Slips Solution

1) Write a java program for a cricket player The program should accept the details from user (max 10): Player code, name, runs, innings- played and number of times not out. The program should contain following menu:

-Enter details of players.

-Display average runs of a single player.

-Display average runs of all players. (Use array of objects & function overloading)

Solution:-

```
import java.io.*;

class Player
{
    String Name;
    int TotalRuns, TimesNotOut, InningsPlayed,pcode;
    float Avg;
    static BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
    void getData()
    {
        try
        {
            System.out.println("Enter Player Code: ");
            pcode=Integer.parseInt(br.readLine());
            System.out.println("Enter Player Name: ");
            Name = br.readLine();
            System.out.println("Enter Total Runs: ");
            TotalRuns = Integer.parseInt(br.readLine());
            System.out.println("Enter Times Not Out: ");
            TimesNotOut = Integer.parseInt(br.readLine());
            System.out.println("Enter Innings Played: ");
            InningsPlayed = Integer.parseInt(br.readLine());
        }
        catch(Exception e)
        {
            System.out.println(e);
        }
    }

    void putData()
    {
```

```
            System.out.println(pcode + "\t"+
Name+"\t"+TotalRuns+"\t"+TimesNotOut+"\t"+InningsPlayed+"\t"+Avg);
      }

      void getAvg()
      {
       Avg= TotalRuns / (InningsPlayed - TimesNotOut);

      }
          static void getAvg(Player p[],int n)
          {
        for (int i=0;i<n;i++)
            {
           p[i].Avg=p[i].TotalRuns / (p[i].InningsPlayed - p[i].TimesNotOut);
        }
        for (int i=0;i<n;i++)
            {
           p[i].putData();
        }

         }
}

public class Ass1
{
      static BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
      public static void main(String args[])
      {
            try
            {
                  System.out.println("Enter No.of Players: ");
                  int n = Integer.parseInt(br.readLine());
                  Player p[] = new Player[n];
                  for(int i=0; i<n; i++)
                  {
                        p[i] = new Player();
                        p[i].getData();
                  }

                        System.out.println("Player Code For Avg Calculation");
                        int m=Integer.parseInt(br.readLine());
                  for(int i=0; i<n; i++)
                        {
                          if(p[i].pcode==m)
```

```
                {
                        p[i].getAvg();
                        p[i].putData();
                    }

                }
             System.out.println("Average Of All The Players");
                 Player.getAvg(p,n);
                 }
            catch(Exception e)
            {
                System.out.println(e);
            }
        }


}

)
```

2) Create a class telephone containing name, telephone number & city and write necessary member functions for the following:
   - Search the telephone number with given name.
   - Search the name with given telephone number.
   - Search all customers in a given city. (Use function overloading)

Solution:-

```
import java.io.*;
class Tel
{
 int pno;
 String city,cname;
 static BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
 void getData()
 {
            try
          {
                    System.out.println("Enter Telephone Code: ");
                    pno=Integer.parseInt(br.readLine());
              System.out.println("Enter Customer Name: ");
              cname = br.readLine();
              System.out.println("Enter The City: ");
              city=br.readLine();

          }
          catch(Exception e)
          {
               System.out.println(e);
          }
 }
 void putData()
 {
    System.out.println(pno + "\t"+ cname+"\t"+city);
 }
 static void search(Tel t[],int no,int n)
 {
   for(int i=0;i<n;i++)
   {
     if(t[i].pno==no)
     {
        System.out.println();
        System.out.println("Name Of Customer:- "+ "\t" + t[i].cname);
      }
```

```
    }

  }
 static void search(Tel t[],String cn,int n)
  {
    for(int i=0;i<n;i++)
    {
      if((t[i].cname).compareTo(cn)==0)
      {
         System.out.println();
         System.out.println("TelePhone No:- "+ "\t" + t[i].pno);
       }
    }

  }
 static void search(Tel t[],String cn,int n,String ct)
  {
    for(int i=0;i<n;i++)
    {
      if((t[i].city).compareTo(ct)==0)
      {
         System.out.println();

         System.out.println("Customer Name:- "+ "\t" + t[i].cname);

       }
    }

  }



}
 class Ass2
 {
      static BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
      public static void main(String args[])
      {
           try
           {
                System.out.println("Enter No.of Records: ");
                int n = Integer.parseInt(br.readLine());
                Tel t[] = new Tel[n];
                for(int i=0; i<n; i++)
```

```
                {
                    t[i] = new Tel();
                    t[i].getData();
                }

                    System.out.println("Telephone No");
                    int m=Integer.parseInt(br.readLine());
                Tel.search(t,m,n);

                    System.out.println();
                    System.out.println("Enter the Customer Name");
                    String cn=br.readLine();
                    Tel.search(t,cn,n);


                    System.out.println();
                    System.out.println("Enter the City");
                    String ct=br.readLine();
                    Tel.search(t,cn,n,ct);

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

3) Write a java program which creates class Student (Rollno, Name,- Number of subjects, Marks of each subject)(Number of subjects varies for each student)
Write a parameterized constructor which initializes roll no, name & Number of subjects and create the array of marks dynamically. Display the details of all students with percentage and class obtained.

Solution :-

```
import java.io.*;
class Student
{
 static BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
 int rno,nofs,tot,per;

 String sname,grade;
 Student(int r,String s,int cnt)
 {
   rno=r;
   sname=s;
   nofs=cnt;
 }
 static void getMark(Student s[],int n) throws Exception
 {
    for(int i=0;i<n;i++)
    {
              System.out.println("Subject of Student :" + (i+1));

              int mks[]=new int[s[i].nofs];
              for(int j=0;j<s[i].nofs;j++)
              {

            System.out.println("Marks");
                      mks[j]=Integer.parseInt(br.readLine());
              s[i].tot=s[i].tot+mks[j];
              }
            s[i].per=s[i].tot/s[i].nofs;
            if(s[i].per>=70)
            {
              s[i].grade="Dist";
            }
               else if(s[i].per>=60 && s[i].per <70)
              s[i].grade="A";
            else if (s[i].per >=50 && s[i].per <60)
              s[i].grade="B";
            else if (s[i].per >=40 && s[i].per < 50)
```

```
                s[i].grade="C";
            else
                s[i].grade="Fail";

        }
        System.out.println("Rno" + "\t" + "Sname" + "\t" + "No Of Sub" + "\t" + "Total" + "\t"
+"Percentage" + "\t" +"Grade");
         for(int i=0;i<n;i++)
         {
                s[i].disp();
         }

  }
    void disp()
   {

     System.out.println( rno + "\t" + sname + "\t" +     nofs + "\t\t" +        tot + "\t\t" +    per +
"\t" +       grade);
   }


