

T.Y.B.C.A

CORE-JAVA

PRACTICAL SOLUTION

BY

Prof. Bhujbal V.N

1. Write a java program to read the characters from a file, if a character is alphabet then reverse its case, if not then display its category on the Screen. (whether it is Digit or Space)

```
import java.io.*;
import java.util.*;
class Check
{
    public static void main(String args[])
    {
        char ch;
        String line="";
        FileInputStream f1;
        int i;
        try
        {
            f1=new FileInputStream("a.txt");
            Scanner scan=new Scanner(f1);
            while(scan.hasNextLine())
            {
                line=scan.nextLine();
                for(int j=0;j<line.length();j++)
                {
                    ch=(char)line.charAt(j);
                    if (ch==' ') //if(Character.isSpaceChar(ch))
                        System.out.println("Character is Space");
                    if(Character.isDigit(ch)==true) // if(Character.isDigit(ch))
                        System.out.println("Character is digit");
                    i=(int)line.charAt(j);
                    if ((i>=65)&&(i<=91))
                    {
                        System.out.println("Character is Capital Alphabet ");
                        i=i+32;
                        ch=(char)i;
                        System.out.println(" Its Small Alphabet is : "+ch);
                    }
                    else if((i>=97)&&(i<=122))
                    {
                        System.out.println("Character is Small Alphabet ");
                        i=i-32;
                        ch=(char)i;
                        System.out.println(" Its Capital Alphabet is : "+ch);
                    }
                }
            }
        }
        catch(Exception e){}
    }
}
```

2. Write a java program to accept n names of cities from user and display them in descending order.

```
import java.io.*;
class CityDemo
{
    public static void main(String args[])
    {
        String city[]=new String[5];
        int i,j;
        try
        {
            BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
            for(i=0;i<5;i++)
            {
                System.out.println("Enter the "+(i+1)+" city");
                city[i]=br.readLine();
            }
            System.out.println("-*-*-*-*-*Descending order of cities-*-*-*-*-*");
            for(i=0;i<5;i++)
            {
                for(j=i+1;j<5;j++)
                {
                    if((city[i].compareTo(city[j]))>0)
                    {
                        String temp=city[i];
                        city[i]=city[j];
                        city[j]=temp;
                    }
                }
            }
            for(i=4;i>=0;i--)
            {
                System.out.println(city[i]);
            }
            System.out.println("-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*");
        }
        catch(Exception e){ }
    }
}
```

3. Write a java program to accept the details of 'n' employees (ENAME,Salary) from the user, store them into the Hashtable and displays the Employee Names having maximum Salary.

```
import java.util.*;
import java.io.*;
class HashEmp
{
    public static void main(String args[])
    {
        DataInputStream din=new DataInputStream(System.in);
```

```

    Hashtable h1=new Hashtable();
    Enumeration e;
    String nm,str;
    double sal,b,max=0;
    int n,i;
    try
    {
        System.out.print("Enter No of Record =");
        n=Integer.parseInt(din.readLine());
        for(i=0;i<n;i++)
        {
            System.out.println("-----");
            System.out.print("Enter Name=");
            nm=din.readLine();
            System.out.print("Enter Salary=");
            sal=Double.valueOf(din.readLine()).doubleValue();
            h1.put(nm,sal);
        }

        e=h1.keys();
        System.out.println("-----Information of Employee-----");
        System.out.println("-----");
        System.out.println("Name      Salary ");
        while(e.hasMoreElements())
        {
            str=(String)e.nextElement();
            System.out.println(str+"      "+h1.get(str));
            b=((Double)h1.get(str)).doubleValue();
            if(max<b)
            {
                max=b;
            }
        }
        System.out.println("-----");
        System.out.println("Maximum Salary is "+max);
        System.out.println("-----");
    }
    catch(Exception e1){}
}
}

```

4. Design a screen in Java to handle the Mouse Events such as MOUSE_MOVED and MOUSE_CLICK and display the position of the Mouse_Click in a TextField.

```

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

public class MousePos extends Frame
{
    TextField statusBar;
    public static void main(String []args)
    {
        new MousePos().show();
    }
}

