1.. **package** string;

**public** **class** JavaString {

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

{

String str="Hello World";

**int** len=str.length();

System.***out***.println("length of string is"+len);

}

}

Output

11

2. **package** string;

**public** **class** JavaString {

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

{

String str1="Hello ";

String str2="How are You";

String result=str1.concat(str2);

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

}

}

Output

Hello How are you

3.a. **package** string;

**public** **class** JavaString {

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

{

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

are stored in heap memory";

String result=str.toUpperCase();

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

}

}

Output

JAVA STRING POOL REFERS TO COLLECTION OF STRINGS WHICH ARE STORED IN HEAP MEMORY

c. **package** string;

**public** **class** JavaString {

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

{

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

are stored in heap memory";

String result=str.toLowerCase();

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

}

}

d. **package** string;

**public** **class** JavaString {

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

{

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

are stored in heap memory";

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

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

}

}

Output

J$v$ String pool refers to collection of Strings which $re stored in he$p memory

e. **package** string;

**public** **class** JavaString {

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

{

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

are stored in heap memory";

**if**(str.contains("collection"))

{

System.***out***.println("string contain collection");

}

}

}

Output

string contain collection

f. **package** string;

**public** **class** JavaString {

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

{

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";

**boolean** result=str.equalsIgnoreCase(str1);

System.***out***.println("Does String str matches str1="+result);

}

}

Output

Does String str matches str1=true

Assignments on StringBuffer Class

1. **public** **class** stringbuffer {

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

{

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

sb.append("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);

}

}

Output

StringBuffer, is a peer class of String,that provides much of,the functionality of strings.

2. **package** string;

**public** **class** JavaString {

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

{

StringBuffer sbf=**new** StringBuffer("It is used to at the specified

index");

System.***out***.println("String buffer=" + sbf);

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

System.***out***.println("String buffer=" + sbf);

}

}

output

String buffer=It is used to at the specified index

String buffer=It is used to insert text at the specified index

**3.package** stringbuffer;

**public** **class** StringBuffer {

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

{

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

System.***out***.println("String buffer="+sbf);

sbf.reverse();

System.***out***.println("String buffer="+sbf);

}}

Output

String buffer=This method returns the reversed object on which it was called

String buffer=dellac saw ti hcihw no tcejbo desrever eht snruter dohtem sihT

Assignments on StringBuilder Class

1. **public** **class** stringbuilder {

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

{

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

sb.append("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);

}

}

Output

StringBuffer, is a peer class of String,that provides much of,the functionality of strings.

2. **public** **class** stringbuilder {

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

{

StringBuilder sbf=**new** StringBuilder("It is used to at the

specified index");

System.***out***.println("String builder=" + sbf);

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

System.***out***.println("String builder=" + sbf);

}

}

Output

String builder=It is used to at the specified index

String builder=It is used to insert text at the specified index

3.**public** **class** stringbuilder {

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

{

StringBuilder sb=**new** StringBuilder("This method returns the

reversed object on which it was called");

System.***out***.println("String builder=" +sb.toString());

StringBuilder reverseStr=sb.reverse();

System.***out***.println("String builder=" + reverseStr.toString());

} }

Output

String builder=This method returns the reversed object on which it was called

String builder=dellac saw ti hcihw no tcejbo desrever eht snruter dohtem sihT