**Assignments on String Class**

1.Write an application to determine the length of string.

**public** **class** StringClass {

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

String str="Hello world";

System.***out***.println(str.length());

}

}

2.Writ an application to join 2 strings.

String str="Hello";

String str1=" How are you?";

System.***out***.println(str.concat(str1));

3.a)print the string in lowercase

**public** **class** StringClass {

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

String str="Java String pool refers to collection of Strings which

are stored in heap memory";

String str1=str.toLowerCase();

System.***out***.println(str1);

}

}

b)print the string in uppercase

String str="Java String pool refers to collection of Strings which are stored in heap memory";

String str1=str.toUpperCase();

System.***out***.println(str1);

c)Replace all the ‘a’ character in string with $ symbol

String str="Java String pool refers to collection of Strings which are stored in heap memory";

String str1=str.replace('a','$');

System.***out***.println(str1);

d)Check if the original string contains the word collection.

System.***out***.println(str.contains("collection");

e)Check if the following string “java string pool refers to collection of strings which are stored in heap memory”

System.***out***.println(str.equals("java string pool refers to collection of strings which are stored in heap memory"));

f)If the string does not match check if there is another method which can be used to check if the strings are equal.

String str="Java String pool refers to collection of Strings which are stored in heap memory";

String str1="java string pool refers to collection of strings which are stored in heap memory";

**if**(str==str1)

{

System.***out***.println("equal");

}

**else**

{

System.***out***.println("not equal");

}

**Assignments on StringBuffer Class**

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

StringBuffer sb= **new** StringBuffer("StringBuffer");

sb.append("is a peer class of string");

sb.append("that provides much of");

sb.append("the functionality of strings");

System.***out***.println(sb);

2)Insert the following string “insert text” into the string “It is used to \_

At the specified index position” at the location denoted by\_.

StringBuffer sb= **new** StringBuffer("It is used to \_ At the specified index

position");

sb.insert(15,"insert text");

System.***out***.println(sb);

3)Reverse the following string “The method returns the reversed object on which

it was called” using stringbuffer class.

StringBuffer sb= **new** StringBuffer("The method returns the reversed

object on which it was called");

sb.reverse();

System.***out***.println(sb);

**Assignments on StringBuider Class**

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

StringBuilder sb= **new** StringBuilder("StringBuilder");

sb.append("is a peer class of string");

sb.append("that provides much of");

sb.append("the functionality of strings");

System.***out***.println(sb);

2)Insert the following string “insert text” into the string “It is used to \_

At the specified index position” at the location denoted by\_.

StringBuilder sb= **new** StringBuilder("It is used to \_ At the specified index

position");

sb.insert(15,"insert text");

System.***out***.println(sb);

3)Reverse the following string “The method returns the reversed object on which

it was called” using stringbuilder class.

StringBuilder sb= **new** StringBuilder("The method returns the reversed

object on which it was called");

sb.reverse();

System.***out***.println(sb);