Construct a linked List containing names of colours: red, blue, yellow and orange. Then extend the program to do the following: i. Display the contents of the List using an Iterator ii. Display the contents of the List in reverse order using a ListIterator iii. Create another list containing pink and green. Insert the elements of this list between blue and yellow.

import java.util.\*;

class Slip14\_2

{

            public static void main(String[] args)

            {

                        LinkedList ll=new LinkedList();

                        ll.add("Red");

                        ll.add("Blue");

                        ll.add("Yellow");

                        ll.add("Orange");

                        Iterator i=ll.iterator();

                        System.out.println("\ncontents of the List using an Iterator:\n");

                        while(i.hasNext())

                        {

                                    String s=(String)i.next();

                                    System.out.println(s);

                        }

                        ListIterator li = ll.listIterator();

                        while(li.hasNext())

                        {

                                    // String elt = (String)

                                    li.next();

                        }

                        System.out.println("\ncontents of the List in reverse order using a ListIterator : ");

                        while(li.hasPrevious())

                        {

                                    System.out.println(li.previous());

                        }

                        ll.add(2,"Pink");           //add element at second position

                        ll.add(3,"Green");        //add element at 3rd position

                        System.out.println("\nlist between blue and yellow is:");

                        System.out.println(ll);

            }

}

**Write a java program to accept 'n' integers from the user and store them in an ArrayList Collection.**

**Display the elements of an ArrayList in Reverse order.**

import java.util.\*;

class array

{

public static void main(String a[])

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter Limit of ArrayList :");

int n=sc.nextInt();

ArrayList alist=new ArrayList();

System.out.println("Enter Elements of ArrayList :");

for(int i=0;i<n;i++)

{

String elmt=sc.next();

alist.add(elmt);

}

System.out.println("Original ArrayList is :"+alist);

Collections.reverse(alist);

System.out.println("Reverse of a ArrayList is :"+alist);

}

}

3) copy from one file to another

1. **import** java.io.\*;
2. **import** java.util.\*;
3. **class** Copyfile {
4. **public** **static** **void** main(String arg[]) **throws** Exception {
5. Scanner sc = **new** Scanner(System.in);
6. System.out.print("Provide source file name :");
7. String sfile = sc.next();
8. System.out.print("Provide destination file name :");
9. String dfile = sc.next();
10. FileReader fin = **new** FileReader(sfile);
11. FileWriter fout = **new** FileWriter(dfile, **true**);
12. **int** c;
13. **while** ((c = fin.read()) != -1) {
14. fout.write(c);
15. }
16. System.out.println("Copy finish...");
17. fin.close();
18. fout.close();
19. }
20. }

Write a program to accept the user name and greets the user by name. Before displaying the user's name, convert it to upper case letters. For example, if the user's name is Raj, then display greet message as "Hello, RAJ, nice to meet you!".

import java.util.Scanner;

public class GreetingWithScanner {

public static void main(String[] args) {

Scanner stdin = new Scanner( System.in );

String usersName; // The user's name, as entered by the user.

String upperCaseName; // The user's name, converted to upper case letters.

System.out.print("Please enter your name: ");

usersName = stdin.nextLine();

upperCaseName = usersName.toUpperCase();

System.out.println("Hello, " + upperCaseName + ", nice to meet you!");

} // end main()

} // end class

Write a program to design following GUI using swing. On click of Show button display the selected Programming language on screen using label.

**import** javax.swing.\*;

**public** **class** Example{

JFrame a;

Example(){

a = **new** JFrame("example");

string courses[] = { "core java","advance java", "java servlet"};

JComboBox c = **new** JComboBox(courses);

c.setBounds(40,40,90,20);

a.add(c);

a.setSize(400,400);

a.setLayout(**null**);

a.setVisible(**true**);

}

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

{

**new** Example();

}

}

Accept “n” integers from the user and store them into the collection. Display them in the sorted order. The collection should not accept duplicate elements (Use suitable collection). Search for the particular element using predefined search method in the collection framework.

import java.util.\*;

import java.io.\*;

class Slip19\_2

{

            public static void main(String[] args) throws Exception

            {

                        int no,element,i;

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

                                    TreeSet ts=new TreeSet();

                                    System.out.println("Enter the of elements :");

                                    no=Integer.parseInt(br.readLine());

                                    for(i=0;i<no;i++)

                                    {

                                                System.out.println("Enter the element : ");

                                                            element=Integer.parseInt(br.readLine());

                                                            ts.add(element);

                                    }

                                    System.out.println("The elements in sorted order :"+ts);

                        System.out.println("Enter element to be serach : ");

                        element = Integer.parseInt(br.readLine());

                        if(ts.contains(element))

                                    System.out.println("Element is found");

                        else

                                    System.out.println("Element is NOT found");

            }

}

Write a program to create the following GUI using Swing components.

import java.awt.event.\*;

import java.awt.\*;

import javax.swing.\*;

public class Calculator extends JFrame implements ActionListener

{

   JButton b10,b11,b12,b13,b14,b15;

   JButton b[]=new JButton[10];

    int i,r,n1,n2;

    JTextField res;

    char op;

   public Calculator()