```

```

    }

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

```

5. Define a class Student with attributes rollno and name. Define default and parameterized constructor. Override the toString() method. Keep the count of Objects created. Create objects using parameterized constructor and Display the object count after each object is created

```

import java.io.*;
class Student1
{
    public int rollno;
    static int cnt=0;
    public String name;

    Student1()
    {
        cnt++;
        System.out.println("Count Objects="+cnt);
    }

    Student1(int rno,String name1)
    {
        try

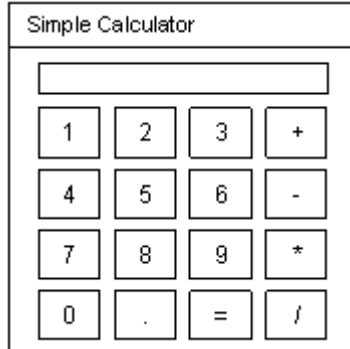
```

```
        {
            rollno=rno;
            name=name1;
            cnt++;
            System.out.println("Count Objects="+cnt);
        }
    } catch(Exception e){}
}

public String toString()
{
    return rollno+" "+name;
}

public static void main(String args[])
{
    try
    {
        int no,rno;
        String name;
        System.out.println("\nEnter no of Students:");
        BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
        no=Integer.parseInt(br.readLine());
        Student1 obj[]=new Student1[no];
        System.out.println("\nEnter the Data:");
        for(int i=0;i<no;i++)
        {
            System.out.println("Roll No:");
            rno=Integer.parseInt(br.readLine());
            System.out.println("Enter the Name:");
            name=br.readLine();
            obj[i]=new Student1(rno,name);
        }

        for(int i=0;i<no;i++)
        {
            System.out.println("Override toString():"+obj[i].toString());
        }
    }
    catch(IOException e)
    {
        System.out.println(e);
    }
}
}
```

6. Create a calculator with functionality in an Applet.**Cal.java**

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

public class cal extends Applet implements ActionListener
{
    int a,b,c;
    TextField t1;
    Button

    b1,b2,b3,b4,b5,b6,b7,b8,b9,b10,b11,b12,b13,b14,b15,b16;
    String s,s1,s2,s3,s4;
    public void init()
    {
        setLayout(null);
        t1=new TextField(10);
        t1.setBounds(80,200,260,40);
        add(t1);
        b1=new Button("0");
        b2=new Button("1");
        b3=new Button("2");
        b4=new Button("3");
        b5=new Button("4");
        b6=new Button("5");
        b7=new Button("6");
        b8=new Button("7");
        b9=new Button("8");
        b10=new Button("9");
        b11=new Button("+");
        b12=new Button("-");
        b13=new Button("*");
        b14=new Button("/");
        b15=new Button("=");
        b16=new Button("CLR");
        b1.setBounds(90,260,40,30);
        add(b1);
        b2.setBounds(140,260,40,30);
        add(b2);
        b3.setBounds(190,260,40,30);
        add(b3);
```

```
        b4.setBounds(240,260,40,30);
        add(b4);
        b5.setBounds(290,260,40,30);
        add(b5);
        b6.setBounds(90,300,40,30);
        add(b6);
        b7.setBounds(140,300,40,30);
        add(b7);
        b8.setBounds(190,300,40,30);
        add(b8);
        b9.setBounds(240,300,40,30);
        add(b9);
        b10.setBounds(290,300,40,30);
        add(b10);
        b11.setBounds(90,340,40,30);
        add(b11);
        b12.setBounds(140,340,40,30);
        add(b12);
        b13.setBounds(190,340,40,30);
        add(b13);
        b14.setBounds(240,340,40,30);
        add(b14);
        b15.setBounds(290,340,40,30);
        add(b15);
        b16.setBounds(90,380,70,20);
        add(b16);
        b1.addActionListener(this);
        b2.addActionListener(this);
        b3.addActionListener(this);
        b4.addActionListener(this);
        b5.addActionListener(this);
        b6.addActionListener(this);
        b7.addActionListener(this);
        b8.addActionListener(this);
        b9.addActionListener(this);
        b10.addActionListener(this);
        b11.addActionListener(this);
        b12.addActionListener(this);
        b13.addActionListener(this);
        b14.addActionListener(this);
        b15.addActionListener(this);
        b16.addActionListener(this);
    }
    public void actionPerformed(ActionEvent ae)
    {
        repaint();
        s=ae.getActionCommand();
        if(s.equals("0") || s.equals("1") || s.equals("2")||
            s.equals("3") || s.equals("4") || s.equals("5")||s.equals
            ("6")||s.equals("7")||s.equals("8")||s.equals("9"))
        {
            s1=t1.getText()+s;
            t1.setText(s1);
        }

        if(s.equals("+"))
```

```
        {
            s2=t1.getText();
            t1.setText("");
            s3="+";
        }

        if(s.equals("-"))
        {
            s2=t1.getText();
            t1.setText("");
            s3="-";
        }

        if(s.equals("*"))
        {
            s2=t1.getText();
            t1.setText("");
            s3="*";
        }

        if(s.equals("/"))
        {
            s2=t1.getText();
            t1.setText("");
            s3="/";
        }

        if(s.equals("="))
        {
            s4=t1.getText();
            a=Integer.parseInt(s2);
            b=Integer.parseInt(s4);
            if(s3.equals("+"))
                c=a+b;

            if(s3.equals("-"))
                c=a-b;

            if(s3.equals("*"))
                c=a*b;
            if(s3.equals("/"))
                c=a/b;
            t1.setText(String.valueOf(c));
        }

        if(s.equals("CLR"))
        {
            t1.setText("");
        }
    }

    public void paint(Graphics g)
    {
        setBackground(Color.gray);
    }
}
```

