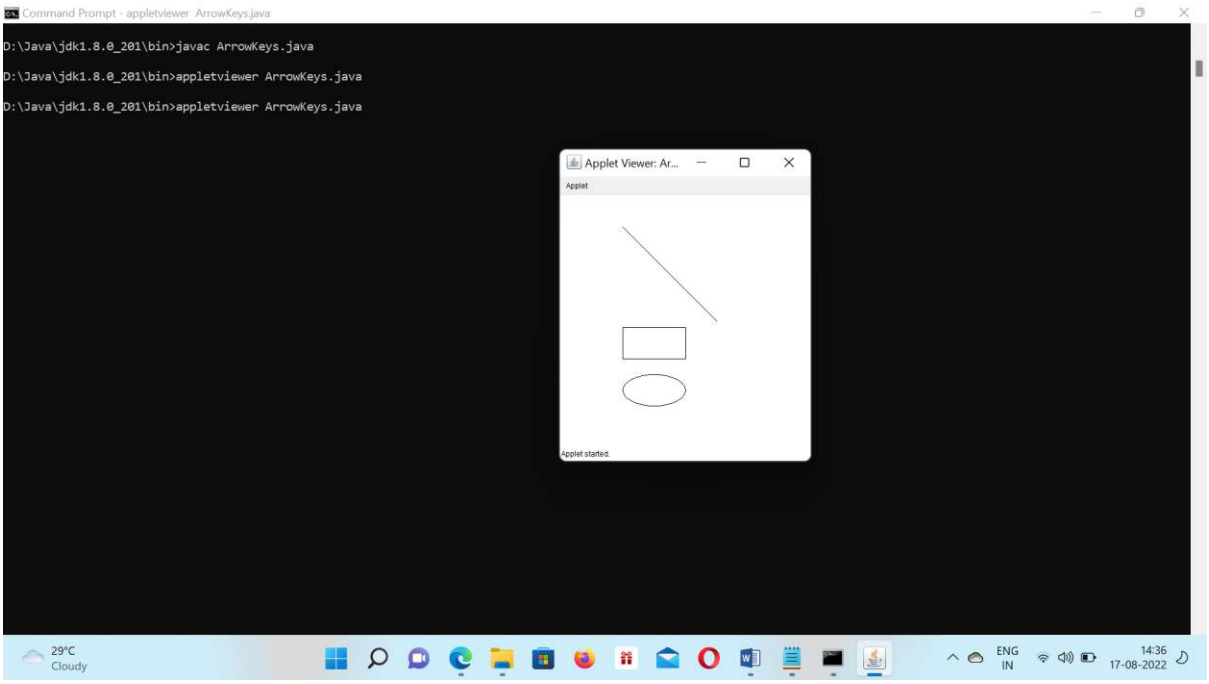
```
import java.awt.*;
import java.applet.*;
import java.awt.event.*;
public class personalDetailsApplet extends Applet implements ActionListener{
String s1= "   ", s2="   ", s3="  ", s4="  ", s5="  ";
```

```
public void init(){
setLayout(null);
setSize(400,300);
Button btn= new Button("CLICK HERE FOR MY PERSONAL DETAILS");
add (btn);
btn.setBounds(20,50,300,30);
btn.addActionListener(this);
}
public void actionPerformed(ActionEvent e){
s1="Full Name: Aishwaraya Rao";
s2="Father Name: Ranjit Mother Name: Vijetha Age: 19";
