1.Write an application to determine the length of the string =”Hello world”.

package string\_method;

import java.util.Scanner;

public class len {

public static void main(String[] args)

{

String str;

Scanner sc=new Scanner(System.in);

System.out.println("enter the string:");

str=sc.nextLine();

System.out.println("length of the string:"+ str.length());

}

}

Output:

enter the string:

Hello world

length of the string:11

2.Write an application to join the two string “Hello” and “How are you?”

package string\_method;

public class combine {

public static void main(String[] args)

{

String s1="Hello ",s2="How are you";

System.out.println("Print:"+ s1 +s2);

}

}

Output:

Print:Hello How are you

3. 3. Given a string “java string pool refers to collection of string which are store in heap memory” perform the following operations:

a. print the string to console in lower case

package string\_method;

import java.util.Scanner;

public class pool {

public static void main(String[] args)

{

String s1,s2;

Scanner sc=new Scanner(System.in);

System.out.println("Uppercase letter:");

s1=sc.nextLine();

s2 = s1.toLowerCase();

System.out.println("lowecase:" +s2);

}

}

Uppercase letter:

SUSHEEL

lowecase:susheel

b. Print the string to console in upper case

package string\_method;

import java.util.Scanner;

public class pool {

public static void main(String[] args)

{

String s1,s2;

Scanner sc=new Scanner(System.in);

System.out.println("Lowercase letter:");

s1=sc.nextLine();

s2 = s1.toUpperCase();

System.out.println("uppercase:" +s2);

}

}

Output:

Lowercase letter:

hello world

uppercase:HELLO WORLD

c. Replace all ‘a’ character in a string with ‘$’ sign

package string\_method;

import java.util.Scanner;

public class pool {

public static void main(String[] args)

{

String s1,s2;

Scanner sc=new Scanner(System.in);

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

s1=sc.nextLine();

s2 = s1.replace("a","$");

System.out.println("string:" +s2);

}

}

Output:

Enter the string:

an apple

string:$n $pple

d. Check if the original string contains the word “collection”

package string\_method;

import java.util.Scanner;

public class pool {

public static void main(String[] args)

{

String s1,s3;

Scanner sc=new Scanner(System.in);

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

s1=sc.nextLine();

System.out.println("Enter the string to be checked:");

s3=sc.nextLine();

boolean s2 = s1.contains(s1);

System.out.println("string:" +s2);

}

}

Output:

Enter the string:

Java uses oops concept

Enter the string to be checked:

oops

string:true

Enter the string:

"java string pool refer to collection of strings which are in heap memory"

Enter the string to be checked:

collection

string:true

f. use another method to check strings are equal.

package string\_method;

import java.util.Scanner;

public class pool {

public static void main(String[] args)

{

String s1,s3;

Scanner sc=new Scanner(System.in);

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

s1=sc.nextLine();

System.out.println("Enter the string to be checked:");

s3=sc.nextLine();

boolean s2=s1.equals(s1);

System.out.println("print:" + s2);

}

}

Output:

Enter the string:

"java string pool refer to collection of strings which are in heap memory"

Enter the string to be checked:

collection

print:true

STRINGBUFFER

1.Write an application to append the following strings “StringBuffer”, “is a peer of a String”, “that provides much of”, “the functionalities of string” using StringBuffer.

package string\_method;

public class buff {

public static void main(String[] args)

{

StringBuffer buffer=new StringBuffer("WELCOME");

buffer.append(" " +"HOME");

System.out.println(buffer);

}

}

Output:

WELCOME HOME

2. Insert the following string “insert text” into the string “it is used to \_at the specified index position” at the location denoted by \_sign using StringBuffer.

package string\_method;

public class buff {

public static void main(String[] args)

{

StringBuffer buffer=new StringBuffer("it is used to at the specified index position");

buffer.insert(14 ,"insert text ");

System.out.println(buffer);

}

}

Output:

it is used to insert text at the specified index position

Reverse the following string “This method is used to return the reverse object on which it was called” using StringBuffer class

package string\_method;

public class buff {

public static void main(String[] args)

{

StringBuffer buffer=new StringBuffer("This method is used to return the reverse object on which it was called");

buffer.reverse();

System.out.println(buffer);

}

}

Output:

dellac saw tihcihw no tcejboesreverehtnruterotdesusidohtemsihT

Provide solution for “Assignments of stringBuffer class” using StringBuilder”

1. Write an application to append the following strings “StringBuilder”, “is a peer of a String”, “that provides much of”, “the functionalities of string” using StringBuilder.

package string\_method;

public class buff {

public static void main(String[] args)

{

StringBuilder buffer=new StringBuilder("StringBuilder, is a peer of a String");

buffer.append(" "+"that provides much of");

buffer.append(" "+" the functionalities of string");

System.out.println(buffer);

}

}

Output:

StringBuilder, is a peer of a String that provides much of the functionalities of string

1. Insert the following string “insert text” into the string “it is used to \_at the specified index position” at the location denoted by \_sign using StringBuilder.

package string\_method;

public class buff {

public static void main(String[] args)

{

StringBuilder buffer=new StringBuilder("it is used to at the specified index position");

buffer.insert(14,"index text ");

System.out.println(buffer);

}

}

Output

it is used to index text at the specified index position

1. Reverse the following string “This method is used to return the reverse object on which it was called” using StringBuilder class.

package string\_method;

public class buff {

public static void main(String[] args)

{

StringBuilder buffer=new StringBuilder("This method is used to return the reverse object on which it was called");

buffer.reverse();

System.out.println(buffer);

}

}

Output:

dellac saw tihcihw no tcejboesreverehtnruterotdesusidohtemsihT