ChannabasaveshwaraInstituteofTechnology

(AffiliatedtoVTU,Belgaum&ApprovedbyAICTE,NewDelhi**)**

(NAACAccredited&ISO9001:2015CertifiedInstitution)

NH206(B.H.Road),Gubbi,Tumkur –572216. Karnataka

Advance java programming lab manual

1. Implementajavaprogramtodemonstratecreatinganarraylist,addingelements,removing elements, sorting elements of ArrayList. Also illustrate the use of toArray() method.

```
import java.util.ArrayList;

importjava.util.Collections;


  publicclassArrayListDemo{

   publicstaticvoidmain(String[]args){

        //CreatinganArrayList

      ArrayList<Integer>numbers=newArrayList<>();


      //AddingelementstotheArrayList

      numbers.add(5);

      numbers.add(2);

      numbers.add(8);

      numbers.add(3);

      numbers.add(1);


      // Displaying the ArrayList

      System.out.println("OriginalArrayList:"+numbers);


      //Removinganelement

      numbers.remove(2);


      //DisplayingtheArrayListafter removal

      System.out.println("ArrayListafterremovingelementatindex2:"+numbers);
```

```
        // Sorting the ArrayList
        Collections.sort(numbers);

        // Displaying the ArrayList after sorting
        System.out.println("ArrayListaftersorting:"+numbers);

        //UsingtoArray()methodtoconvertArrayListtoarray
        Integer[] array = numbers.toArray(new Integer[0]);

        //Displayingthearray
        System.out.println("ArrayconvertedfromArrayList:"); for
        (int i = 0; i < array.length; i++) {
            System.out.print(array[i]+"");
        }
        System.out.println();
    }
}
```

OUTPUT:

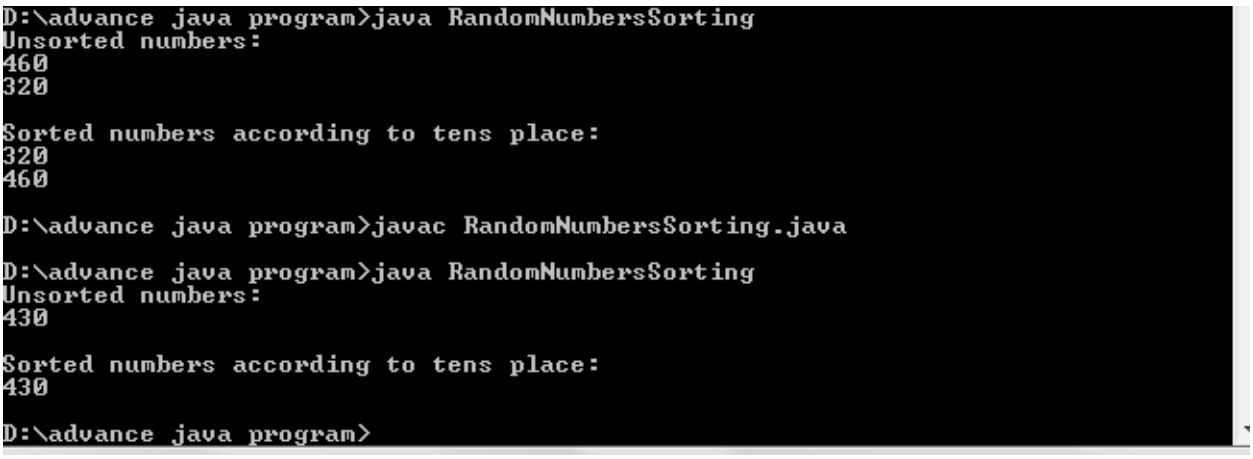
```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

D:\advance java program>javac ArrayListDemo.java

D:\advance java program>java ArrayListDemo
Original ArrayList: [5, 2, 8, 3, 1]
ArrayList after removing element at index 2: [5, 2, 3, 1]
ArrayList after sorting: [1, 2, 3, 5]
Array converted from ArrayList:
1 2 3 5

D:\advance java program>
```