s3= "Roll No: BNU35628 College Name: Jain Degree College";
s4= "Nationality: Indian Contact No : 9999988888";
s5=" Address: 75th Cross,Indira Nagar,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);
}
}

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



20. Write a program to move different shapes according to the arrow key pressed.

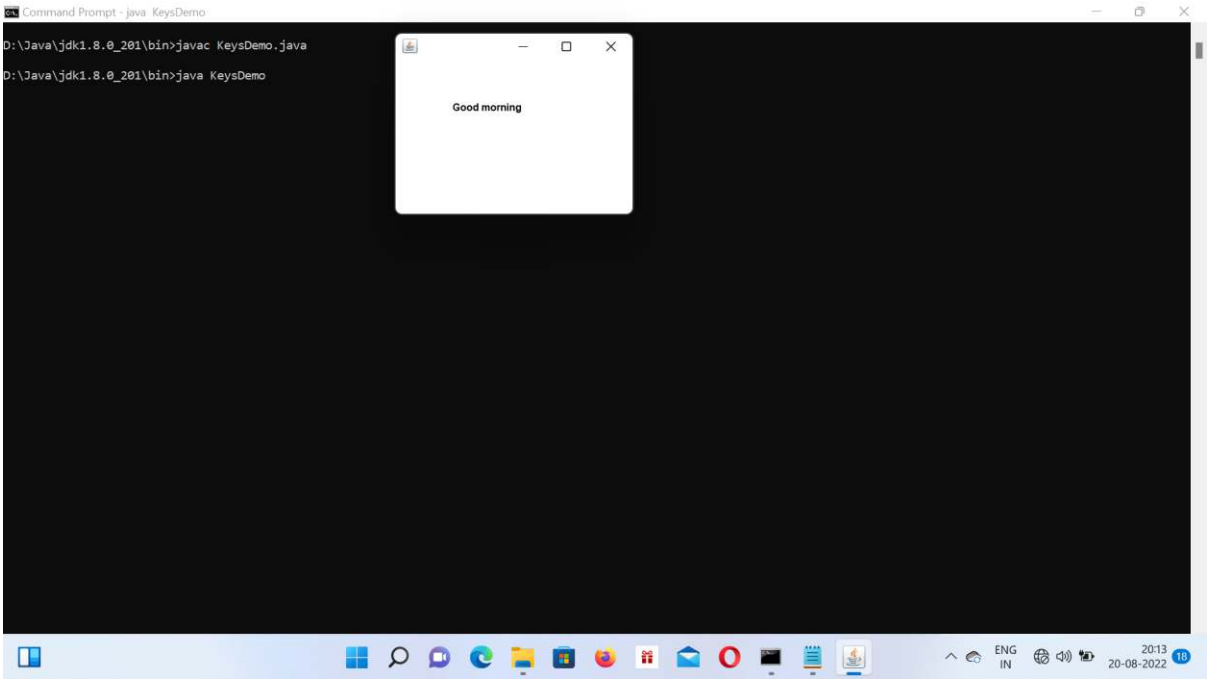
```
import java.applet.*;
import java.awt.*;
import java.awt.event.*;
/*
<applet code="ArrowKeys"Width=400 height=400>
</applet>
*/
public class ArrowKeys extends Applet implements KeyListener{