}
class Ass3
{
  public static void main(String args[]) throws Exception
  {
     BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
     System.out.println("How Many Records");
     int n=Integer.parseInt(br.readLine());
     Student s[]=new Student[n];
     for(int i=0;i<n;i++)
     {
       System.out.println("Enter the RollNo");
       int r=Integer.parseInt(br.readLine());
       System.out.println("Enter the Student Name");
       String sn=br.readLine();
       System.out.println("How Many Subject");
       int cn=Integer.parseInt(br.readLine());
       s[i]=new Student(r,sn,cn);
     }
     System.out.println("Student Details");
     Student.getMark(s,n);
   }
}
```

4) In a bank, different customers having saving account. Some customers may have taken a loan from the bank. So bank always maintains information about bank depositors and borrowers. Design a Base class Customer (name, phone-number).Derive a class Depositor (accno , balance) from Customer. Again derive a class Borrower (loan-no, loan-amt) from Depositor. Write necessary member functions to read and display the details of 'n' customers.

Solution:-

```
import java.io.*;
class Customer
{
 String cname;
 int pno;
 Customer(int p,String s)
 {
   pno=p;
   cname=s;
 }
}
class Depositor extends Customer
{
  int acno,bal;
  Depositor(int p,String s,int a,int b)
  {
    super(p,s);
    acno=a;
    bal=b;
  }
}
class Borrower extends Depositor
{
      int loanNo,loanAmt;
      Borrower(int p,String s,int a,int b,int ln,int la)
      {
                super(p,s,a,b);
                loanNo=ln;
                loanAmt=la;
      }
      void putData()
      {
                System.out.println(pno+"\t"+cname+"\t"+acno +
"\t"+bal+"\t"+loanNo+"\t"+loanAmt);
```

```
    }

    static void  Depo(Borrower b[],int n)
    {
            for(int i=0;i<n;i++)
            {
                    if(b[i].bal>100)
                {
                    System.out.println(b[i].cname);
                }
            }
    }

    static void  Borrow(Borrower b[],int n)
    {
            for(int i=0;i<n;i++)
            {
                    if(b[i].loanAmt>100)
                {
                    System.out.println(b[i].cname);
                }
            }
    }

}
class Ass4
{
    public static void main(String args[]) throws Exception
    {
            int n;
            BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
            System.out.println("Enter the No Of Records");
            n=Integer.parseInt(br.readLine());
            Borrower b[]=new Borrower[n];
            for(int i=0;i<n;i++)
            {
                    System.out.println("Enter the Ph_No");
                    int p=Integer.parseInt(br.readLine());
                System.out.println("Enter the Customer Name");
                    String cn=br.readLine();
                System.out.println("Enter the Ac_No");
                int an=Integer.parseInt(br.readLine());
                System.out.println("Enter the Amt For Deposite");
                int bal=Integer.parseInt(br.readLine());
```

```
                    System.out.println("Enter the Loan_No");
                    int loanNo=Integer.parseInt(br.readLine());
                    System.out.println("Enter the loanAmt");
                    int loanAmt=Integer.parseInt(br.readLine());
                    b[i]=new Borrower(p,cn,an,bal,loanNo,loanAmt);
                }
    System.out.println("List Of Depositor");
    Borrower.Depo(b,n);

    System.out.println("List Of Borrower");
    Borrower.Borrow(b,n);

  }
}
```

5) Write a program in java to check whether the entered character is alphabet, digit or space character. If it is alphabet then print whether it is capital or small alphabet. Also change the alphabet into the reverse case

Solution:-

```
import java.io.*;
class Ass5
{
  public static void main(String args[]) throws Exception
  {
    BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
    String str;
    char ch,c;
    System.out.println("Enter the Character");
    str=br.readLine();
    ch=str.charAt(0);
    if(Character.isSpaceChar(ch))
    {
      System.out.println("Entered Character Is Space");
    }
    if(Character.isDigit(ch))
    {
       System.out.println("Entered Character is Digit");
    }
    if(Character.isLetter(ch))
    {
       System.out.println("Entered Character is Alphabet");
      if(Character.isLowerCase(ch))
       {
          c=Character.toUpperCase(ch);
          System.out.println("After Changing the Case " + c);
       }
       else
      {
         c=Character.toLowerCase(ch);
         System.out.println("After Changing the Case " + c);
      }
     }
  }
}
```

6) Using Class implement Hash table to

1) Accept ‘n’ records of students (Name, Percentage)
2) Display details of all students

Find out highest marks

Solution:-

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

class Ass6
{
    static BufferedReader br=new BufferedReader (new InputStreamReader(System.in));

    public static void main(String args[])
    {
            try
            {
                    System.out.print("Enter no of Records:");
                    int n=Integer.parseInt(br.readLine());

                    Hashtable hashtable=new Hashtable();

                    for(int i=1; i<=n; i++)
                    {
                            System.out.println("Enter the name of Student :"+i+":");
                            String name=br.readLine();

                            System.out.println("Enter perct of Student:"+ i +":");
                            float percentage=Float.valueOf(br.readLine()).floatValue();
                            hashtable.put(name,percentage);
                    }

                    System.out.println("Name \t Percentage");
                    Enumeration keys=hashtable.keys();
                    float max=0;

                    while(keys.hasMoreElements())
                    {
                            String key=(String) keys.nextElement();
                            float
value=Float.valueOf(hashtable.get(key).toString()).floatValue();
```

```
                System.out.println(key+"\t" + value);

                if(max<value)

                        max=value;
            }

            System.out.println("Maximum percentage=" +max);
        }
        catch(Exception e)
        {
            System.out.println(e);
        }
}
}
```

7) Write a Java Program to accept the details of Employee from the user and display it on the next Frame. (Use AWT)