2. Developaprogramtoreadrandomnumbersbetweenagivenrange thataremultiplesof2and5,sort the numbers according to tens place using comparator.

```
import java.util.ArrayList;

importjava.util.Comparator;

import java.util.Random;


classTensPlaceComparatorimplementsComparator<Integer>{
   @Override
   publicintcompare(Integernum1,Integernum2){ int
      tensPlace1 = num1 % 100 / 10;
      inttensPlace2=num2%100/10; return
      tensPlace1 - tensPlace2;
   }
}


public class RandomNumbersSorting {
   publicstaticvoidmain(String[]args){
      int minRange = 100;
      intmaxRange=999;
      intcount=10;//Numberofrandomnumbersto generate


      ArrayList<Integer>randomNumbers=newArrayList<>();


      //Generaterandomnumberswithinthegivenrangeandaddmultiplesof 2and5to

the list
```

```
        Randomrand=newRandom();
        for (int i = 0; i < count; i++) {
            intrandomNumber=rand.nextInt((maxRange-minRange)+1)+minRange; if
            (randomNumber % 2 == 0 && randomNumber % 5 == 0) {
                randomNumbers.add(randomNumber);
            }
        }

        // Displaying the unsorted list
        System.out.println("Unsortednumbers:");
        for (Integer num : randomNumbers) {
            System.out.println(num);
        }

        // Sorting the numbers according to their tens place
        randomNumbers.sort(newTensPlaceComparator());

        //Displayingthesortedlist
        System.out.println("\nSortednumbersaccordingtotensplace:"); for
        (Integer num : randomNumbers) {
            System.out.println(num);
        }
    }
}
```

OUTPUT:



3. Implementajavaprogramtoillustratestoringuserdefinedclassesincollection.

```
importjava.util.ArrayList;


classPerson{
   privateStringname;
   private int age;


   publicPerson(Stringname,intage){
      this.name = name;
      this.age= age;
   }


   publicStringgetName(){ return
      name;
   }
```

```
    publicintgetAge(){
        return age;
    }

    @Override
    publicStringtoString(){
        return"Name:"+name+",Age:"+ age;
    }
}

publicclassUserDefinedClassInCollection{
    public static void main(String[] args) {
        // Creating an ArrayList to store Person objects
        ArrayList<Person>people=newArrayList<>();

        //AddingPersonobjectstotheArrayList people.add(new
        Person("Alice", 30));
        people.add(newPerson("Bob", 25));
        people.add(newPerson("Charlie", 40));

        // Displaying the list of people
        System.out.println("Listofpeople:");
        for (Person person : people) {
            System.out.println(person);
        }
```

```
    }
}
```

OUTPUT:

4. Implementajavaprogramtoillustratetheuseofdifferenttypesofstringclassconstructors

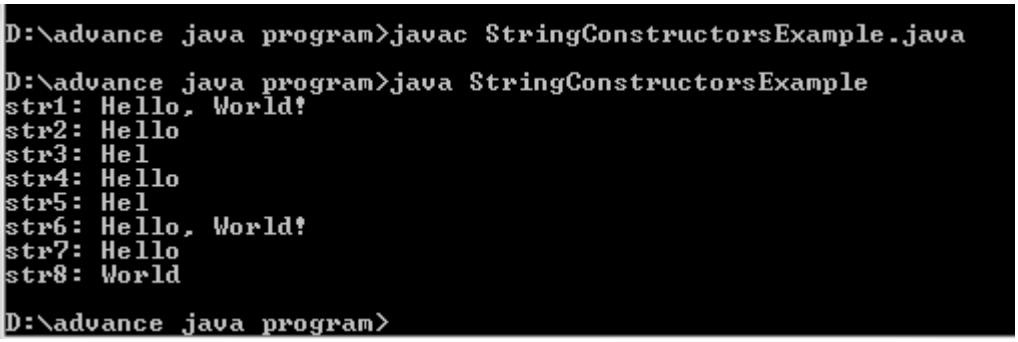
```
publicclassStringConstructorsExample{
    public static void main(String[] args) {
        //Usingdifferentconstructorstocreate Stringobjects

        //1. Creating a Stringusing astring literal

 Stringstr1="Hello, World!";

        //2.CreatingaStringfromacharacterarray
        char[] charArray = {'H', 'e', 'l', 'l', 'o'};
        String str2 = new String(charArray);

        //3.CreatingaStringfromaportionofacharacterarray String
        str3 = new String(charArray, 0, 3); // "Hel"

        //4.Creating aStringfromabytearray
        byte[]byteArray ={72,101, 108, 108,111}; // ASCIIvaluesfor"Hello"
```

```
Stringstr4=new String(byteArray);

//5.CreatingaStringfromaportionofabytearray String
str5 = new String(byteArray, 0, 3); // "Hel"

//6.CreatingaStringfromanotherStringobject
String str6 = new String(str1);

//7. CreatingaString usingStringBuilder
StringBuilderstringBuilder=newStringBuilder("Hello");
String str7 = new String(stringBuilder);

//8. Creating aString usingStringBuffer
StringBufferstringBuffer=newStringBuffer("World");
String str8 = new String(stringBuffer);

// Displaying the created Strings
System.out.println("str1:"+str1);
System.out.println("str2:"+str2);
System.out.println("str3:"+str3);
System.out.println("str4:"+str4);
System.out.println("str5:"+str5);
System.out.println("str6:"+str6);
System.out.println("str7:"+str7);
System.out.println("str8:"+ str8);
```

}

}

OOUTPUT:



5. Implementajavaprogramtoillustratetheuseofdifferenttypesofcharacterextraction,string comparison, string search and string modification methods.

public class StringMethodsExample {

publicstaticvoidmain(String[]args){

// String for demonstration

Stringstr="Hello,World!";

//Character extractionmethods

charfirstChar=str.charAt(0);//Extractsthefirstcharacter('H')

charlastChar=str.charAt(str.length()-1);//Extractsthelastcharacter

('!')

Stringsubstring=str.substring(7, 12);//Extractssubstringfromindex7to11

("World")

//Stringcomparisonmethods

booleanisEqual=str.equals("Hello,World!");//Comparesexactcontent

```
boolean isIgnoreCaseEqual = str.equalsIgnoreCase("HELLO, WORLD!"); // Compares

ignoringcase

//Stringsearchmethods

intindexOfComma=str.indexOf(',');//Findstheindexof thefirst occurrence

of','

intlastIndexOfL=str.lastIndexOf('l');//Findstheindexofthelast

occurrenceof'l'

booleancontainsWorld =str.contains("World");//Checksif thestringcontains

"World"

//Stringmodificationmethods

StringreplacedStr=str.replace('o','0');//Replaces'o'with'0'

StringupperCaseStr=str.toUpperCase();//Convertsthestringtouppercase

String lowerCaseStr = str.toLowerCase(); // Converts the string to lowercase

String trimmedStr = str.trim(); // Removes leading and trailing whitespaces

//Displayingtheresults

System.out.println("Firstcharacter:"+firstChar);

System.out.println("Last character: " + lastChar);

System.out.println("Substring: " + substring);

System.out.println("Is equal? " + isEqual);
```

```
        System.out.println("Isignorecaseequal?"+isIgnoreCaseEqual);
        System.out.println("Index of comma: " + indexOfComma);
        System.out.println("Last index of 'l': " + lastIndexOfL);
        System.out.println("Contains 'World'? " + containsWorld);
        System.out.println("Replaced string: " + replacedStr);
        System.out.println("Uppercase string: " + upperCaseStr);
        System.out.println("Lowercase string: " + lowerCaseStr);
        System.out.println("Trimmed string: '" + trimmedStr + "'");
    }
}
```