```
        g.drawRect(80,200,260,200);
        showStatus("Applet");
        g.drawString("CALCULATER",200,50);
    }
}
```

Cal.html

```
<html>
<applet code="cal" width=500 height=500></applet>
</html>
```

7. Write a java program to display “Hello Java” with settings Font- Georgia, Foreground color- Red, background color – Blue on the Frame (Use Label)

```
import java.awt.*;
public class StringDemo extends Frame
{
    Font f;
    Label l1,l2;

    public StringDemo()
    {
        setLayout(new FlowLayout());
        l1=new Label("Hello Java");
        l1.setForeground(Color.RED);
        l1.setFont(new Font("Georgia", Font.BOLD, 12));
        setBackground(Color.BLUE);
        add(l1);
    }

    public static void main(String args[])
    {
        StringDemo f1=new StringDemo();
        f1.setTitle("String operation");
        f1.setSize(300,300);
        f1.setVisible(true);
    }
}
```

8. Write a java program to design a following GUI (Use Swing).
[Marks 30]

Personal Information

First Name :

Last Name :

Address :

Mobile Number :

Gender ☐ Male ☐ Female

Your Interests ☐ Computer ☐ Sports ☐ Music

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

public class SwingDemo extends JFrame
{

    private JPanel jpInfo = new JPanel();
    private JLabel lbName, lbLname, lbadd, lbmobno, lbgender, lbint, lbinfo;
    private JTextField txt1, txt2, txt3, txt4;
    private CheckboxGroup rdgroup;
    Checkbox radio1, radio2, check1, check2, check3;

    private JButton btnSubmit, btnreset;

    SwingDemo()
    {

        rdgroup = new CheckboxGroup();

        jpInfo.setBounds(0, 0, 500, 115);
        jpInfo.setLayout(null);

        lbinfo = new JLabel("Personal Information:");
        lbinfo.setBounds(80, 20, 150, 25);

        lbName = new JLabel("First Name:");
        lbName.setForeground(Color.black);
        lbName.setBounds(15, 55, 80, 25);
        txt1 = new JTextField();
        txt1.setBounds(100, 55, 150, 25);

        lbLname = new JLabel("Last Name:");
        lbLname.setForeground(Color.black);
        lbLname.setBounds(15, 75, 80, 25);
        txt2 = new JTextField();
```

```
txt2.setBounds (100,75,150,25);
```

```
lbadd = new JLabel ("Address:");  
lbadd.setForeground (Color.black);  
lbadd.setBounds (15, 95, 80, 25);  
txt3 = new JTextField ();  
txt3.setBounds (100,95,150,25);
```

```
lbmobno = new JLabel ("Mobile No:");  
lbmobno.setForeground (Color.black);  
lbmobno.setBounds (15, 125, 80, 25);  
txt4 = new JTextField ();  
txt4.setBounds (100,125,150,25);
```

```
lbgender = new JLabel ("Gender");  
lbgender.setBounds (15, 145, 80, 25);  
radio1=new Checkbox("Male",rdgroup,true);  
radio1.setBounds (100, 150,50,20);  
radio2=new Checkbox("Female",rdgroup,true);  
radio2.setBounds (150, 150,100,20);  
lbint=new JLabel ("Your Interest:");  
lbint.setBounds (15,170,80,30);
```

```
check1=new Checkbox("Computer");  
check1.setBounds (100,175,80,20);  
check2=new Checkbox("Sport");  
check2.setBounds (190,175,50,20);  
check3=new Checkbox("Music");  
check3.setBounds (250,175,50,20);
```

```
//Aligning The Buttons.
```

```
btnSubmit = new JButton ("Submit");  
btnSubmit.setBounds (20, 230, 100, 25);
```

```
btnreset=new JButton("Reset");  
btnreset.setBounds(120,230,100,25);
```

```
//Adding the All the Controls to Panel.
```

```
jpInfo.add (lbName);  
;  
jpInfo.add (lbLname);  
jpInfo.add (lbadd);  
jpInfo.add (lbmobno);  
jpInfo.add (lbgender);  
jpInfo.add (lbinfo);
```

```
jpInfo.add (txt1);  
jpInfo.add (txt2);
```

```
        jpInfo.add (txt3);
        jpInfo.add (txt4);

        jpInfo.add (check1);
        jpInfo.add (check2);
        jpInfo.add (check3);

        jpInfo.add (radio1);
        jpInfo.add (radio2);
        jpInfo.add (lbint);

        jpInfo.add (btnSubmit);

        jpInfo.add( btnreset);
        //Adding Panel to Window.
        getContentPane().add (jpInfo);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        //In the End Showing the New Account Window.
        setVisible (true);
        setSize(350,350);
        setTitle("Personal Information");

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

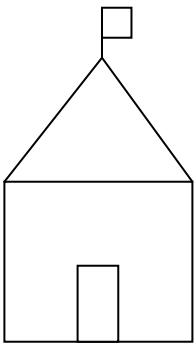
9. Write a java program to display the contents of a file in reverse order