Solution:-

Frame 1:-

```
import java.awt.*;
import java.awt.event.*;
import java.io.*;
class emp extends Frame implements ActionListener
{
    Label l1,l2,l3,l4,l5,l6;
    TextField txt1,txt2,txt3,txt4,txt5,txt6;
    Button next,cancel;
    Panel p1,p2,p3;
    Label ll;
    emp()
    {
            ll=new Label("EMPLOYEE INFORMTION");

            l1=new Label("Emp No. : -");
            l2=new Label("Name : -");
            l3=new Label("Address : -");
            l4=new Label("Phone/Mobile No. : -");
            l5=new Label("Designation : -");
            l6=new Label("Salary : -");

                txt1=new TextField(20);
                txt2=new TextField(20);
                txt3=new TextField(20);
                txt4=new TextField(20);
                txt5=new TextField(20);
                txt6=new TextField(20);

            next=new Button("NEXT");    next.addActionListener(this);
            cancel=new Button("CANCEL");   cancel.addActionListener(this);


            p1=new Panel();
            p1.setLayout(new GridLayout(6,2));
            p1.add(l1);
                p1.add(txt1);
            p1.add(l2);
```

```
                p1.add(txt2);
        p1.add(l3);
                p1.add(txt3);
        p1.add(l4);
                p1.add(txt4);
        p1.add(l5);
                p1.add(txt5);
        p1.add(l6);
                p1.add(txt6);

        p2=new Panel();
        p2.setLayout(new GridLayout(1,2));
        p2.add(next);
        p2.add(cancel);

        p3=new Panel();
        p3.add(ll);

        //Panel Arrangment
        add(p3,BorderLayout.NORTH);
        add(p1,BorderLayout.CENTER);
        add(p2,BorderLayout.SOUTH);

        setVisible(true);
        setSize(400,400);


    }
    public void actionPerformed(ActionEvent ae)
    {
        if(ae.getSource()==next)
        {

                new
empdet(txt1.getText(),txt2.getText(),txt3.getText(),txt4.getText(),txt5.getText(),txt6.getText
());
        }
        if(ae.getSource()==cancel)
        {
                System.exit(0);
        }
    }

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

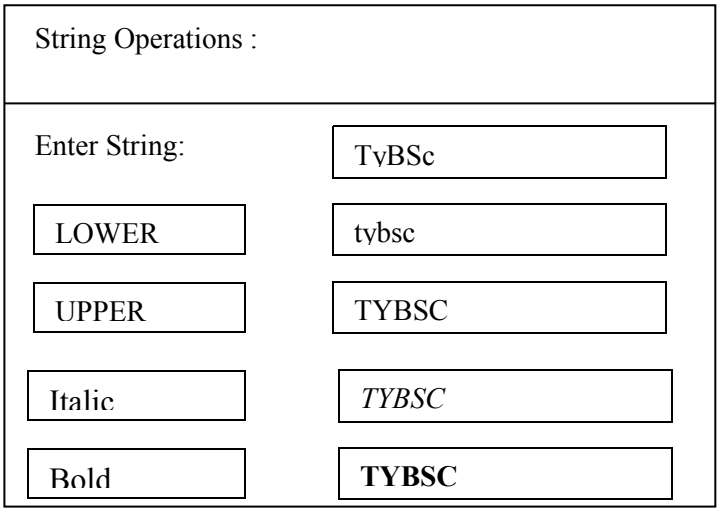
```
 {
     new emp().show();
 }
}
```

Frame 2:-

```
import java.awt.*;
import java.awt.event.*;
import java.io.*;
class empdet extends Frame implements ActionListener
{
     Label l1,l2,l3,l4,l5,l6;
     Label lt1,lt2,lt3,lt4,lt5,lt6;
      Button cancel;
     Panel p1=new Panel();
     Panel p2=new Panel();
     Panel p3=new Panel();
     Label ll;
     empdet(String b,String c,String d,String e,String f,String g)
     {
               ll=new Label("EMPLOYEE INFORMTION");

               l1=new Label("Emp No. : -");
               l2=new Label("Name : -");
               l3=new Label("Address : -");
               l4=new Label("Phone/Mobile No. : -");
               l5=new Label("Designation : -");
               l6=new Label("Salary : -");

               lt1=new Label(b);
               lt2=new Label(c);
               lt3=new Label(d);
               lt4=new Label(e);
               lt5=new Label(f);
               lt6=new Label(g);

               cancel=new Button("CLOSE");          cancel.addActionListener(this);


               p1.setLayout(new GridLayout(6,2));
               p1.add(l1);
               p1.add(lt1);
```

```
            p1.add(l2);
            p1.add(lt2);
            p1.add(l3);
            p1.add(lt3);
            p1.add(l4);
            p1.add(lt4);
            p1.add(l5);
            p1.add(lt5);
            p1.add(l6);
                    p1.add(lt6);

            p2.setLayout(new GridLayout(1,1));

            p2.add(cancel);

            p3.add(ll);

            add(p3,BorderLayout.NORTH);
            add(p1,BorderLayout.CENTER);
            add(p2,BorderLayout.SOUTH);

            setVisible(true);
            setSize(400,400);


    }
    public void actionPerformed(ActionEvent ae)
    {

            if(ae.getSource()==cancel)
            {
                    System.exit(0);
            }
    }


}
```

8) Write a JAVA program to design a screen using Awt :



Solution:-

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

public class Ass8 extends Frame implements ActionListener
{
    Button lowerBtn, upperBtn, boldBtn, italicBtn;
    TextField inputText;
    Label inputLabel,lowerLabel, upperLabel, boldLabel, italicLabel;
    public Ass8()
    {
        lowerBtn = new Button("Lower");
        upperBtn = new Button("Upper");
        boldBtn = new Button("Bold");
        italicBtn = new Button("Italic");

        inputText = new TextField(20);

        inputLabel = new Label("Enter String:",Label.RIGHT);
        lowerLabel = new Label("",Label.LEFT);
        upperLabel = new Label("",Label.LEFT);
        boldLabel = new Label("",Label.LEFT);
        italicLabel = new Label("",Label.LEFT);

        setLayout(new GridLayout(5,2));
```

```
        add(inputLabel);
        add(inputText);
        add(lowerBtn);
        add(lowerLabel);
        add(upperBtn);
        add(upperLabel);
        add(boldBtn);
        add(boldLabel);
        add(italicBtn);
        add(italicLabel);

        lowerBtn.addActionListener(this);
        upperBtn.addActionListener(this);
        boldBtn.addActionListener(this);
        italicBtn.addActionListener(this);

        addWindowListener(new WindowAdapter()
        {
           public void windowClosing(WindowEvent e)
           {
              System.exit(0);
           }
        });

        setTitle("String Operations");
        setSize(200,200);
        setVisible(true);
    }

    public void actionPerformed(ActionEvent ae)
    {
        Button btn = (Button)ae.getSource();
        String txt = inputText.getText();
        if(btn == lowerBtn)
        {
            lowerLabel.setText(txt.toLowerCase());
        }
        if(btn == upperBtn)
        {
            upperLabel.setText(txt.toUpperCase());
        }
        if(btn == boldBtn)
        {
            Font f = new Font("Arial",Font.BOLD,15);
```

```
            boldLabel.setFont(f);
            boldLabel.setText(txt);
        }
        if(btn == italicBtn)
        {
            Font f = new Font("Arial",Font.ITALIC,15);
            italicLabel.setFont(f);
            italicLabel.setText(txt);
        }
    }

    public static void main(String args[])
    {
        new Ass8();
    }
}
```

9) Define an Employee class with suitable attributes having getSalary() method, which returns salary withdrawn by a particular employee. Write a class Manager which extends a class Employee, override the getSalary() method, which will return salary of manager by adding traveling allowance, house rent allowance etc.

