



## Welcome to UX MILK

UX MILKはより良いサービスやプロダクトを作りたい人のためのメディアです。

[このサイトについて](#)

## おすすめ

フェアな議論とフェアな視点がエシカルなデザインを生む

UX MILK編集部

丁寧でやさしい「スローデザイン」を実践し日々良い選択をする

UX MILK編集部

コンセプト作りとプロト検証は同時並行、フライング・ペンギンズが実践する超並列UXデザイン

UX MILK編集部

## 目次

[記事トップ](#)

[文字列を置換する：replace\(\)](#)

[文字列を正規表現で置換する：replaceAll\(\)](#)

[最初のひとつだけ置換する：replaceFirst\(\)](#)

## 購読

平日・週2回更新



2018/5/24更新

# Javaで文字列を置換する： replace(), replaceAll(), replaceFirst()



2

Java で文字列を置換するときには、String クラスに組み込まれている、replace(), replaceAll(), replaceFirst() のメソッドを使うと簡単に実現することができます。ここでは、これらのメソッドの使い方と違いについて説明します。

## 文字列を置換する：replace()

replace() を使うときは以下のように記述します。

対象の文字列.**replace**(置換される文字列 , 置換 する文字列)

replace() は、対象の文字列から第1引数で指定した文字列を検索し、マッチした文字列を第2引数で指定した文字列で置換します。そして、置換後の文字列を戻り値として返します。

以下は、replace() を使用したサンプルです。

```
String str1 = "abc123abc123";
String str2 = str1.replace("abc", "0");
System.out.println(str2); // 01230123
```

上のプログラムでは、文字列 abc123abc123 の中にある文字列 abc を 0 に置換しています。結果として、置換後の文字列 01230123 が表示れます。

## 文字列を正規表現で置換する：replaceAll()

単純な文字列置換の場合は、replace() を使うことで実現できますが、正規表現を使いたい場合は replaceAll() を使います。

対象の文字列.**replaceAll**(正規表現 , 置換 する文字列)

replaceAll() は、対象の文字列から第1引数で指定した正規表現のパターンで文字列を検索し、マッチした文字列を第2引数で指定した文字列で置換します。そして、置換後の文字列を戻り値として返します。

以下は、replaceAll() を使用したサンプルです。

```
String str3 = "abc123abc123";
String str4 = str3.replaceAll("[a-z]+", "0");
System.out.println(str4); // 01230123
```

replaceAll() の第1引数では正規表現を指定しており、小文字のアルファベットの繰り返しを見つけたら 0 に置換します。結果として、置換後の文字列 01230123 が表示れます。

## 最初のひとつだけ置換する：replaceFirst()

replaceFirst() は対象の文字列の中から最初にマッチした文字列だけを置換します。

対象の文字列.**replaceFirst**(正規表現 , 置換 する文字列)

replaceFirst() は、対象の文字列から第1引数で指定した正規表現のパターンで文字列を検索し、最初にマッチした文字列のみを第2引数で指定した文字列で置換します。そして、置換後の文字列を戻り値として返します。

以下は、replaceFirst() を使用したサンプルです。

```
String str5 = "abc123abc123";
String str6 = str5.replaceFirst("[a-z]+", "0");
System.out.println(str6); // 0123abc123
```

replaceFirst() の第1引数では正規表現を指定しており、小文字のアルファベットの繰り返しを見つけたら1回だけ 0 に置換します。結果として、置換後の文字列 0123abc123 が表示れます。

Java



2

## おすすめ



**フェアな議論とフェアな視点がエシカルなデザインを生む**

2020/6/1 UX MILK編集部



**丁寧でやさしい「スローデザイン」を実践し日々良い選択をする**

2020/4/20 UX MILK編集部



**コンセプト作りとプロト検証は同時並行、フライング・ペンギンズが実践する超並列UXデザイン**

2020/3/25 UX MILK編集部

PR