```
import java.io.*;
import java.util.*;
class RevFile
{
    public static void main(String args[])
    {
        char ch;
        String line="";
        FileInputStream f1;
        int i=0;
        try
        {
            f1=new FileInputStream("a.txt");
            Scanner scan=new Scanner(f1);

            while(scan.hasNextLine())
            {
                String st=scan.nextLine();
```

```
        StringBuffer buf=new StringBuffer(st);
        buf=buf.reverse();
        String rs=buf.toString();
        System.out.println(rs);
    }
}
catch(Exception e) {}
}
```

10. Write an applet application in Java for designing Temple.



DrawTemple.java

```
import java.applet.Applet;
import java.awt.Color;
import java.awt.Graphics;
public class DrawTemple extends Applet
{
    public void paint(Graphics g)
    {
        g.drawRect(100,250,100,100);
        g.drawRect(140,300,20,50);
        g.drawLine(100,250,150,130);
        g.drawLine(200,250,150,130);
        g.drawLine(150,130,150,50);
        g.drawRect(150,50,10,30);
    }
}
```

DrawTemple.html

```
<html>
<applet code="DrawTemple" width=500 height=500></applet>
</html>
```

11. Write a java program to accept Employee name from the user and check whether it is valid or not. If it is not valid then throw user defined Exception “Name is Invalid” otherwise display it.

```
import java.io.*;
import java.util.*;
class InvalidName extends Exception
{
    InvalidName(String msg)
    {
        super(msg);
    }
}

public class ExceptionDemo
{
    String name,str;
    char ch;
    int j,flag=0;
    DataInputStream din=new DataInputStream(System.in);
    BufferedReader br=new BufferedReader(new InputStreamReader(System.in));

    public void userExce()
    {
        try
        {
            System.out.print("Enter the Employee Name : ");
            str=br.readLine();
            name=str;
            for(int j=0;j<name.length();j++)
            {
                ch=(char)name.charAt(j);
                if(Character.isDigit(ch))
                {
                    flag=1;
                }
                if(Character.isLetter(ch))
                {
                }
                else
                {
                    flag=1;
                }
            }

            if(flag==1)
            {
                throw new InvalidName("Name is Invalid");
                //throws userdefined Exception.
            }
            else
            {
            }
            }
        }
    }
}
```

```

        {
            System.out.println("Name="+name);
        }

    } // Closing of Try Block.
    catch(InvalidName z)
    {
        System.out.println(z.getMessage());
    }
    catch(Exception e)
    {
        System.out.println(e.getMessage());
    }
} //Closing of userExce function.

public static void main(String args[])
{
    ExceptionDemo n1=new ExceptionDemo();
    n1.userExce();
}
}

```

12. Write a java program to accept list of file names through command line and delete the files having extension “.txt”. Display the details of remaining files such as FileName and size.

```

import java.io.*;
import java.util.*;
class DelFileDemo1
{
    public static void main(String args[])throws IOException
    {
        Vector v=new Vector();
        int i;
        for(i=0;i<args.length;i++)
        {
            v.addElement(args[i]);
        }

        int size=v.size();
        String arr[]=new String[size];
        v.copyInto(arr);
        for(i=0;i<size;i++)
        {
            File f1=new File(arr[i]);
            if(arr[i].endsWith(".txt")==true)
            {
                System.out.println("txt File Found!!!!");
                f1.delete();
            }
        }
    }
}

```

```

        System.out.println(args[i]+" File Deleted!!!!!!");
        System.out.println("-----");
    }
    else
    {
        System.out.println("-----");
        System.out.println(" Name : "+f1.getName());
        System.out.println(" Size : "+f1.length());
        System.out.println("-----");
    }
}
}
}

```

13. Write a java program to copy the contents of one file into the another file, while copying change the case of alphabets and replace all the digits by '*' in target file.