Solution:-

```
class Emp
{
 int eno;
 float bsal,gs,tax,pf,ns,ded,da;
 String ename;
 Emp(int e,String en,float s)
 {
  eno=e;
  ename=en;
  bsal=s;
 }
 float getSalary()
 {
  da=bsal*0.10f;
  gs=bsal+da;
  tax=bsal*0.05f;
  pf=bsal*0.04f;
  ded=tax+pf;
  ns=gs-ded;
  return (ns);
 }
 void disp()
 {
  System.out.println(eno + " "+ename + " " +ns);
 }
}
class Manager extends Emp
{
 float hra,ta,mns;
 Manager(int e,String en,float s,float hr,float t)
 {
   super(e,en,s);
   hra=hr;
   ta=t;
 }
 float getSalary()
 {
```

```
   float f=super.getSalary();
   mns=f+hra+ta;
   return(mns);
  }
}
class Ass9
{
  public static void main(String args[])
  {
   Manager mn=new Manager(1,"satyavan",12500.0f,8200.0f,7420.0f);
   float som=mn.getSalary();
   System.out.println("Salary Of manager Is \t" + som);
  }
}
```

10) Write a Java program to create 2 files F1 and F2.Copy the contents of file F1 by changing the case into file F2.

F2 = F1

Also copy the contents of F1 and F2 in F3.

F3 = F1+F2

Display the contents of F3.

Use Command Line Arguments.

Solution:-

```
import java.io.*;

class Ass10
{
   public static void main(String args[])
   {
      try
      {
         FileInputStream f1,f2;
         FileOutputStream f3;
         int c;
         switch(args.length)
         {
         case 2:
            f1 = new FileInputStream(args[0]);
            f3 = new FileOutputStream(args[1]);
            while((c=f1.read())!=-1)
            {
               if(Character.isUpperCase((char)c))
                  c = Character.toLowerCase((char)c);
               if(Character.isLowerCase((char)c))
                  c = Character.toUpperCase((char)c);
               f3.write((char)c);
            }
            f1.close();
            f3.close();
            break;
         case 3:
            f1 = new FileInputStream(args[0]);
            f2 = new FileInputStream(args[1]);
            f3 = new FileOutputStream(args[2]);
            while((c=f1.read())!=-1)
            {
               f3.write((char)c);
```

```
            }
            while((c=f2.read())!=-1)
            {
               f3.write((char)c);
            }
            f1.close();
            f2.close();
            f3.close();
            f1 = new FileInputStream(args[2]);
            while((c=f1.read())!=-1)
            {
               System.out.print((char)c);
            }
            f1.close();
            break;
         default:
            System.out.println("Invalid No.of Arguments");
         }
      }
      catch(Exception e)
      {
         System.out.println(e);
      }
   }
}
```

11) Create an abstract class Person. Derive two classes Employee and Worker from it. Use proper method to accept and display the details for the same.
The fields of Employee are Emp_no,Emp_name and address. Similar fields for worker are name and working hours.

Solution:-

```
import java.io.*;
abstract class Person
{
  abstract void disp();
}
class Emp extends Person
{
  int eno;
  String ename,addr;
  Emp(int e,String en,String ad)
  {
    eno=e;
     ename=en;
     addr=ad;
  }
  void disp()
  {
     System.out.println("Emp Information");

     System.out.println(eno + " " + ename + " "+addr);
  }
}
class Worker extends Person
{
   String wName;
   int whr;
   Worker(int w,String wn)
   {
     whr=w;
     wName=wn;
   }
   void disp()
   {
     System.out.println("Worker Information");
     System.out.println(wName + " " + whr);
   }
}
```

```
class Ass11
{
  public static void main(String args[]) throws  Exception
  {
     int e,wh;
     String en,wn,addr;
     BufferedReader br=new BufferedReader (new InputStreamReader(System.in));
     System.out.println("Enter the Eno");
     e=Integer.parseInt(br.readLine());
     System.out.println("Enter the Ename");
     en=br.readLine();
     System.out.println("Enter the Addr");
     addr=br.readLine();
     System.out.println("Enter the WName");
     wn=br.readLine();
     System.out.println("Enter the Working hrs");
     wh=Integer.parseInt(br.readLine());
     Emp Eobj=new Emp(e,en,addr);
     Eobj.disp();
     Worker WObj=new Worker(wh,wn);
     WObj.disp();
  }
}
```

12) Write a java program to accept the details of employee from the user and display Payment slip on the next Frame.

Solution:-

Frame1:-

```
  import java.awt.*;
import java.awt.event.*;
import java.io.*;
class emp extends Frame implements ActionListener
{
    Label l1,l2,l3,l4,l5,l6;
    TextField txt1,txt2,txt3,txt4,txt5,txt6;
    Button next,cancel;
    Panel p1,p2,p3;
    Label ll;
    emp()
    {
            ll=new Label("EMPLOYEE SALARY INFORMTION");


            l1=new Label("Emp No. : -");
            l2=new Label("Name : -");
            l3=new Label("Basic Salary : -");
            l4=new Label("DA : -");
            l5=new Label("HRA : -");
            l6=new Label("Traveling Charges: -");

                txt1=new TextField(20);
                txt2=new TextField(20);
                txt3=new TextField(20);
                txt4=new TextField(20);
                txt5=new TextField(20);
                txt6=new TextField(20);

            next=new Button("NEXT");    next.addActionListener(this);
            cancel=new Button("CANCEL");   cancel.addActionListener(this);


            p1=new Panel();
            p1.setLayout(new GridLayout(6,2));
            p1.add(l1);
                p1.add(txt1);
```

```
            p1.add(l2);
                p1.add(txt2);
            p1.add(l3);
                p1.add(txt3);
            p1.add(l4);
                p1.add(txt4);
            p1.add(l5);
                p1.add(txt5);
            p1.add(l6);
                p1.add(txt6);

            p2=new Panel();
            p2.setLayout(new GridLayout(1,2));
            p2.add(next);
            p2.add(cancel);

            p3=new Panel();
            p3.add(ll);

            //Panel Arrangment
            add(p3,BorderLayout.NORTH);
            add(p1,BorderLayout.CENTER);
            add(p2,BorderLayout.SOUTH);

            setVisible(true);
            setSize(400,400);
            setTitle("Salary Details");


    }
    public void actionPerformed(ActionEvent ae)
    {
            if(ae.getSource()==next)
            {

                new
empdet(txt1.getText(),txt2.getText(),txt3.getText(),txt4.getText(),txt5.getText(),txt6.getText
());
            }
            if(ae.getSource()==cancel)
            {
                System.exit(0);
            }
    }
```

```
 public static void  main(String args[])
 {
     new emp().show();
 }
}
```

