Experiement 5:-String processing using StringBuffer and StringBuilder class

1)Write a program to demonstrate that strings in StringBuffer and StringBuilder class

are mutable.

public class mutable {

    public static void main(String[] args){

StringBuffer sb=new StringBuffer("Java");

System.out.println("Before appending to the string buffer "+sb);

sb.append(" Programming");

System.out.println("After appending Programming to a string buffer

'Java':-  "+sb);

StringBuilder sbb=new StringBuilder("Java");

sbb.append(" Programming");

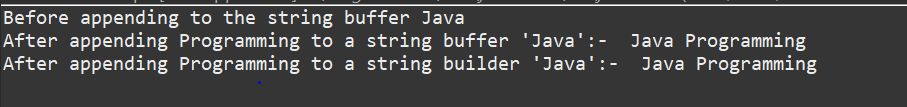
System.out.println("After appending Programming to a string builder

'Java':-  "+sb);

}

}

OUTPUT:-



2)Write a program to demonstrate common StringBuffer and StringBuilder class methods.

import java.lang.StringBuffer.\*;

import java.util.Scanner;

public class stringbuffer {

    public static void main(String[] args)

    {

        Scanner sc=new Scanner(System.in);

        StringBuffer sb=new StringBuffer(sc.next());

        System.out.println("The amount of memory used is  "+sb.capacity());

        System.out.println("The letter at index 2 is "+sb.charAt(2));

        StringBuffer s=new StringBuffer("bcde");

        System.out.println("to compare the stringbuilder with another stringbuffer bcde "+sb.compareTo(s));

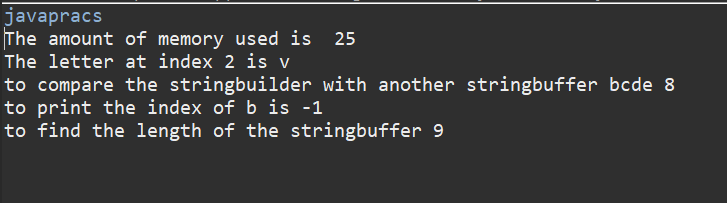
        System.out.println("to print the index of b is "+sb.indexOf("b"));

        System.out.println("to find the length of the stringbuffer "+sb.length());

    }

}

OUTPUT:-



1. **Accept a paragraph from the user. Perform the following string processing tasks**
2. **Find total no. of words.**
3. **Find duplicate words.**
4. **Display substring based on user input.**
5. **Replace most frequently used word**

import java.util.ArrayList;

import java.util.Collection;

import java.util.Collections;

import java.util.Scanner;

import java.util.StringTokenizer;

public class parawordcounting {

    public static void main(String[] args) {

        Scanner sc=new Scanner(System.in);

        ArrayList<String> al=new ArrayList<String>();

        ArrayList<String> words=new ArrayList<String>();

        ArrayList<String> duplicate=new ArrayList<String>();

        ArrayList<String> original=new ArrayList<String>();

        System.out.println("Enter the paragraph.Press enter 2 times to end the paragraph");

        String s=" ";

        while(!s.equals(""))

        {

            s=sc.nextLine();

            al.add(s);

        }

        al.remove(al.size()-1);

        for(String p:al)

        {

            StringTokenizer st=new StringTokenizer(p," ");

            while(st.hasMoreTokens())

            {

                words.add(st.nextToken());

            }

        }

        Collections.sort(words);

        int max=0,k=0;

        int arr[]=new int[words.size()];

        String maxrepeated="";

        for(int i=0;i<words.size();i++)

        {

            if(original.contains(words.get(i)))

            {

                original.remove(words.get(i));

                duplicate.add(words.get(i));

                arr[k]=2;

                if(i+1<words.size()){

                while(words.get(i).equals(words.get(i+1)))

                {

                    arr[k]++;

                    if(arr[k]>max)

                    {

                        max=arr[k];

                        maxrepeated=words.get(i);

                    }

                    i++;

                }}

                k++;

            }

            else

                original.add(words.get(i));

        }

        System.out.println("the originals are");

        for(String q:original)

            System.out.println(q);

        System.out.println("the duplicates are");

        for(int i=0;i<duplicate.size();i++)

            System.out.println(duplicate.get(i)+" "+arr[i]);

        System.out.println("The most repeated word is "+maxrepeated+

".\nIt is repeated "+max+" times");

        System.out.println("To search a particular substring");

        String subsstring=sc.nextLine();

        int c=0;

        for(String p:al)

        {

            for(int i=0;i<p.length()-subsstring.length();i++)

            {

                if(subsstring.equals(p.substring(i,

i+subsstring.length())))

                    c=1;

            }

        }

        if(c==1)

            System.out.println("The substring is found");

        else

            System.out.println("The substring is not found");

        System.out.println("Enter the word to replace the most repeated  word");

        String replace=sc.nextLine();

        String newpara="",x;

        for(String p:al)

        {

            StringTokenizer st=new StringTokenizer(p," ");

            while(st.hasMoreTokens())

            {

                x=st.nextToken();

                if(x.equals(maxrepeated))

                    newpara=newpara+replace+" ";

                //    System.out.println(newpara);}

                else

                    newpara=newpara+x+" ";

            }

            newpara+="\n";

        }

        System.out.println(newpara);

    }

}

OUTPUT:-