```

import java.io.*;
import java.util.*;
class FileCopyCase
{
    public static void main(String args[])
    {
        int i;
        FileInputStream f1=null; // Create Object FileInputStream
        FileOutputStream f2=null; // Create Object FileOutputStream
        try
        {
            f1=new FileInputStream("f1.txt"); //Read first File1
            f2=new FileOutputStream("f2.txt"); //Open Second File
            while((i=f1.read())!=-1)
            {
                if ((i>=65)&&(i<=96))
                {
                    i=i+32;
                    f2.write(i);
                }
                else if((i>=97)&&(i<=122))
                {
                    i=i-32;
                    f2.write(i);
                }
                if(Character.isDigit(i))
                {
                    f2.write('*');
                }
            }
            f1.close();
            f2.close();
            System.out.println("-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*");
            System.out.println("File f1 Copied into File f2 Successfully !!!!");
        }
        catch(Exception e){}
    }
}

```

}

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

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

**15. Define an abstract class Shape with abstract methods area() and volume().
Write a java program to calculate area and volume of Cone and Cylinder.**

```
abstract class shape
{
    final double pi=3.14;
    double rd,ht;
    shape (double r,double h) //Constructor
    {
        rd=r;
        ht=h;
    }
    abstract double area();
    abstract double volume();
}

class cylinder extends shape
{

```

```
cylinder(double r,double h)
{
    super(r,h);
}

double area() // Area Of cylinder
{
    return((2*pi*rd*ht)+(2*pi*rd*rd));
}

double volume() //Volume Of cylinder
{
    return(pi*rd*rd*ht);
}
}

class cone extends shape
{
    cone(double r,double h)
    {
        super(r,h);
    }

    double area() // Area Of cone
    {
        return(pi*rd*ht);
    }

    double volume() //Volume Of cone
    {
        return(pi*rd*rd*ht);
    }
}

public class ShapeDemo
{
    public static void main(String args[])
    {
        System.out.println("-----");
        System.out.println(" Area  & Volume of Sphere & Cylinder & Cone");
        cylinder cy=new cylinder(5.00,6.00);
        cylinder s2;
        s2=cy;
        System.out.println("-----");
        System.out.println(" Area Of cylinder : "+s2.area());
        System.out.println(" Volume Of cylinder : "+s2.volume());
        cone c=new cone(4.00,5.00);
        cone s3;
        s3=c;
        System.out.println("-----");
        System.out.println(" Area Of cone: "+s3.area());
        System.out.println(" Volume Of cone : "+s3.volume());
        System.out.println("-----");
    }
}
```

}

16. Write 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.



```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
public class moveCell extends JFrame implements ActionListener
{
    JButton moveBtns[]=new JButton[4];
    JPanel gridPanels[]=new JPanel[16];
    String icons[]={"Left","Up","Right","Down"};
    int cnt=1;
    public moveCell()
    {
        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 moveCell().show();
}
}

```

- 17. 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 TextFields.**

```

import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
public class Login extends JFrame implements ActionListener
{
    public static int cnt;
    JButton bclear,bcancel,blog;
    JTextField tf2;
    JPasswordField pf;
    JLabel lb1,lb2,lb3,hn,x,y;

    public Login()
    {
        Container cp=getContentPane();
        cp.setLayout(null);
        cp.setBackground(Color.white);
        lb1=new JLabel("LOGIN FORM");
        lb1.setBounds(200,50,150,20);
        cp.add(lb1);
        lb2=new JLabel("Enter User Name :");
        lb2.setBounds(20,70,200,20);
        lb3=new JLabel("Enter Password :");
        lb3.setBounds(20,100,100,20);
    }
}

```

```
tf2=new JTextField();
tf2.setBounds(140,70,150,30);

pf=new JPasswordField(12);
pf.setBounds(140,100,150,30);

blog=new JButton("Login");
blog.setBounds(80,150,120,25);

bclear=new JButton("Clear");
bclear.setBounds(200,150,120,25);

bcancel=new JButton("Exit");
bcancel.setBounds(310,150,120,25);

cp.add(lb2);
cp.add(tf2);
cp.add(lb3);
cp.add(pf);
cp.add(blog);
cp.add(bclear);
cp.add(bcancel);

setSize(500,300);
setTitle("Login");
setVisible(true);

setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

blog.addActionListener(this);
bclear.addActionListener(this);
bcancel.addActionListener(this);
}
public void actionPerformed(ActionEvent ae)
{
    if(ae.getSource()==blog)
    {
        String id=tf2.getText();
        String pwd=pf.getText();
        int no;
        no=id.compareTo(pwd);
        if((id.equals("")) || (pwd.equals("")))
        {
            JOptionPane.showMessageDialog((Component) null,"Please Enter
            username or password...!", "",JOptionPane.OK_CANCEL_OPTION);
        }
        else if(no==0)
        {
            JOptionPane.showMessageDialog(this,"Login Successfully");
            System.exit(0);
        }
        else
        {

```

```

        JOptionPane.showMessageDialog(this, cnt+":Fail to Login you have 3
        attempt:");
        tf2.setText("");
        pf.setText("");
        cnt++;
        if(cnt==3)
        {
            JOptionPane.showMessageDialog(this, "! your time limit over you
            can not login:");
            System.exit(0);
        }
    }
}
if(ae.getSource()==bclear)
{
    tf2.setText("");
    pf.setText("");
}
if(ae.getSource()==bcancel)
{
    System.exit(0);
    setVisible(false);
}
}
public static void main(String args[])
{
    Login obj=new Login();
}
}

```

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

```

import java.io.*;
class FileCount
{
    public static void main(String args[]) throws Exception
    {
        int ccnt=0,lcnt=1,wcnt=0,c;
        FileInputStream fin=new FileInputStream("a.txt");

        while((c=fin.read())!=-1)
        {
            System.out.println(c);

            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);
}
}

```

19. Write a java program to accept a number from the user, if number is zero then throw user defined Exception “Number is 0” otherwise calculate the sum of first and last digit of a given number (Use static keyword).

