Strings

Exercises and solutions

1. What is a string in Java? Can you change the contents of a String object after it is created?

**Answer:**

A sequence of zero or more characters is known as a string. A string in Java is immutable.

1. What is a string literal?

**Answer:**

A string literal consists of a sequence of zero or more characters enclosed in double quotes.

1. What is the difference between the String class and the StringBuilder class?

**Answer:**

The String class is an immutable class. The StringBuilder class is a mutable companion class for the String class. String can be used in switch statement as case labels, but a StringBuilder cannot.

1. What is the difference between the StringBuffer class and the StringBuilder class?

**Answer:**

StringBuffer is thread-safe and StringBuilder is not thread-safe.

1. Write the output when the following snippet of code is executed.  
     
   String s1 = "Hello";  
   String s2 = "\"Hello\"";  
   System.out.println("s1 = " + s1);  
   System.out.println("s2 = " + s2);

**Answer:**

s1 = Hello

s2 = "Hello"

1. Write the output when the following snippet of code is executed.  
     
   String s1 = "Who\nknows";  
   System.out.println("s1 = " + s1);

**Answer:**

s1 = Who

knows

1. Write the output when the following snippet of code is executed.  
     
   String s1 = "Having fun with strings";  
   int len = s1.length();  
   char c = s1.charAt(4);

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

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

**Answer:**

len = 23

c = n

1. Write the output when the following snippet of code is executed.  
     
   String s1 = "Fun";  
   String s2 = new String("Fun");  
   System.out.println(s1 == s2);  
   System.out.println(s1.equals(s2));  
   System.out.println("Fun" == "Fun");

**Answer:**

false

true

true

1. Write the output when the following snippet of code is executed.  
     
   StringBuilder sb = new StringBuilder(200);  
   sb.append("Hello").append(false);   
   System.out.println("length = " + sb.length());  
   System.out.println("capacily = " + sb.capacity());  
   System.out.println(sb.toString());

**Answer:**

length = 10

capacily = 200

Hellofalse

1. Write the output when the following snippet of code is executed.  
     
   String s1 = 10 + 20 + " = what";  
   String s2 = 10 + String.valueOf(20) + " = what";  
   System.out.println(s1);  
   System.out.println(s2);

**Answer:**

30 = what

1020 = what

1. Complete the code for a method named equalsContents() as declared below. The method should return true if both arguments have the same contents after removing leading and trailing whitespaces and ignoring cases. If both arguments are null it should return true. Otherwise, it should return false.  
     
   public static boolean equalsContents(String s1, String s2) {  
    /\* your code goes here\*/  
   }

**Solution:**

public static boolean equalsContents(String s1, String s2) {

if (s1 == null && s2 == null) {

return true;

} else if (s1 == null || s2 == null) {

return false;

} else {

return s1.trim().equalsIgnoreCase(s2.trim());

}

}

1. Complete following code, so that the year, month, and day are printed as 1969, 09, and 19.  
     
   String date = "1969-09-19";  
   String year = date./\*your code goes here\*/;  
   String month = date./\*your code goes here\*/;  
   String day = date./\*your code goes here\*/;  
   System.out.println("year = " + year);  
   System.out.println("month = " + month);  
   System.out.println("day = " + day);

**Solution:**

String date = "1969-09-19";

String year = date.**substring(0, 4)**;

String month = date.**substring(5, 7)**;

String day = date.**substring(8, 10)**;

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

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

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

1. Complete the following snippet of code, so it prints the expected output, which is shown after the snippet.   
     
   String s1 = "noon and spoon";  
   String s2 = s1./\*Your code goes here\*/;  
   System.out.println(s1);  
   System.out.println(s2);  
     
   Expected output is as follows:  
   noon and spoon  
   nun and spun

**Solution:**

String s1 = "noon and spoon";  
String s2 = s1.replace("oo", "u");  
System.out.println(s1);  
System.out.println(s2);

1. Complete the following snippet of code, so it prints the expected output, which is shown after the snippet.   
     
   String s1 = "noon and spoon";  
   String s2 = s1./\*Your code goes here\*/;  
   System.out.println(s1);  
   System.out.println(s2);  
     
   Expected output is as follows:  
   noon and spoon  
   nn and spn

**Solution:**

String s1 = "noon and spoon";

String s2 = s1.**replace("oo", "")**;

System.out.println(s1);

System.out.println(s2);

1. Complete code for a reverse(String str) method. It takes a string and returns the reverse of that string. Do not use StringBuilder or StringBuffer classes.  
     
   public static String reverse(String str) {  
    /\* Your code goes here \*/   
   }

**Solution:**

The following solution uses a char[] to reverse a string. Note that a string may contain Unicode surrogate pairs, which need to be taken into account when reversing the string. This solution takes care of the Unicode surrogate pairs in strings.

public static String reverse(String s) {

if (s == null || s.length() < 2) {

return s;

}

char[] chars = new char[s.length()];

for (int i = s.length() - 1, charsIndex = 0; i >= 0; --i) {

if (Character.isLowSurrogate(s.charAt(i))) {

--i;

chars[charsIndex++] = s.charAt(i);

chars[charsIndex++] = s.charAt(i + 1);

} else {

chars[charsIndex++] = s.charAt(i);

}

}

return new String(chars);

}

1. What is the value of the expression "abc".compareTo("abc").

**Answer:**

0