OUTPUT:

```
D:\advance java program>javac StringMethodsExample.java

D:\advance java program>java StringMethodsExample
First character: H
Last character: !
Substring: World
Is equal? true
Is ignore case equal? true
Index of comma: 5
Last index of 'l': 10
Contains 'World'? true
Replaced string: Hell0, W0rld!
Uppercase string: HELLO, WORLD!
Lowercase string: hello, world!
Trimmed string: 'Hello, World!'

D:\advance java program>
```

6. Implementajavaprogramtoillustratetheuseofdifferenttypesof StringBuffer methods.

```
publicclassStringBufferMethodsExample{ public
    static void main(String[] args) {
  //CreatingaStringBufferobject
  StringBufferstringBuffer=new StringBuffer("Hello");

   //Appendingcharactersorstrings
   stringBuffer.append(",World!");

  //Insertingcharactersorstringsatspecificpositions
  stringBuffer.insert(5, "Java ");

  // Deleting characters from a StringBuffer
  stringBuffer.delete(5,10);//Deletes"Java"

  //Replacingcharactersor strings
  stringBuffer.replace(5,6,"Universe");//Replaces"W"with"Universe"

  //ReversingthecontentoftheStringBuffer
  stringBuffer.reverse();

  //GettingthelengthoftheStringBuffer int
  length = stringBuffer.length();

  //Gettingthecapacityof theStringBuffer
```

```
        intcapacity= stringBuffer.capacity();


        // Setting the length of the StringBuffer
        stringBuffer.setLength(7);//Truncatesthestringto"olleh,"


        //ConvertingStringBuffertoString
        Stringstr= stringBuffer.toString();


        // Displaying themodified StringBufferand otherinformation
        System.out.println("ModifiedStringBuffer:"+stringBuffer);
        System.out.println("Length: " + length);
        System.out.println("Capacity: " + capacity);
        System.out.println("String representation: " + str);
    }
}
```

OUTPUT:

```
D:\advance java program>javac StringBufferMethodsExample.java

D:\advance java program>java StringBufferMethodsExample
Modified StringBuffer: !dlroW
Length: 20
Capacity: 21
String representation: !dlroW

D:\advance java program>
```

7. Demonstrate a swing event handling application that creates 2 buttons Alpha and Beta and displays the text “Alpha pressed” when alpha button is clicked and “Beta pressed” when beta button is clicked.

```
importjavax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
importjava.awt.event.ActionListener;

publicclassSwingEventHandlingDemoextendsJFrameimplementsActionListener{
   private JButton alphaButton;
   privateJButtonbetaButton;

   public SwingEventHandlingDemo() {
      setTitle("Swing Event Handling Demo");
      setSize(300, 150);
      setDefaultCloseOperation(EXIT_ON_CLOSE);

      //Creatingbuttons
      alphaButton=newJButton("Alpha"); betaButton
      = new JButton("Beta");

      // Setting action command for buttons
      alphaButton.setActionCommand("Alpha");
      betaButton.setActionCommand("Beta");

      //Addingactionlistenertobuttons
```

```
    alphaButton.addActionListener(this);
    betaButton.addActionListener(this);

    // Creating panel to hold buttons
    JPanel panel = new JPanel();
    panel.setLayout(newFlowLayout());
    panel.add(alphaButton);
    panel.add(betaButton);

    //Addingpaneltotheframe
    add(panel);

    setVisible(true);
}


public void actionPerformed(ActionEvent e)
    {StringactionCommand=e.getActionCommand();
    if (actionCommand.equals("Alpha")) {
        JOptionPane.showMessageDialog(this,"Alphapressed");
    } else if (actionCommand.equals("Beta")) {
        JOptionPane.showMessageDialog(this,"Betapressed");
    }
}
```

```
    public static void main(String[] args) {
        SwingUtilities.invokeLater(SwingEventHandlingDemo::new);
    }
}
```

OUTPUT:

```
D:\advance java program>javac SwingEventHandlingDemo.java
D:\advance java program>java SwingEventHandlingDemo
```

8. A program to display greeting message on the browser “HelloUsername”, “How Are You?”, acceptusername from the client using servlet.