  {

     super("Calulator");

      setLayout(new BorderLayout());

      JPanel p=new JPanel();

      p.setLayout(new GridLayout(4,4));

      for(int i=0;i<=9;i++)

      {

        b[i]=new JButton(i+"");

        p.add(b[i]);

        b[i].addActionListener(this);

      }

      b10=new JButton("+");

      p.add(b10);

      b10.addActionListener(this);

      b11=new JButton("-");

      p.add(b11);

      b11.addActionListener(this);

      b12=new JButton("\*");

      p.add(b12);

      b12.addActionListener(this);

      b13=new JButton("/");

      p.add(b13);

      b13.addActionListener(this);

      b14=new JButton("=");

      p.add(b14);

      b14.addActionListener(this);

      b15=new JButton("C");

      p.add(b15);

      b15.addActionListener(this);

      res=new JTextField(10);

      add(p,BorderLayout.CENTER);

      add(res,BorderLayout.NORTH);

      setVisible(true);

      setSize(200,200);

     }

public void actionPerformed(ActionEvent ae)

{

  JButton pb=(JButton)ae.getSource();

if(pb==b15)

{

r=n1=n2=0;

res.setText("");

}

else

if(pb==b14)

{

n2=Integer.parseInt(res.getText());

   eval();

   res.setText(""+r);

}

else

{

  boolean opf=false;

  if(pb==b10)

{ op='+';

  opf=true;

}

  if(pb==b11)

{ op='-';opf=true;}

  if(pb==b12)

{ op='\*';opf=true;}

  if(pb==b13)

{ op='/';opf=true;}

  if(opf==false)

  {

      for(i=0;i<10;i++)

   {

   if(pb==b[i])

        {

              String t=res.getText();

           t+=i;

             res.setText(t);

   }

   }

  }

  else

  {

     n1=Integer.parseInt(res.getText());

     res.setText("");

  }

}

}

int eval()

{

switch(op)

{

case '+':   r=n1+n2;  break;

case '-':    r=n1-n2;   break;

case '\*':    r=n1\*n2; break;

case '/':    r=n1/n2; break;

}

return 0;

}

  public static void main(String arg[])

  {

      new Calculator();

   }

}

Create a hash table containing student name and percentage. Display the details of hash table. Also search for the specific student and display percentage of that student.

**/\* Slip no : 15\_2\*/**

import java.io.\*;

import java.util.\*;

class Slip15\_2

{

            public static void main(String args[]) throws Exception

            {

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

                                    Hashtable ht=new Hashtable();

                                    float per;

                                    String name=null;

                        System.out.println("\n Enter no of Students : ");

                        int n=Integer.parseInt(br.readLine());

                        for(int i=0;i<n;i++)

                        {

                                    System.out.print("Enter Student name :");

                                                name=br.readLine();

                                                System.out.print("Enter Student's per :");

                                                per = Float.parseFloat(br.readLine());

                                                ht.put(name,per);

                        }

                        System.out.println("Hash table = "+ht);

                        Enumeration k = ht.keys();

                        Enumeration v= ht.elements();

                                    System.out.println("ID\tName");

                        while(k.hasMoreElements())

                        {

                                    System.out.println(k.nextElement()+"\t"+v.nextElement());

                        }

                        System.out.println("Enter name to be search : ");

                        String snm=br.readLine();

                        k = ht.keys();

                        v= ht.elements();

                        int cnt=0;

                                    while(k.hasMoreElements())

                                    {

                                                name = (String)k.nextElement();

                                                if(snm.equals(name))

                                                {           cnt++;

                                                            System.out.println("Percentage = "+v.nextElement());

                                                }

                                    }

                        if(cnt==0)

                                    System.out.println("Student not found");

            }

}

Write a program that displays the number of characters, lines and words of a file. [20 Mark

**import** java.io.\*;

**public** **class** Test {

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

**throws** IOException

    {

        File file = **new** File("C:\\Users\\hp\\Desktop\\TextReader.txt");

        FileInputStream fileInputStream = **new** FileInputStream(file);

        InputStreamReader inputStreamReader = **new** InputStreamReader(fileInputStream);

        BufferedReader bufferedReader = **new** BufferedReader(inputStreamReader);

        String line;

**int** wordCount = 0;

**int** characterCount = 0;

**int** paraCount = 0;

**int** whiteSpaceCount = 0;

**int** sentenceCount = 0;

**while** ((line = bufferedReader.readLine()) != **null**) {

**if** (line.equals("")) {

                paraCount += 1;

            }

**else** {

                characterCount += line.length();

                String words[] = line.split("\\s+");

                wordCount += words.length;

                whiteSpaceCount += wordCount - 1;

                String sentence[] = line.split("[!?.:]+");

                sentenceCount += sentence.length;

            }

        }

**if** (sentenceCount >= 1) {

            paraCount++;

        }

        System.out.println("Total word count = "+ wordCount);

        System.out.println("Total number of sentences = "+ sentenceCount);

        System.out.println("Total number of characters = "+ characterCount);

        System.out.println("Number of paragraphs = "+ paraCount);

        System.out.println("Total number of whitespaces = "+ whiteSpaceCount);

    }

}