Frame 2:-

```
import java.awt.*;
import javax.swing.*;
import java.awt.event.*;
import java.io.*;
import java.util.*;

class empdet extends Frame implements ActionListener
{
     Label l1,l2,l3,l4,l5,l6,l7;
     Label lt1,lt2,lt3,lt4,lt5,lt6,lt7;
      Button cancel;
     JPanel p1=new JPanel();
     JPanel p2=new JPanel();
     JPanel p3=new JPanel();
     Label ll,ldt;
     empdet(String b,String c,String d,String e,String f,String g)
     {
               ll=new Label("Employee Payment Information");
               Date dt=new Date();
               ldt=new Label("Date: -"+d);

               l1=new Label("Emp No. : -");
               l2=new Label("Name : -");
               l3=new Label("Basic Salary : -");
               l4=new Label("DA : -");
               l5=new Label("HRA : -");
               l6=new Label("Traveling Charges: -");
               l7=new Label("Net Salary : -");

               lt1=new Label(b);
               lt2=new Label(c);
               lt3=new Label(d);
               lt4=new Label(e);
               lt5=new Label(f);
               lt6=new Label(g);
```

```
            double da=Integer.parseInt(d)*(Integer.parseInt(e)/100.0);
            double hra=Integer.parseInt(d)*(Integer.parseInt(f)/100.0);

            System.out.println(da+" "+hra);
            lt7=new Label(""+((Integer.parseInt(d)-(da+hra))+Integer.parseInt(g)));

            cancel=new Button("CLOSE");          cancel.addActionListener(this);


            p1.setLayout(new GridLayout(7,2));
            p1.add(l1);
                    p1.add(lt1);
            p1.add(l2);
                    p1.add(lt2);
            p1.add(l3);
                    p1.add(lt3);
            p1.add(l4);
                    p1.add(lt4);
            p1.add(l5);
                    p1.add(lt5);
            p1.add(l6);
                    p1.add(lt6);
            p1.add(l7);
                    p1.add(lt7);

            p2.setLayout(new GridLayout(1,1));

            p2.add(cancel);

            p3.add(ll);
            p3.add(ldt);

      p1.setBorder(BorderFactory.createTitledBorder(BorderFactory.createEtchedBorder(Col
or.BLACK, Color.BLACK)));
      p2.setBorder(BorderFactory.createTitledBorder(BorderFactory.createEtchedBorder(Col
or.BLACK, Color.BLACK)));
      p3.setBorder(BorderFactory.createTitledBorder(BorderFactory.createEtchedBorder(Col
or.BLACK, Color.BLACK)));

            add(p3,BorderLayout.NORTH);
            add(p1,BorderLayout.CENTER);
            add(p2,BorderLayout.SOUTH);
```

```
            setVisible(true);
            setSize(400,400);
     setTitle("Payment Slip of Employee");

    }
    public void actionPerformed(ActionEvent ae)
    {

            if(ae.getSource()==cancel)
            {
                    System.exit(0);
            }
    }


}
```

13) Write a java Program to accept ‘n’ no’s through the command line and store all the prime no’s and perfect no’s into the different arrays and display both the arrays.

Solution:-

```
import java.io.*;
class Ass13
{
    public static void main(String args[])
    {
        int n=args.length;
        int pr[]=new int[n];
        int per[]=new int[n];
        int k=0,i,j;
        int sum=0;

        //Prime No

        for(i=0;i<n;i++)
        {
            for(j=2;j<Integer.parseInt(args[i]);j++)
            {
                if(Integer.parseInt(args[i])%j==0)
                    break;

            }
            if(Integer.parseInt(args[i])==j)
                pr[k++]=Integer.parseInt(args[i]);
        }

        System.out.println("Prime Array = ");

        for(j=0;j<k;j++)
        {
            System.out.print("  "+pr[j]);
        }


        //Perfect No
        k=0;

        for(i=0;i<n;i++)
        {
            sum=0;
```

```
                for(j=1;j<Integer.parseInt(args[i]);j++)
                {

                        if(Integer.parseInt(args[i])%j==0)
                        {
                                sum=sum+j;

                        }
                }

                if(sum==Integer.parseInt(args[i]))
                {
                        per[k++]=sum;


                }


        }

        System.out.println("\nPerfect Array = ");

        for(i=0;i<k;i++)
        {
                System.out.print("  "+per[i]);
        }



    }

}
```

14) Write a program in Java to accept name of cities from the user and sort them in ascending order (Use Command Line Arguments)

Solution:-

```
class Ass14
{
  public static void main(String args[])
  {
    int n=args.length;
    int i,j;
    String str[]=new String[n];
    for(i=0;i<n;i++)
    {
      str[i]=args[i];
    }
    for(i=0;i<n;i++)
    {
      for(j=i+1;j<n;j++)
      {
        if((str[j].compareTo(str[i]))<0)
        {
          String tmp=str[j];
         str[j]=str[i];
          str[i]=tmp;
        }
      }
    }
    System.out.println("City Names After Sorting");
    for(i=0;i<n;i++)
    {
      System.out.println(str[i]);
    }
  }
}
```

15) Write a Java program to simulate the arithmetic calculator (+ ,- ,* , /).Also add 2 buttons namely 'on' to start the calculator and 'clear' to clear the text box.
Display appropriate error message in the text box.

Solution:-

```
import java.awt.*;
import java.awt.event.*;
public  class Ass15 extends Frame implements ActionListener
{
 Button on,clear,addn,subn,muln,divn,end,equal;
 TextField input;
 Panel pan1,pan2;
 int no1,no2,ans;
 char oper;
    public Ass15()
    {
            pan1= new Panel();
            pan2=new Panel();
            addn=new Button("+");
            subn=new Button("-");
            muln=new Button("*");
            divn=new Button("/");
            equal=new Button("=");
            end=new Button("Exit");
            on=new Button("On");
            clear=new Button("Clear");
            input=new TextField();
            pan1.add(addn);
            pan1.add(subn);
            pan1.add(muln);
            pan1.add(divn);
            pan1.add(equal);

            pan2.add(on);
            pan2.add(clear);
            pan2.add(end);

            setLayout(new BorderLayout());
            add(input,"North");
            add(pan1,"Center");
            add(pan2,"South");

            input.setEnabled(false);
```

```
        addn.setEnabled(false);
        subn.setEnabled(false);
        muln.setEnabled(false);
        divn.setEnabled(false);

        addn.addActionListener(this);
        subn.addActionListener(this);
        muln.addActionListener(this);
        divn.addActionListener(this);
        equal.addActionListener(this);
        end.addActionListener(this);
        on.addActionListener(this);
        clear.addActionListener(this);

        setTitle("ArithMetic Calculator");
        setSize(300,200);
        setVisible(true);
}
public static void main(String args[])
{
        new Ass15().show();

}
public void actionPerformed(ActionEvent e)
{
        Button btn= (Button)e.getSource();
        if(btn==end)
        {
                System.exit(0);
        }
        if(btn==addn || btn==subn || btn==muln || btn==divn)
        {
                no1=Integer.parseInt(input.getText());
                oper=btn.getLabel().charAt(0);
        }
        if(btn==equal)
        {
                no2=Integer.parseInt(input.getText());
                switch(oper)
                {
                        case '+' :
                                        ans=no1 + no2;
                                        break;
                        case '-' :
```

```
                                        ans=no1 - no2;
                                        break;
                                case '*' :
                                        ans=no1 * no2;
                                        break;
                                case '/' :
                                        ans=no1 / no2;
                                        break;
                        }
                        input.setText(Integer.toString(ans));
        }
        if(btn==clear)
        {
                input.setText("");
                input.requestFocus();
        }
        if(btn==on)
        {
               input.setEnabled(true);
                addn.setEnabled(true);
                subn.setEnabled(true);
                muln.setEnabled(true);
                divn.setEnabled(true);

        }
}
}
```

16) Write a Java program which will create a frame if we try to close it, it should it's color and it remains visible on the screen(Use swing).

