1. public class D {

public static void main(String[] args) {

String s1 = "abc";

String s2 = new String("rst");

System.out.println(s1.toString());

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

} }

1. public class I {

public static void main(String[] args) {

String s1 = "abc";

String s2 = "abc";

String s3 = new String(s1);

String s4 = new String(s2);

System.out.println("------");

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

System.out.println(s3 == s4);

System.out.println(s1 == s4);

System.out.println(s3 == s4);

System.out.println("--------------");

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

System.out.println(s1.equals(s3));

} }

1. public class K {

public static void main(String[] args) {

String s1 = "ABC";

String s2 = "abc";

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

System.out.println("====");

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

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

System.out.println("=======");

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

System.out.println("done");

} }

1. public class Q {

public static void main(String[] args) {

System.out.println(2 + 4);

System.out.println(2 + "abc");

System.out.println("abc" + null);

System.out.println("abc" + 2 + null);

} }

1. public class T {

public static void main(String[] args) {

String s1 = "ja";

String s2 = "va";

String s3 = s1.concat(s2);

String s4 = "java";

System.out.println(s3 == s4);

} }

1. public class V {

public static void main(String[] args) {

String s1 = "null";

System.out.println(s1.length());

} }

1. public class Z {

public static void main(String[] args) {

String s1 = "abc123xyz";

System.out.println(s1.length());

System.out.println("-------------");

System.out.println(s1.charAt(0));

System.out.println(s1.charAt(3));

System.out.println(s1.charAt(9));

} }

1. public class A {

public static void main(String[] args) {

String s1 = "a1b2c3x1y2x3";

System.out.println(s1.length());

System.out.println(s1.indexOf('2'));

System.out.println(s1.indexOf('x'));

System.out.println(s1.indexOf('x',8));

System.out.println(s1.indexOf(2, 3));

}

}

1. public class C {

public static void main(String[] args) {

String s1 = "abc technology";

String s2 = s1.substring(5);

String s3 = s1.substring(5,9);

System.out.println(s2);

System.out.println("---------");

System.out.println(s3);

System.out.println("---------------");

} }

1. public class E {

public static void main(String[] args) {

String s1 = "abc";

String s2 = "Abc";

String s3 = "xyz";

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

System.out.println(s1.compareTo(s3));

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

} }

1. public class G {

public static void main(String[] args) {

String s1 = "abc tech";

System.out.println(s1.startsWith("abc"));

System.out.println(s1.startsWith("123"));

System.out.println(s1.endsWith("te"));

System.out.println("done");

} }

1. public class J {

public static void main(String[] args) {

StringBuffer sb1 = new StringBuffer("abc");

sb1.append("xyz");

sb1.append("\n");

sb1.append("hello\t");

sb1.append("test");

System.out.println(sb1);

} }

1. public class M {

public static void main(String[] args) {

StringBuffer sb1 = new StringBuffer();

sb1.append("abc");

sb1.append("xyz");

StringBuffer sb2 = new StringBuffer();

sb2.append("abc");

sb2.append("xyz");

System.out.println(sb1.equals(sb2));

System.out.println("done");

} }

1. public class P {

public static void main(String[] args) {

String s1 = " abc xyz ";

s1.trim();

System.out.println(s1.length());

s1 = s1.trim();

System.out.println(s1.trim().length());

} }

1. public class T {

public static void main(String[] args) {

StringBuilder sb = new StringBuilder();

sb.append("abc");

sb.append("hello");

System.out.println(sb);

System.out.println("--------");

System.out.println(sb.delete(0, 3));

System.out.println("---------");

System.out.println(sb.deleteCharAt(4));

} }

1. public class A {

public static void main(String[] args) {

System.out.printf("hello %s","india");

System.out.println("");

String s1 = "indian";

System.out.printf("hi %s",s1);

} }

1. public class B {

public static void main(String[] args) {

System.out.printf("hello %d","india");

System.out.println("");

String s1 = "indian";

System.out.printf("hi %s",s1);

} }

1. public class C {

public static void main(String[] args) {

System.out.printf("---%2$d-----%1$d",20,45);

System.out.println("");

System.out.printf("---%2$s-----%1$s","abc","xyz");

} }

1. public class D {

public static void main(String[] args) {

System.out.printf("(%+d)",100);

} }

1. import java.util.regex.Matcher;

import java.util.regex.Pattern;

public class E {

public static void main(String[] args) {

String src = "abcababcababc";

String exp = "a";

Pattern p1 = Pattern.compile(exp);

Matcher m1 = p1.matcher(src);

while(m1.find()) {

System.out.println(m1.start()+":"+m1.group());

}

}

}

1. import java.util.regex.Matcher;

import java.util.regex.Pattern;

public class F {

public static void main(String[] args) {

String s = "AbcabcAbcABC";

String s1 = "A";

Pattern p = Pattern.compile(s1);

Matcher m = p.matcher(s);

while(m.find()) {

System.out.println(m.start()+":"+m.group());

} }

}

1. import java.util.regex.Matcher;

import java.util.regex.Pattern;

public class G {

public static void main(String[] args)

