1. **Length of the str ?**

public class Mainkirti{

public static void main(String args[]){

method();

}

public static void method(){

String s1="hello world";

System.out.println("string length is: "+s1.length());

}

}

1. **To join two strings ?**

class Mainkirti{

public static void main(String args[]){

method();

}

public static void method(){

String s="kirti"+" kalambe";

System.out.println(s);

}

}

1. **Convert str into lower case?**

class Mainkirti{

public static void main(String args[]){

string();

}

public static void string(){

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

System.out.println(s.toLowerCase());

}

}

1. **Convert str into Uppercase?**

class Mainkirti{

public static void main(String args[]){

string();

}

public static void string(){

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

System.out.println(s.toUpperCase());

}

}

1. Replace string a to $?

public class Mainkirti{

public static void main(String args[]){

method();

}

public static void method(){

String s1="java string pool refers to collection of strings which are stored in heap memory";

String replaceString=s1.replace('a','$');

System.out.println(replaceString);

}

}

1. Original string contains the word collection?

public class Mainkirti{

public static void main(String args[]){

method();

}

public static void method(){

String s1="java string pool refers to collection of strings which are stored in heap memory";

System.out.println(s1.contains("collection"));

}

}

1. Check the string it matches to original string?

public class Mainkirti{

public static void main(String args[]){

method();

}

public static void method(){

String s1="java string pool refers to collection of strings which are stored in heap memory";

String s2="java string pool refers to collection of strings which are stored in heap memory";

System.out.println(s1.equals(s2));

}

}

1. Append strings?

public class Mainkirti{

public static void main(String args[]){

StringBuffer sb =new StringBuffer("StringBuffer");

StringBuffer sb1 =new StringBuffer("is a peer class of string ");

StringBuffer sb2 =new StringBuffer("that provide much of ");

StringBuffer sb3 =new StringBuffer("the functionality of strings ");

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

sb1.append("that provide much of ");

sb2.append("the functionality of strings ");

System.out.println(sb);

System.out.println(sb1);

System.out.println(sb2);

}

}

1. Reverse string ?

public class Mainkirti{

public static void main(String args[]){

StringBuffer sb =new StringBuffer("This method returns the reversed object on which it was called ");

sb.reverse();

System.out.println(sb);

}

}

STRING BUILDER

1. public class Mainkirti{

public static void main(String args[]){

StringBuilder sb =new StringBuilder("StringBuffer");

StringBuilder sb1 =new StringBuilder("is a peer class of string ");

StringBuilder sb2 =new StringBuilder("that provide much of ");

StringBuilder sb3 =new StringBuilder("the functionality of strings ");

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

sb1.append("that provide much of ");

sb2.append("the functionality of strings ");

System.out.println(sb);

System.out.println(sb1);

System.out.println(sb2);

}

}

1. public class Mainkirti{

public static void main(String args[]){

StringBuilder sb =new StringBuilder ("This method returns the reversed object on which it was called ");

sb.reverse();

System.out.println(sb);

}

}