Solution:-

```
import java.awt.*;
import java.awt.event.*;
import java.io.*;
import javax.swing.*;
class sframe extends JFrame
{
   JPanel p = new JPanel();
      sframe()
     {



                setVisible(true);
                setSize(400,400);
                setTitle("Swing Background");


       add(p);

                addWindowListener(new WindowAdapter()
                {
                public void windowClosing(WindowEvent e)
                {


              p.setBackground(Color.BLACK);
               JOptionPane.showMessageDialog(null,"Login
Successful","Login",JOptionPane.INFORMATION_MESSAGE);

       }
                });


     }

 public static void  main(String args[])
 {
     new sframe().show();
 }
```

```
}
```

17) Create a class Shape which consists one final method area () and volume ().Create three subclasses Rect, Circle and Triangle and calculate area and volume of it

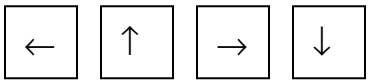
Solution:-

```
abstract class Shape
{
  final void area()
  {
    System.out.println("Final Method Can not be Override");
  }
  abstract void vol();
}
class Cir extends Shape
{
  float r,a,v;
  Cir(float ra)
  {
    r=ra;
  }
  void calArea()
  {
    area();
    a=3.14f*r*r;
    System.out.println("Area Of Circle is \t" + a);
  }
  void vol()
  {
    v=3.14f*r*r*r;
    System.out.println("Volume of Circle Is \t" +v);
  }
}



class Rect extends Shape
{
  float l,a,v,b,h;
  Rect(float la,float ba,float ha)
  {
    l=la;
    b=ba;
    h=ha;
  }
```

```
  void calArea()
  {
    area();
    a=l*b;
    System.out.println("Area Of Circle is \t" + a);
  }
  void vol()
  {
    v=l*b*h;
    System.out.println("Volume of Circle Is \t" +v);
  }
}


class Tri extends Shape
{
  float a,v,b,h;
  Tri(float ba,float ha)
  {
    b=ba;
    h=ha;
  }
  void calArea()
  {
    area();
    a=0.5f*h*b;
    System.out.println("Area Of Circle is \t" + a);
  }
  void vol()
  {
    v=(b*h)/3;
    System.out.println("Volume of Circle Is \t" +v);
  }
}
class Ass17
{
  public static void main(String args[])
  {
    Cir c=new Cir(5.0f);
    c.calArea();
    c.vol();
    Rect r=new Rect(5.0f,4.0f,4.2f);
    r.calArea();
    r.vol();
```

```
    Tri t=new Tri(6.0f,2.0f);
    t.calArea();
    t.vol();
  }

}
```

18) Create an application in Java using Awt to display 4 X 4 squares on the screen. One of the block will be active with black color. All other blocks should be filled with blue color. Provide command buttons as follows to move the active cell. The active cell should be changed only if it is within the boundary of the squares otherwise give the beep.



Solution:-

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
public class Ass18 extends JFrame implements ActionListener
{
 JButton moveBtns[]=new JButton[4];
 JPanel gridPanels[]=new JPanel[16];
 String icons[]={"Left","Up","Right","Down"};
 int cnt=1;
 public Ass18()
 {
    setLayout(new GridLayout(5,4));
    for(int i=0;i<16;i++)
    {
        gridPanels[i]=new JPanel();
        add(gridPanels[i]);
        gridPanels[i].setBorder(BorderFactory.createLineBorder(Color.black,1));
        gridPanels[i].setBackground(Color.blue);
    }
          gridPanels[0].setBackground(Color.black);
      for(int i=0;i<4;i++)
          {
                    moveBtns[i]=new JButton(icons[i]);
                    add(moveBtns[i]);
                    moveBtns[i].addActionListener(this);
          }

          setSize(300,300);
          setVisible(true);
          setTitle("Grid Example");
          setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
 }
          public void actionPerformed(ActionEvent e)
          {
                    JButton btn=(JButton) e.getSource();
```

```
                gridPanels[cnt-1].setBackground(Color.blue);
                if(btn==moveBtns[0])
                {
                        if(cnt%4!=1)
                                cnt--;
                }
                if(btn==moveBtns[1])
                {
                        if(cnt>4)
                                cnt=cnt-4;
                }
                if(btn==moveBtns[2])
                {
                        if(cnt%4!=0)
                                cnt++;
                }
                if(btn==moveBtns[3])
                {
                        if(cnt<13)
                                cnt=cnt+4;
                }
                gridPanels[cnt-1].setBackground(Color.black);
        }


                public static void main(String args[])
                {
                        new Ass18().show();
                }
}
```

19) Create an abstract class Shape. Derive three classes sphere, cone and cylinder from it. Calculate area and volume of all. (Use Method overriding)

Solution:-

```
import java.io.*;
abstract class Shape
{
 abstract void area();
 abstract void vol();
}
class Sphere extends Shape
{
 float r,a,v;
 Sphere(float ra)
 {
  r=ra;
 }
 void area()
 {
   a=4* 3.14f*r*r;
   System.out.println("Area OF Sphere is \t" +a);
 }
 void vol()
 {
  v=(4 * 3.14f * r*r*r)/3;
  System.out.println("Volume Of Sphere Is \t" +v);
 }
}
 class Cyl extends Shape
{
 float r,a,v,h;
 Cyl(float ra,float ha)
 {
  r=ra;
  h=ha;
 }
 void area()
 {
   a=2* 3.14f*r*(h+r);
   System.out.println("Area OF Cyl is \t" +a);
 }
 void vol()
 {
```