{

String s = "abcxyz1pqr2km9rtz";

Pattern p = Pattern.compile("[a-p]");

Matcher m = p.matcher(s);

while(m.find()) { System.out.println(m.start()+":"+m.group());

} }

}

1. import java.util.regex.Matcher;

import java.util.regex.Pattern;

public class H {

public static void main(String[] args) {

String s = " abc 123 sd abcd 123 rty ";

Pattern p = Pattern.compile("[ ]");

Matcher m = p.matcher(s);

int i=0;

while(m.find()) {

i++;

System.out.println(i);

} }

}

1. import java.util.regex.Matcher;

import java.util.regex.Pattern;

public class I {

public static void main(String[] args) {

String s = "abcxyz1pqr2km9rst";

Pattern p = Pattern.compile("[a-bP-T5-9]");

Matcher m = p.matcher(s);

while(m.find()) { System.out.println(m.start()+":"+m.group());

}

} }

1. public class J {

public static void main(String[] args) {

String s = "a1b2c3h4z9";

String [] x = s.split("[0-4]");

for(String s1:x) {

System.out.println(s1);

}

System.out.println("========");

String [] y = s.split("[a-c]");

for(String s2:y) {

System.out.println(s2);

} } }

1. import java.util.StringTokenizer;

public class V {

public static void main(String[] args) {

String str = "This is String , split by StringTokenizer, created by Abc";

StringTokenizer st = new StringTokenizer(str);

System.out.println("---- Split by space ------");

while (st.hasMoreElements()) {

System.out.print(st.nextElement());

} }

}

1. import java.util.Date;

public class Q {

public static void main(String[] args) {

Date date = new Date();

System.out.println(date);

} }

1. import java.util.Date;

public class P {

public static void main(String[] args) {

Date date = new Date(1000);

System.out.println(date);

} }

1. import java.util.Date;

public class K {

public static void main(String[] args) {

Date date = new Date(-1000\*60\*60\*24);

System.out.println(date);

} }

1. import java.util.Calendar;

import java.util.Date;

public class W {

public static void main(String[] args) {

Calendar calsi = Calendar.getInstance();

calsi.add(Calendar.DATE, 1);

Date date = calsi.getTime();

System.out.println(date);

} }

1. import java.util.Calendar;

import java.util.Date;

public class X {

public static void main(String[] args) {

Calendar cal = Calendar.getInstance();

cal.add(Calendar.YEAR, 1);

Date date = cal.getTime();

System.out.println(date);

} }

1. import java.text.NumberFormat;

public class Z1 {

public static void main(String[] args) {

double num = 129876.9090973563;

System.out.println(num);

NumberFormat nf = NumberFormat.getInstance();

String s1 = nf.format(num);

System.out.println(s1);

} }

1. public class A1 {

public static void main(String[] args) {

String s1="abc xyz hello abc";

String s2=s1.replaceAll("abc", "text");

System.out.println(s1);

System.out.println(s2);

} }

1. class InvertString {

public static void main(String args[]) {

StringBuffer a = new StringBuffer("Java programming isfun");

System.out.println(a.reverse());

} }

1. public class ReverseString {

public static void main(String[] args) {

String s1 = "abc technology";

char c[] = new char[5];

System.out.printf("given string : %s",s1);

System.out.printf("\nlength of string:%d",s1.length());

System.out.print("\n the reversed string is: ");

for( int count = s1.length()-1; count >= 0; count--) {

System.out.printf("%s",s1.charAt(count));

} }

}

1. public class I1 {

public static void main(String[] args) {

String s1="hello";

String s2="hello";

String s3=new String("hello");

String s4=new String("hello");

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

System.out.println(s1==s3);

System.out.println(s3==s4);

} }

1. public class H1 {

public static void main(String[] args) {

String s1=new String("abc technologies");

System.out.println(s1.substring(2,8));

System.out.println(s1.length());

} }

1. public class E1 {

public static void main(String[] args) {

String s1=" abc 123 ";

s1.trim();

String s2="abc 123";

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

} }

1. String is class or interface or abstract class?
2. What is return type for hashcode?
3. What is the string?
4. The StringBuilder class was introduced in JDK.5(True **or** False)?
5. The StringBuilder is non-synchronized?(True **or** False)
6. What is the use of CharAt() method?
7. What is the use of concat() method in java?
8. When String equals() method return false?
9. Explain IndexOf() method in two line.
10. The string replace() method was introduced as of JDK1.5?(True **or** False)
11. Which method of string class is used to obtain character at specified index?
12. String is class in java and defined in java.lang package.(True **or** False).
13. How can we make String upper case and lower case?
14. String implements the comparable interface.(Yes **or** No).
15. What is the difference between StringBuffere and StringBuilder?
16. String is thread-safe in java?(Yes **or** No**)** Give reason.
17. What is the use of trim() method in java?
18. Inside a String hashCode() and equals() method got override(True **or** False).
19. StringBuffer class available inside java.lang package(Yes **or** No)
20. What is the wildcard character?
21. What is the use of StringTokenizer in java?
22. In which package StringTokenizer is available?