

# commons-lang3

- description : commons-lang3
- author :
- email : dhan@repia.com
- lastupdate : 2020-03-24

## MavanRepository

```

<!-- https://mvnrepository.com/artifact/org.apache.commons/commons-lang3 -->
<dependency>
  <groupId>org.apache.commons</groupId>
  <artifactId>commons-lang3</artifactId>
  <version>3.8</version>
</dependency>

```

[MVN Repository](#)

## StringEscapeUtils

StringEscapeUtils ( [View Source](#) ) 2020-04-06

## StringUtils

StringUtils.abbreviate ( [View Source](#) , [View Examples](#) )

4 [View Source](#) [View Examples](#) ... ( 3 ) [View Source](#) [View Examples](#)

```

String abbreviate = StringUtils.abbreviate("ShowMeTheMoney", 7);
System.out.println(abbreviate);
//           : Show...

```

StringUtils.appendIfMissing ( [View Source](#) , 가 , [View Examples](#) )

가 , 가

```

String appendIfMissing1 = StringUtils.appendIfMissing("ShowMeTheMoney", "-From.Choi", "Money");

```

```
System.out.println(appendIfMissing1);
//           : ShowMeTheMoney

String appendIfMissing2 = StringUtils.appendIfMissing("ShowMeTheMoney", "-
From.Choi", "test");
System.out.println(appendIfMissing2);
//           : ShowMeTheMoney-From.Choi
```

StringUtils.appendIfMissingIgnoreCase ( , 가 , )

StringUtils.appendIfMissing

```
String appendIfMissingIgnoreCase =
StringUtils.appendIfMissingIgnoreCase("ShowMeTheMoney", "-From.Choi",
"money");
System.out.println(appendIfMissingIgnoreCase);
//           : ShowMeTheMoney
```

StringUtils.capitalize ( )

```
String capitalize = StringUtils.capitalize("aaa");
System.out.println(capitalize);
//           : Aaa
```

StringUtils.chomp ( )

\n, \r, \r\n

```
String chomp = StringUtils.chomp("sss\n");
System.out.println(chomp);
//           : SSS
```

StringUtils.chomp ( , )

가

```
String chomp = StringUtils.chomp("sss#", "#");
System.out.println(chomp);
//           : SSS
```



## StringUtils.compare

```
int compareIgnoreCase = StringUtils.compareIgnoreCase("A", "a");
System.out.println(compareIgnoreCase);
//           : 0
```

StringUtils.contains ( , )

'true'

```
boolean contains = StringUtils.contains("Hello", "el");
System.out.println(contains);
//           : true
```

StringUtils.containsAny ( , 1, 2, ...)

'true'

```
boolean containsAny = StringUtils.containsAny(" ", "가", " ", " ");
System.out.println(containsAny);
//           : true
```

StringUtils.containsIgnoreCase ( , )

## StringUtils.contains

```
boolean containsIgnoreCase = StringUtils.containsIgnoreCase("HELLO", "he");
System.out.println(containsIgnoreCase);
//           : true
```

StringUtils.containsNone ( , )

'false'

```
boolean containsNone = StringUtils.containsNone("hello", "h");
System.out.println(containsNone);
//           : false
```

```
boolean containsNone = StringUtils.containsNone("hello", "aaah");
System.out.println(containsNone);
//           : false
```

```
StringUtils.containsOnly ( , )
```

```
'true' .
```

```
boolean containsOnly = StringUtils.containsOnly("hello", "olleh");
System.out.println(containsOnly);
//      : true

boolean containsOnly2 = StringUtils.containsOnly("hello", "hell");
System.out.println(containsOnly2);
//      : false
```

```
StringUtils.containsWhitespace ( )
```

```
'true' .
```

```
boolean containsWhitespace = StringUtils.containsWhitespace(" ");
System.out.println(containsWhitespace);
//      : true
```

```
StringUtils.countMatches ( , )
```

```
1 , 0 .
```

```
int countMatches = StringUtils.countMatches("ABCD", "A");
System.out.println(countMatches);
//      : 1

int countMatches2 = StringUtils.countMatches("ABCD", "AB");
System.out.println(countMatches2);
//      : 1

int countMatches3 = StringUtils.countMatches("ABCD", "AC");
System.out.println(countMatches3);
//      : 0
```

```
StringUtils.defaultIfBlank ( , )
```

```
space( ), , null .
```

```
String defaultIfBlank1 = StringUtils.defaultIfBlank(null, "A");
System.out.println(defaultIfBlank1);
//      : A

String defaultIfBlank2 = StringUtils.defaultIfBlank("", "A");
```

```
System.out.println(defaultIfBlank2);  
//      : A  
  
String defaultIfBlank3 = StringUtils.defaultIfBlank("  ", "A");  
System.out.println(defaultIfBlank3);  
//      : A
```

StringUtils.defaultIfEmpty ( , )

, null

```
String defaultIfEmpty1 = StringUtils.defaultIfEmpty(null, "B");  
System.out.println(defaultIfEmpty1);  
//      : B  
  
String defaultIfEmpty2 = StringUtils.defaultIfEmpty("", "B");  
System.out.println(defaultIfEmpty2);  
//      : B  
  
String defaultIfEmpty3 = StringUtils.defaultIfEmpty("  ", "B");  
System.out.println(defaultIfEmpty3);  
//      :
```