```
   v= 3.14f * r*r*h;
   System.out.println("Volume Of Cyl Is \t" +v);
  }
}
class Cone extends Shape
{
  float r,a,v,l,h;
  Cone(float ra,float ha,float la)
  {
    r=ra;
    l=la;
    h=ha;
  }
  void area()
  {
     a= 3.14f*r*(l+r);
     System.out.println("Area OF Cone is \t" +a);
  }
  void vol()
  {
    v=(3.14f * r*r*h)/3;
    System.out.println("Volume Of Cone Is \t" +v);
  }
}
class Ass19
{
  public static void main(String args[]) throws Exception
  {
    Sphere sp=new Sphere( 5.2f);
    sp.area();
    sp.vol();
    Cyl cp=new Cyl( 5.2f,3.2f);
    cp.area();
    cp.vol();
    Cone co=new Cone(2.3f,1.2f, 5.2f);
    co.area();
    co.vol();
  }
}
```

20) Write a Java program to design a screen using Awt that will take a user name and password. If the user name and password are not same, raise an exception with appropriate message. User can have 3 login chances only. Use clear button to clear the text boxes.

Solution:-

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

class InvalidPasswordException extends Exception
{}

public class Ass20 extends JFrame implements ActionListener
{
	JLabel name, pass;
	JTextField nameText;
	JPasswordField passText;
	JButton login, end;
	static int attempt=0;

	public Ass20()
	{
		name = new JLabel("Name:",JLabel.RIGHT);
		pass = new JLabel("Password:",JLabel.RIGHT);

		nameText = new JTextField(20);
		passText = new JPasswordField(20);

		login = new JButton("Login");
		end = new JButton("End");

		login.addActionListener(this);
		end.addActionListener(this);


		setLayout(new GridLayout(3,2));
		add(name);
		add(nameText);
		add(pass);
		add(passText);
		add(login);
		add(end);
```

```
        setTitle("Login Check");
        setSize(300,300);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setVisible(true);
    }

    public void actionPerformed(ActionEvent ae)
    {
        JButton btn = (JButton)ae.getSource();
        if(btn == end)
        {
            System.exit(0);
        }
        if(btn == login)
        {
            try
            {
                String user = nameText.getText();
                String pass = new String(passText.getPassword());
                if(user.compareTo(pass)==0)
                {
                    JOptionPane.showMessageDialog(null,"Login
Successful","Login",JOptionPane.INFORMATION_MESSAGE);
                    System.exit(0);
                }
                else
                {
                    throw new InvalidPasswordException();
                }
            }
            catch(Exception e)
            {
                attempt++;
                JOptionPane.showMessageDialog(null,"Login
Failed","Login",JOptionPane.ERROR_MESSAGE);
                nameText.setText("");
                passText.setText("");
                nameText.requestFocus();
                if(attempt == 3)
                {
                    JOptionPane.showMessageDialog(null,"3 Attempts
Over","Login",JOptionPane.ERROR_MESSAGE);
                    System.exit(0);
                }
```

```
                }
            }
        }

        public static void main(String args[])
        {
            new Ass20();
        }
}
```

21) Design a screen in Java to handle the mouse events and display the positions of the mouse click in a text box.

Solution:-

```
import java.awt.*;
import java.awt.event.*;
public class Ass21 extends Frame
{
	TextField statusBar;
	public static void main(String []args)
	{
		new Ass21().show();
	}

	public Ass21()
	{
		addMouseListener(new MouseAdapter()
		{
			public void mouseClicked(MouseEvent e)
			{
				statusBar.setText("Clicked at (" + e.getX() + "," + e.getY() + ")");
				repaint();
			}
			 public void mouseEntered(MouseEvent e)
			{
				statusBar.setText("Entered at (" + e.getX() + "," + e.getY() + ")");
				repaint();
			}


		});
		addWindowListener(new WindowAdapter()
		{
			public void windowClosing(WindowEvent e)
			{
				System.exit(0);
			}
		});
		setLayout(new FlowLayout());
		setSize(275,300);
		setTitle("Mouse Click Position");
		statusBar = new TextField(20);
		add(statusBar);
```

```
            setVisible(true);
        }
    }
```

22) Write a java program that displays the number of characters, lines & words from a file

Solution:-

```
import java.io.*;
class Ass22
{
        public static void main(String args[]) throws Exception
        {
                int ccnt=0,lcnt=1,wcnt=0,c;
                FileInputStream fin=new FileInputStream(args[0]);
              while((c=fin.read())!=-1)
              {
                        ccnt++;
                      if(c==32 || c==13)
                              wcnt++;
                      if(c==13)
                               lcnt++;
               }
              System.out.println("Number of Characters are " + ccnt);
              System.out.println("Number of Words are " + wcnt);
              System.out.println("Number of Lines are " + lcnt);
        }
}
```

23) Write a java program that reads line of integers. Display each integer and also display sum of all integers

Solution:-

```
import java.io.*;
class Ass23
{
  public static void  main(String args[]) throws Exception
  {
    BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
    int n,k,s=0;
    System.out.println("Enter The Integer Number");
    n=Integer.parseInt(br.readLine());
    while(n>0)
    {
          k=n % 10;
       System.out.print(" "+k+" ");
       n=n/10;
       s=s+k;
    }
    System.out.println();
    System.out.println("Sum Of Digits Of Number is \t" + s);
  }
}
```

24) Write a java program to accept a number from the user, if number is zero then throw user defined exception “Number is 0” otherwise check whether no is prime or not (Use static keyword).

Solution:-

```
import java.io.*;
class ZeroExc extends Exception { }
class Ass24
{
 static int n;

 public static void main(String args[])
 {
   int i;

  try
  {
        BufferedReader br=new BufferedReader (new InputStreamReader(System.in));
    System.out.println("Enter the Number");
        int n=Integer.parseInt(br.readLine());
    if(n==0)
    {
     throw new ZeroExc();
    }
    else
        {
                for(i=2;i<n;i++)
                {
                        if(n %i==0)
                                break;
                }
                if(i==n)
                        System.out.println("Number Is Prime");
                else
                        System.out.println("Number Is Not Prime");
        }
  }
  catch(ZeroExc e)
  {
    System.out.println("Number Is Zero");
  }
  catch(Exception ea)
  {
```

```
    }
  }
}
```

25) Write a package for Games, which have two classes Indoor and Outdoor. Use a function display () to generate the list of players for the specific games. Use default and parameterized constructors. Make use of protected access specifier. Also use finalize () method.

Solution:-

```
import Games.*;

class Ass25
{
	public static void main(String args[])
	{
		Indoor In[] = new Indoor[3];
		In[0] = new Indoor("Amit");
		In[1] = new Indoor("Aniket");
		In[2] = new Indoor("Amol");
		System.out.println("Indoor Players...");
		for(int i=0;i<In.length;i++)
			In[i].display();
		System.out.println("Indoor Players...");
		Outdoor Out[] = new Outdoor[3];
		Out[0] = new Outdoor("Anil");
		Out[1] = new Outdoor("Ameya");
		Out[2] = new Outdoor("Ajay");
		for(int i=0;i<Out.length;i++)
			Out[i].display();
	}
}
```

Games:-

```
package Games;
public class Indoor
{
	String player;

	public Indoor(String p)
	{
		player = p;
	}
	public void display()
	{
		System.out.println(player);
```

```
	}
	protected void finalize()
	{
		System.out.println("Terminating Indoor...");
	}
}

package Games;
public class Outdoor
{
	String player;

	public Outdoor(String p)
	{
		player = p;
	}
	public void display()
	{
		System.out.println(player);
	}
	protected void finalize()
	{
		System.out.println("Terminating Outdoor...");
	}
}
```