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);
}
}
```



21. Write a java 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

```
import java.awt.*;
import java.awt.event.*;
public class KeysDemo extends Frame implements KeyListener{
Label lbl;
KeysDemo() {
addKeyListener(this);
requestFocus();
lbl=new Label();
lbl.setBounds(100,100,200,40);
lbl.setFont(new Font("Calibri",Font.BOLD,16));
add(lbl);
setSize(400,300);
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() == 'N' || e.getKeyChar() =='n')
    lbl.setText("Good night");
else  if(e.getKeyChar() == 'E' || e.getKeyChar() == 'e')
    lbl.setText("good evening");
```

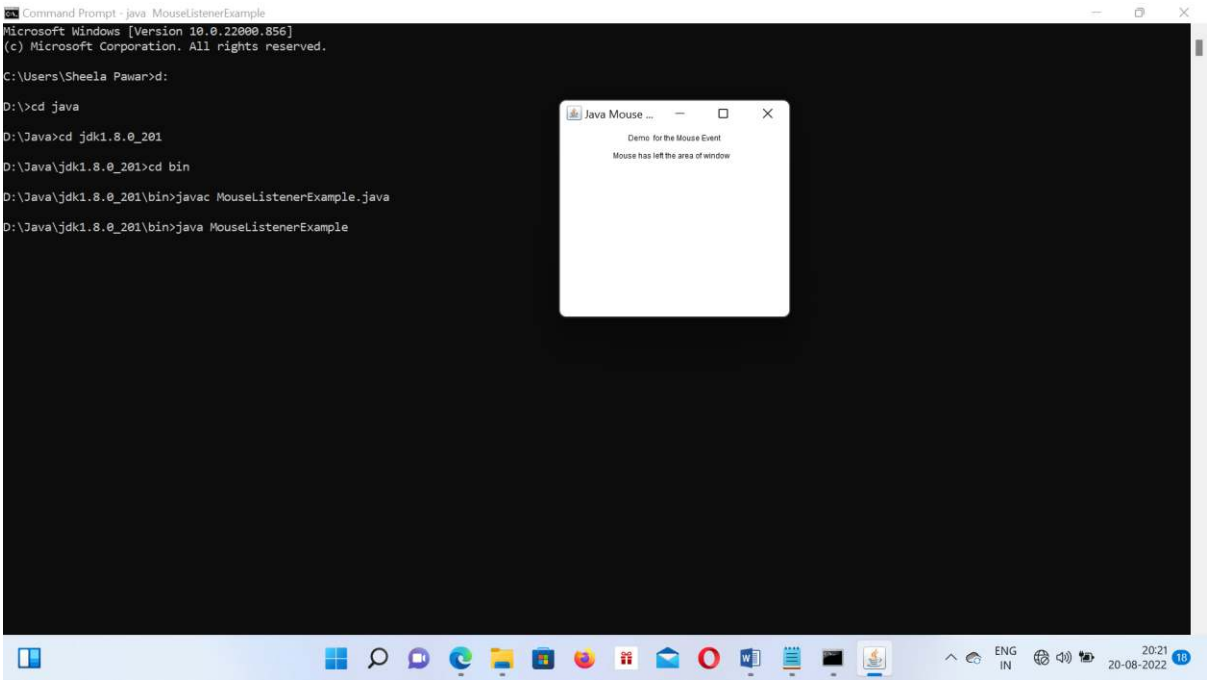
```
}
public void keyReleased(KeyEvent e){
}
public void keyTyped (KeyEvent e){
}
public static void main(String[] args){
    new KeysDemo();
}}
```



22. Demonstrate the various mouse handling events using suitable example

```
import java.awt.*;
import java.awt.event.MouseEvent;
import java.awt.event.MouseListener;
public class MouseListenerExample implements MouseListener{
//create two labels lbl1 and lbl2
Label lbl1, lbl2;
//create a frame frame
Frame fr;
//create a string s
String s;
MouseListenerExample(){
    fr = new Frame("Java Mouse Listener Example");
    lbl1 = new Label("Demo  for the Mouse Event",Label.CENTER);
    lbl2 = new Label();
    //set the layout of frame as Flowlayout
    fr.setLayout(new FlowLayout());
    //add label 1 to frame
    fr.add(lbl1);
```

```
        //add label 2 to frame
        fr.add(lbl2);
        //Register the created class MouseListenerExample with MouseListener
        fr.addMouseListener(this);
        //set the size of the where width is 250 and height iss 250
        fr.setSize(250,250);
        //set the visibility of frame as true
        fr.setVisible(true);
    }
    //implementation of mouseClicked method
    public void mouseClicked(MouseEvent ev){
        lbl2.setText("Mouse Button Clicked");
        fr.setVisible(true);
    }
    //implementation of mouseEntered method
    public void mouseEntered(MouseEvent ev){
        lbl2.setText("Mouse has entered the area of window");
        fr.setVisible(true);
    }
    //implementation of mouseExited method
    public void mouseExited(MouseEvent ev){
        lbl2.setText("Mouse has left the area of window");
        fr.setVisible(true);
    }
    //implementation of mousePressed method
    public void mousePressed(MouseEvent ev){
        lbl2.setText("Mouse button is being pressed");
        fr.setVisible(true);
    }
    //implementation of mouseReleased method
    public void mouseReleased(MouseEvent ev){
        lbl2.setText("Mouse released");
        fr.setVisible(true);
    }
    //main method
    public static void main(String args[]){
        new MouseListenerExample();
    }
    }
```



23. Write a program to create menu bar and pull-down menus.

```
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("Copy");
    MenuItem b2 = new MenuItem("Find");
    MenuItem c1 = new MenuItem("Show");
    fileMenu.add(a1);
    fileMenu.add(a2);
    fileMenu.add(a3);
    editMenu.add(b1);
    editMenu.add(b2);
    viewMenu.add(c1);
    fr.setMenuBar(mb);
    fr.setSize(300,300);
    fr.setLayout(null);
    fr.setVisible(true);
    }
```

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