```

import java.io.*;
class ZeroNum extends Exception
{
    ZeroNum(String msg)
    {
        super(msg);
    }
}

public class ExceptionDemo
{
    public static int no;    // Use of Static keyword.
    int j,k,i,flag=0;
    DataInputStream din=new DataInputStream(System.in);

    public void Num()
    {
        try
        {
            System.out.print("Enter the Number : ");
            no=Integer.parseInt(din.readLine());

            if(no==0)
            {
                throw new ZeroNum("Exception!!!! Number is 0");
                //throws userdefined Exception.
            }
            else
            {
                for(i=1;i<no;i++)
                {
                    if(flag==1)
                    {
                        no=no/10;
                        j=no%10;
                    }
                    else
                    {
                        k=no%10;
                        flag=1;
                    }
                }
                //Closing of For loop
                System.out.println("Addition="+(j+k));
            }
        }
    }
}

```

```

        } // Closing of Try Block.
        catch(ZeroNum z)
        {
            System.out.println(z.getMessage());
        }
        catch(Exception e)
        {
            System.out.println(e.getMessage());
        }
    } //Closing of Num function.

    public static void main(String args[])
    {
        ExceptionDemo n1=new ExceptionDemo();
        n1.Num();
    }
}

```

20. Write a package for Games in Java, which have two classes Indoor and Outdoor. Use a function display () to generate the list of players for the specific games. (Use Parameterized constructor, finalize() method and Array Of Objects)

```

import Games.*;
class PackageDemo
{
    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();
    }
}

```

Game Package

Indoor.java

```

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...");
}
}
```

Outdoor.java

```
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...");
    }
}
```

21. Write a Java program to design a screen using Swing that will create four TextFields. First for the text, second for what to find and third for replace. Display result in the fourth TextField. Display the count of total no. of replacements made. The button clear to clear the TextFields.

The image shows a Java Swing window titled "Find and Replace". Inside the window, there are four text input fields arranged vertically. To the left of each field is a label: "Enter Text", "Text To Find", "Text to Replace", and "No. Of occurrences". Below these fields are three buttons: "Find", "Replace", and "Clear".

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

public class FindReplace 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 FindReplace()
    {
        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 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);
        }
    }

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

```

22. Define an Interface Shape with abstract method area(). Write a java program to calculate an area of Circle and Sphere.(use final keyword)

```

interface Shape
{
    abstract void area();
}

class Cir implements Shape
{
    float r,a,v;
    Cir(float ra)

```

```
    {  
        r=ra;  
    }  
    public final void area()  
    {  
        a=3.14f*r*r;  
        System.out.println("Area Of Circle is \t" + a);  
    }  
}
```

```
class Sphere implements Shape  
{  
    float rad,a;  
    Sphere(float r)  
    {  
        rad=r;  
    }  
    public final void area()  
    {  
        a=4*3.14f*rad;  
        System.out.println("Area Of Sphere is \t" + a);  
    }  
}
```

```
class InterfaceDemo  
{  
    public static void main(String args[])  
    {  
        Cir c=new Cir(5.0f);  
        c.area();  
  
        Sphere r=new Sphere(5.0f);  
        r.area();  
    }  
}
```

- 23. Write a java program to accept the details of n Cricket Players from user (Player code, name, runs, innings- played and number of times not out). The program should contain following menus:**
- Display average runs of a single player.**
 - Display average runs of all players. (Use array of objects, Method overloading and static keyword)**

```

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 PlayerDemo
{
    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);
    }
}
}

```

24. Write a Java Program to accept the details of Employee(Eno, EName,Sal) from the user and display it on the next Frame. (Use AWT)

```

import java.awt.*;
import java.awt.event.*;
class nextFrame extends Frame
{
    Label l1,l2,l4;
    TextField tf1,tf2,tf4;
    nextFrame nf1;
    public nextFrame(int no1,String name1,int sal1)
    {
        setLayout(new GridLayout(4,2));
        l1=new Label("Emp No      : ");
        l2=new Label("Emp Name    : ");
        l4=new Label("Emp Salary  : ");
        tf1=new TextField(14);
        tf2=new TextField(14);
        tf4=new TextField(14);
    }
}