StringUtils.defaultString ( , )

null

```
String defaultString1 = StringUtils.defaultString(null, "C");  
System.out.println(defaultString1);  
//      : C  
  
String defaultString2 = StringUtils.defaultString("", "C");  
System.out.println(defaultString2);  
//      :  
  
String defaultString3 = StringUtils.defaultString("  ", "C");  
System.out.println(defaultString3);  
//      :
```

StringUtils.deleteWhitespace ( )

```
String deleteWhitespace = StringUtils.deleteWhitespace(" ");  
System.out.println(deleteWhitespace);
```

```
// :
```

```
StringUtils.difference ( 1, 2)
```

```
2
```

```
String difference = StringUtils.difference(" .", " .");
System.out.println(difference);
// :
```

```
StringUtils.endsWith ( , )
```

```
'true',
```

```
'false'
```

```
boolean endsWith = StringUtils.endsWith(" .", ".");
System.out.println(endsWith);
// : true
```

```
boolean endsWith = StringUtils.endsWith(" ", ".");
System.out.println(endsWith);
// : false
```

```
StringUtils.endsWithAny ( , 1, 2, ...)
```

```
StringUtils.endsWith
```

```
'true'
```

```
boolean endsWithAny = StringUtils.endsWithAny("AAA", "a");
System.out.println(endsWithAny);
// : false
```

```
boolean endsWithAny = StringUtils.endsWithAny("AAA", "a", "A");
System.out.println(endsWithAny);
// : true
```

```
StringUtils.endsWithIgnoreCase ( , )
```

```
StringUtils.endsWith
```

```
boolean endsWithIgnoreCase = StringUtils.endsWithIgnoreCase("BBB", "b");
System.out.println(endsWithIgnoreCase);
// : true
```

StringUtils.equals ( 1, 2)

'true'

```
boolean equals = StringUtils.equals("abcd", "abcd");  
System.out.println(equals);  
//      : true  
  
boolean equals = StringUtils.equals("abcd", "Abcd");  
System.out.println(equals);  
//      : false
```

StringUtils.equalsIgnoreCase ( , )

StringUtils.equals

```
boolean equalsIgnoreCase = StringUtils.equalsIgnoreCase("abcd", "Abcd");  
System.out.println(equalsIgnoreCase);  
//      : true
```

StringUtils.equalsAny ( , 1, 2, ...)

StringUtils.equals

'true'

```
boolean equalsAny = StringUtils.equalsAny("abcd", "a", "ab", "abcd");  
System.out.println(equalsAny);  
//      : true
```

StringUtils.equalsAnyIgnoreCase ( , 1, 2, ...)

StringUtils.equalsAny

```
boolean equalsAnyIgnoreCase = StringUtils.equalsAnyIgnoreCase("abc",  
"abcde", "ABC");  
System.out.println(equalsAnyIgnoreCase);  
//      : true
```

StringUtils.firstNonBlank ( 1, 2, 3, ...)

null, ,

## StringUtils v.3.8

```
String firstNonBlank = StringUtils.firstNonBlank(null, "", " ", "first",
"second");
System.out.println(firstNonBlank);
//      : first
```

StringUtils.getCommonPrefix ( 1, 2, 3, ...)

```
String getCommonPrefix = StringUtils.getCommonPrefix("a", "ab", "abc");
System.out.println(getCommonPrefix);
//      : a
```

StringUtils.getDigits ( )

```
String getDigits = StringUtils.getDigits("123 English 456");
System.out.println(getDigits);
//      : 123456
```

StringUtils.indexOf ( , )

0  
-1

```
int indexOf1 = StringUtils.indexOf("abc", "b");
System.out.println(indexOf1);
//      : 1

int indexOf2 = StringUtils.indexOf("abc", "z");
System.out.println(indexOf2);
//      : -1
```

StringUtils.indexOfIgnoreCase ( , )

## StringUtils.indexOf

```
int indexOfIgnoreCase = StringUtils.indexOfIgnoreCase("abs", "A");
System.out.println(indexOfIgnoreCase);
```

// : 0

StringUtils.indexOfAny ( , 1, 2, ...)

StringUtils.indexOf

```
int indexOfAny = StringUtils.indexOfAny("가", "가", " ", " ", " ");
System.out.println(indexOfAny);
// : 2
```

StringUtils.indexOfAnyBut ( , )

0

```
int indexOfAnyBut = StringUtils.indexOfAnyBut("가", " ");
System.out.println(indexOfAnyBut);
// : 1
```

StringUtils.indexOfDifference ( 1, 2)

0

```
int indexOfDifference = StringUtils.indexOfDifference("가", "가");
System.out.println(indexOfDifference);
// : 3
```

## Ref

- [StringUtils](#)  
[, eleven, stringutils, commonslang, commonslang3](#)

From: <http://125.132.25.164/dokuwiki/> -

- 2023.12

Permanent link: <http://125.132.25.164/dokuwiki/doku.php?id=wiki:java:commons-lang&rev=1586767239>

Last update: **2022/03/10 19:52**

