

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 (. 2020-04-06)

StringUtils

```
StringUtils.abbreviate ( , )
.
.
.
4 . ... (3 ) .
```

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

```
StringUtils.appendIfMissing( , 가 , )
( 가 ) 가 가
```

```
String appendIfMissing = StringUtils.appendIfMissing("ShowMeTheMoney", "-
```

```

From.Choi", "
Money");
System.out.println(appendIfMissing);
//      : ShowMeTheMoney
appendIfMissing = StringUtils.appendIfMissing("ShowMeTheMoney", "-
From.Choi", "test");
System.out.println(appendIfMissing);
//      : 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("sss");
System.out.println(capitalize);
      : Sss

```

StringUtils.chomp();

\n,\r,\r\n

```

String.chomp = StringUtils.chomp("sssss\n");
System.out.println(chomp);
//      : SSSSSS

```

StringUtils.chomp(,);

가

```

String.chomp2 = StringUtils.chomp("sssss#", "#");
System.out.println(chomp2);
//      : SSSSSS

```



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

StringUtils.contains(,)

true .

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

StringUtils.containsAny(, ,)

true .

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

StringUtils.containsIgnoreCase(,)

() true .

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

StringUtils.containsNone(,)

false .

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

StringUtils.containsOnly(,)

true .

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

StringUtils.containsWhitespace(,)

true .

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

StringUtils.countMatches(,)

1 가 .

```
int countMatches = StringUtils.countMatches("c caaaa", "cc");
System.out.println(countMatches);
//           : 1
```

StringUtils.defaultIfBlank(,)

, , null

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

StringUtils.defaultIfEmpty(,)

, null

```
String defaultIfEmpty = StringUtils.defaultIfEmpty("", " ", null);
```

```
System.out.println(defaultIfEmpty);  
//      :
```

```
StringUtils.defaultString(      ,      )  
  
      null  
,  
(      " "      .)
```

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

```
StringUtils.deleteWhitespace(      )
```

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

```
StringUtils.difference(      ,      )
```

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

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

```
      ,  
      ,  
      ,  
      ,  
      true
```

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

```
StringUtils.endsWithAny(      ,      )
```

```
      ,  
      ,  
      ,  
      ,  
(vararg      ) true      . ( /      )
```

```
boolean endsWithAny = StringUtils.endsWithAny("abc", ".s", ".S");
System.out.println(endsWithAny);
//           : false
```

StringUtils.endsWithAny(,)

true .
(/)

```
boolean endsWithIgnoreCase = StringUtils.endsWithIgnoreCase("abc", ".s", ".S");
System.out.println(endsWithIgnoreCase);
//           : true
```

StringUtils.equals(,)

true .

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

StringUtils.equalsAny(,)

(vararg)

true .

```
boolean equalsAny = StringUtils.equalsAny("ss", "sS", "SS");
System.out.println(equalsAny);
//           : false
```

StringUtils.equalsAnyIgnoreCase(,)

(vararg)

true .
(/)

```
boolean equalsAnyIgnoreCase = StringUtils.equalsAnyIgnoreCase("ss", "sS", "SS");
System.out.println(equalsAnyIgnoreCase);
```

```
// : true
```

```
StringUtils.equalsIgnoreCase( , )
```

```
true
( / )
```

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

```
StringUtils.firstNonBlank( , , )
```

null, , 가
StringUtils 3.8 가

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

```
StringUtils.getCommonPrefix(new String[] { " ", " " })
```

```
String getCommonPrefix = StringUtils.getCommonPrefix("sss", "ssS", "ssa");
System.out.println(getCommonPrefix);
// : ss
```

```
StringUtils.getDigits(" ")
```

Unicode Number() .

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

```
StringUtils.indexOf(" ", " ")
```

```
int indexOf = StringUtils.indexOf("sS", "S");
System.out.println(indexOf);
//           : 1
```

```
StringUtils.indexOfAny("sS", new String[] {"s", "S"})
```

```
(vararg)
```

```
int indexOfAny = StringUtils.indexOfAny("sS", new String[] {"s", "S"});
System.out.println(indexOfAny);
//           : 0
```

```
StringUtils.indexOfAnyBut("sS", new char[] {'s', 'S'})
```

```
int indexOfAnyBut = StringUtils.indexOfAnyBut("sS", new char[] {'s', 'S'});
System.out.println(indexOfAnyBut);
//           : 2
```

```
StringUtils.indexDifference("sS", "S");
```

```
int indexDifference = StringUtils.indexDifference("sS", "S");
System.out.println(indexDifference);
//           : 3
```

```
StringUtils.indexOfIgnoreCase("sS", "S")
```

```
( / )
```

```
int indexOfIgnoreCase = StringUtils.indexOfIgnoreCase("sS", "S");
System.out.println(indexOfIgnoreCase);
//           : 0
```

Ref

- [StringUtils](#)

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

From:

<http://rwiki.repia.com/> -

. - 2023.12

Permanent link:

<http://rwiki.repia.com/doku.php?id=wiki:java:commons-lang&rev=1586756554>



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