26) Create a package TYBSc which will have 2 classes as class Mathematics with a methods to add two numbers, add three float numbers and class Maximum with a method to find maximum of three numbers. (Make use of finalize)

Solution:-

```
import TYBSc.*;
class Ass26
{
	public static void main(String args[])
	{
		Maximum ob = new Maximum();
		System.out.println(ob.max(5,6,7));
		Mathematics ob1 = new Mathematics();
		System.out.println(ob1.add(5,6));
		System.out.println(ob1.add(5f,6f,8f));
	}
}
```

TyBsc Package:-

```
package TYBSc;
public class Mathematics
{
	public int add(int x, int y)
	{
		return x+y;
	}
	public float add(float x, float y, float z)
	{
		return x+y+z;
	}
	protected void finalize()
	{
		System.out.println("Terminating Mathematics...");
	}
}
```

```
package TYBSc;
public class Maximum
{
	public int max(int a, int b, int c)
	{
		return a>b?(a>c?a:c):(b>c?b:c);
```

```
	}
	protected void finalize()
	{
		System.out.println("Terminating Maximum...");
	}
}
```

27) Write a class Student with attributes roll no, name, age and course. Initialize values through parameterized constructor. If age of student is not in between 15 and 21 then generate user-defined exception “Age Not Within The Range”.If name contains numbers or special symbols raise exception “Name not valid”.

Solution:-

```
x
import java.io.*;
class AgeOutOfRangeException extends Exception{}
class NameNotValidException extends Exception{}

class Student
{
    int RollNo, Age,j,k=-1;
    String Name,Course;
    char ch[]={'!','@','#','$','%','^','&','*'};
    Student(int rno, String name, int age, String course)
    {
        try
        {
            RollNo = rno;
            Name = name;
            Age = age;
            Course = course;
            if(Age<15 || Age>21)
                throw new AgeOutOfRangeException();
            for(int i=0,j=0; i<Name.length() && j<ch.length ; i++,j++)
                if(Character.isDigit(Name.charAt(i))||(k!=Name.indexOf(ch[j])))
                    throw new NameNotValidException();
        }
        catch(AgeOutOfRangeException e)
        {
            System.out.println("Age Not Within The Range");
            System.exit(0);
        }
        catch(NameNotValidException e)
        {
            System.out.println("Name not valid");
            System.exit(0);
        }
    }
    public void printData()
    {
        System.out.println("Roll No:"+RollNo+"\n"+
```

```
                    "Name   :"+Name+"\n"+
                    "Age    :"+Age+"\n"+
                    "Course :"+Course);
    }
}

class Ass28
{
    public static void main(String args[]) throws Exception
    {
        BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
          System.out.println("Enter the RollNo");
          int rno=Integer.parseInt(br.readLine());
        System.out.println("Enter the Name");
        String sn=br.readLine();
        System.out.println("Enter the age");
          int age=Integer.parseInt(br.readLine());
        System.out.println("Enter the cource");
        String cn=br.readLine();
        Student s1 = new Student(rno,sn,age,cn);
        s1.printData();

    }
}
```

28) Write a Java program to accept list of files as command line arguments. Display the name and size of all the files. Delete all files with extension as '.html' from the current directory. Appropriate error messages should be printed.(Use vector Array)

Solution:-

```
import java.io.*;
import java.util.*;
class Ass29
{
  public static void main(String args[]) throws Exception
 {
    String str[]=new String[args.length];
    Vector list=new Vector();
    for(int i=0;i<args.length;i++)
    {
        File file=new File(args[i]);
        if(file.isFile()==true)
      {
                System.out.println(args[i]+"\t"+file.length());
      }
      else
      {
                System.out.println(args[i]+ "is not a file");
      }
    }

    for(int i=0;i<args.length;i++)
    {
      list.addElement(args[i]);
    }
      list.copyInto(str);
   for(int i=0;i<str.length;i++)
    {
      if(str[i].endsWith("html"))
      {

          File tmp=new File(str[i]);
          if(tmp.delete())
                {
                        System.out.println ("File Deleted");
                }
      }
    }
```

```
}
}
```

29) Write a Java program to design a screen using Awt that will create four text fields. First for the text, second for what to find and third for replace. Display result in the fourth text field. Also display the count of total no. of replacements made. The button clear to clear the text boxes.

Solution:-

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

public class Ass30 extends JFrame implements ActionListener
{
    JLabel textLabel, findLabel, replaceLabel, occurrenceLabel;
    JTextArea text;
    JTextField findText, replaceText, occurrenceText;
    JButton find, replace, exit;
    JPanel pan1,pan2;
    int occurrences=0,i=0;

    public Ass30()
    {
        textLabel = new JLabel("Enter Text:",JLabel.RIGHT);
        findLabel = new JLabel("Text to Find:",JLabel.RIGHT);
        replaceLabel = new JLabel("Text to Replace:",JLabel.RIGHT);
        occurrenceLabel = new JLabel("No.of Occurrences:",JLabel.RIGHT);

        text = new JTextArea(1,20);
        findText = new JTextField(20);
        replaceText = new JTextField(20);
        occurrenceText = new JTextField(20);

        pan1 = new JPanel();
        pan1.setLayout(new GridLayout(4,2));
        pan1.add(textLabel);
        pan1.add(text);
        pan1.add(findLabel);
        pan1.add(findText);
        pan1.add(replaceLabel);
        pan1.add(replaceText);
        pan1.add(occurrenceLabel);
        pan1.add(occurrenceText);

        find = new JButton("Find");
```

```
        replace = new JButton("Replace");
        exit = new JButton("Exit");

        find.addActionListener(this);
        replace.addActionListener(this);
        exit.addActionListener(this);

        pan2 = new JPanel();
        pan2.setLayout(new FlowLayout());
        pan2.add(find);
        pan2.add(replace);
        pan2.add(exit);
        add(pan1,"Center");
        add(pan2,"South");

        setTitle("Find And Replace");
        setSize(300, 200);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setVisible(true);
}
public static void main(String[] args)
{
        new Ass30();
}
public void actionPerformed(ActionEvent ae)
{
        JButton btn = (JButton)ae.getSource();
        if(btn == find)
        {
                String s = text.getText();
                String f = findText.getText();
                i = s.indexOf(f);
                 if(i!=-1)
                {
                        occurrences++;
                        occurrenceText.setText(Integer.toString(occurrences));
                        text.select(i,i+f.length());
                        text.requestFocus();

                }
        }
        if(btn == replace)
        {
                if(text.getSelectedText().length()!=0)
```

```
                {
                    String r = replaceText.getText();
                    text.replaceSelection(r);
                }
            }
            if(btn == exit)
            {
                System.exit(0);
            }
        }
}
```