```

```
        add(l1);
        add(tf1);

        add(l2);
        add(tf2);

        add(l4);
        add(tf4);

        tf1.setText(String.valueOf(no1));
        tf2.setText(name1);
        tf4.setText(String.valueOf(sal1));
    }
}

public class Employee extends Frame implements ActionListener
{
    Frame f;
    private Label l1,l2,l4;
    private Button b1,b2;
    private TextField tf1,tf2,tf4;
    static Employee e1;
    public Employee()
    {
        setLayout(new GridLayout(5,2));
        l1=new Label("Enter Emp No           : ");
        l2=new Label("Enter Emp Name           : ");
        l4=new Label("Enter Emp Salary           : ");
        b1=new Button("Display");
        b2=new Button("Exit");
        tf1=new TextField(14);
        tf2=new TextField(14);
        tf4=new TextField(14);

        add(l1);
        add(tf1);

        add(l2);
        add(tf2);

        add(l4);
        add(tf4);

        add(b1);
        add(b2);

        b1.addActionListener(this);
        b2.addActionListener(this);
    }

    public void actionPerformed(ActionEvent e)
    {
        int no,sal;
        String name,address;
```

```

        if(e.getSource()==b1)
        {
            no=Integer.parseInt(tf1.getText());
            name=tf2.getText();
            sal=Integer.parseInt(tf4.getText());
            f = new nextFrame(no,name,sal);
            f.setTitle("Employee Detail");
            f.setSize(250, 250);
            f.setVisible(true);
        }
        else if(e.getSource()==b2)
        {
            System.exit(0);
        }
    }
    public static void main(String args[])
    {
        e1=new Employee();
        e1.setTitle("Employee Information");
        e1.setSize(300,300);
        e1.setVisible(true);
    }
}

```

25. 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.

```

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(enno + " "+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 EmployeeManager
{
    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);
    }
}

```

26. 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.

```

class PrimePerfectDemo
{
    public static void main(String args[])
    {
        int i,j;
        int no[]=new int[10];
        int prime[]=new int[10];
        int perfect[]=new int[10];
        try
        {
            for(i=0;i<5;i++)
            {
                no[i]=Integer.parseInt(args[i]);
            }
            System.out.println("-----Prime Number-----");
            for(i=0;i<5;i++)
            {
                int f=0;
                for(j=2;j<no[i];j++)
                {
                    if(no[i]%j==0)
                    {
                        f=1;
                        break;
                    }
                }
            }
        }
    }
}

```

```

        if(f==0)
        {
            prime[i]=no[i];
            System.out.println(prime[i]);
        }
    }
    System.out.println("-----Perfect Number-----");
    for(i=0;i<5;i++)
    {
        int s=0;
        for(j=1;j<no[i];j++)
        {
            if(no[i]%j==0)
            {
                s=s+j;
            }
        }
        if(no[i]==s)
        {
            perfect[i]=no[i];
            System.out.println(perfect[i]);
        }
    }
}

catch(Exception e) {}
}
}

```

27. .Write a java program to accept the details employee (Eno, Ename ,Salary) and add it into the JTable by clicking on the Button.(Max : 5 Records)

```

import java.io.*;
import javax.swing.*;
import javax.swing.event.*;
import java.awt.*;
import java.awt.event.*;
import javax.swing.table.DefaultTableModel;
public class Employee extends JFrame implements ActionListener
{
    private JPanel jpInfo = new JPanel();
    private JLabel lbID,lbName,lbSal;
    private JTextField txt1,txt2,txt3;
    DefaultTableModel model;
    JTable etab;
    int r=0;
    private JButton btnSubmit;
    Employee()
    {
        jpInfo.setBounds (0, 0, 500, 115);
        jpInfo.setLayout (null);
        lbID = new JLabel ("EMP ID: ");
        lbID.setForeground (Color.black);
        lbID.setBounds (15, 55, 80, 25);
        txt1 = new JTextField ();
        txt1.setBounds (70,55,80,25);
    }
}

```

```
        lbName = new JLabel ("EMP NAME: ");
        lbName.setForeground (Color.black);
        lbName.setBounds (200, 55, 100, 25);
        txt2 = new JTextField ();
        txt2.setBounds (270,55,80,25);

        lbSal = new JLabel ("EMP SALARY: ");
        lbSal.setForeground (Color.black);
        lbSal.setBounds (390, 55, 120, 25);
        txt3 = new JTextField ();
        txt3.setBounds (480,55,80,25);

        //Aligning The Buttons.
        btnSubmit = new JButton ("Add");
        btnSubmit.setBounds (600, 55, 80, 25);
        btnSubmit.addActionListener(this);

        //Adding the All the Controls to Panel.
        jpInfo.add (lbID);
        jpInfo.add (lbName);
        jpInfo.add (lbSal);
        jpInfo.add (txt1);
        jpInfo.add (txt2);
        jpInfo.add (txt3);
        jpInfo.add (btnSubmit);
        model = new DefaultTableModel();
        model.addColumn("EMP ID");
        model.addColumn("EMP NAME");
        model.addColumn("EMP SAL");
        etab = new JTable(model);
        JScrollPane jp = new JScrollPane(etab ,
        ScrollPaneConstants.VERTICAL_SCROLLBAR_AS_NEEDED,ScrollPaneConstants.HORIZONTAL_SCROLLBAR_AS_NEEDED);
        jp.setBounds(120,200,550,130);
        jpInfo.add(jp);

        //Adding Panel to Window.
        getContentPane().add (jpInfo);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        //In the End Showing the New Account Window.
        setVisible (true);
        setSize(720,500);
        setTitle("Employee Informaction");

    }

    @Override
    public void actionPerformed(ActionEvent ae)
    {

        if(ae.getSource()==btnSubmit)
        {
            try
```

```

        {
            int rt = Integer.parseInt(txt1.getText());
            String un = txt2.getText();
            double to = Double.parseDouble(txt3.getText());
            model.insertRow(r++,new Object[]{new Integer(rt),new
            String(un),new Double(to) });

            txt1.setText("");
            txt2.setText(" ");
            txt3.setText(" ");
        }
        catch(Exception e1)
        {
            Toolkit.getDefaultToolkit().beep();
            JOptionPane.showMessageDialog(null,e1.toString(),
            "Confirmation",JOptionPane.ERROR_MESSAGE );
        }
    }

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

```

28. Write a java program to read n Students names from user, store them into the ArrayList collection. The program should not allow duplicate names. Display the names in Ascending order.

```

import java.io.*;
import java.util.*;
class ArrayListDemo
{
    public static void main(String args[])
    {
        int no,i,j;
        String name;
        ArrayList obj=new ArrayList();
        BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
        try
        {
            System.out.println("Enter Number of Student:-");
            no=Integer.parseInt(br.readLine());
            String str[]=new String[no];
            for(i=0;i<no;i++)
            {
                System.out.println("Enter student Name:-" + (i+1));
            }
        }
        catch (Exception e)
        {
            System.out.println("Error: " + e.getMessage());
        }
    }
}

```

```

        name=br.readLine();
        if(obj.contains(name))
        {
            System.out.println("Element Allready Inserted!!!\n
            Reenter Element");
            i--;
        }
        else
        {
            obj.add(name);
        }
    }
    obj.toArray(str);
    for(i=0;i<no;i++)
    {
        for(j=i+1;j<no;j++)
        {
            if((str[i].compareTo(str[j]))>0)
            {
                String temp=str[i];
                str[i]=str[j];
                str[j]=temp;
            }
        }
    }
    System.out.println("Sorted Element:");
    for(i=0;i<no;i++)
    System.out.println(str[i]);
}

catch(IOException e){}

}
}

```

29. Write a java program to accept n employee names from user, store them into the LinkedList class and display them by using.

a. Iterator Interface

b. ListIterator Interface

```

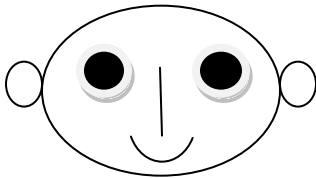
import java.io.*;
import java.util.*;
class LinkedListDemo
{
    public static void main(String args[])
    {
        int no,i;
        String name;
        LinkedList obj=new LinkedList();
        BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
        try
        {
            System.out.println("Enter Number of Employee:-");
            no=Integer.parseInt(br.readLine());

```

```
        for(i=0;i<no;i++)
        {
            System.out.println("Enter Employee Name:-" + (i+1));
            name=br.readLine();
            obj.add(name);
        }

        System.out.println("\nUsing Iterator");
        Iterator itr=obj.iterator();
        while(itr.hasNext())
        {
            System.out.println(itr.next());
        }
        System.out.println("\nUsing List Iterator");
        ListIterator it=obj.listIterator();
        while(it.hasNext())
        {
            System.out.println(it.next());
        }
    }
    catch(IOException e){}
}
}
```

30 Write an applet application in Java for smile face.



```
import java.applet.Applet;
import java.awt.Color;
import java.awt.Graphics;
public class DrawFace extends Applet
{
    public void paint(Graphics g)
    {
        g.drawOval(130,200,150,130);
        g.fillOval(150,250,20,20);
        g.fillOval(230,250,20,20);
        g.drawLine(200,235,200,290);
        g.drawOval(100,250,30,30);
        g.drawOval(280,250,30,30);
        g.drawArc(175,260,50,50,180,180);
    }
}

<html>
<applet code="DrawFace" width=500 height=500></applet>
</html>
```