



FORMAT STRING JAVA



El método `Java String format()`

se utiliza para formatear la cadena. Hay muchas cosas que se pueden hacer con este método, por ejemplo, se pueden concatenar los strings usando este método y al mismo tiempo, se puede formatear la salida de un string concatenado. En este tutorial, veremos varios ejemplos del método `Java String format()`



Sintaxis del método format()

Formato de cadena estática pública (Local 1, Formato de cadena, Objeto... args)

Devuelve una cadena formateada usando la localización, la cadena de formato y los argumentos especificados.

Y

```
Public static String format(String format, Object... args)
```

Devuelve una cadena con formato usando la cadena de formato y los argumentos especificados.

Un ejemplo simple del método Java String format()

```
public class Example{ public static void main(String  
args[]){ String str = "just a string"; //concatenating  
string using format String formattedString =  
String.format("My String is %s", str); /*formatting the  
value passed and concatenating at the same time * %.6f es  
por tener 6 dígitos en la parte fraccionaria */ String  
formattedString2 = String.format("My String is  
%.6f",12.121); System.out.println(formattedString);  
System.out.println(formattedString2); }}
```

Salida:

```
Mi cuerda es: cuerda fría, cuerda fría y 88
```

Como puedes ver, como hemos pasado la cadena «cool string» dos veces en el método format() usando el argumento positions format specifiers.



Rellenando la cadena con el argumento format()

En este ejemplo, nos queda rellenar un número con 0 y convertir el número en una cadena formateada. En el ejemplo anterior hemos formateado los números de flotación y las Cadenas y en este ejemplo estamos formateando un número entero. El punto importante que hay que recordar es que los especificadores de formato para estos son diferentes.

%s – para las cadenas

%f – para los flotadores

%d – para números enteros

```
public class Example{ public static void main(String  
args[]){ int str = 88; /* Rellena un número entero con 0 y  
lo convierte * en un String usando el método Java String  
format(). */ String formattedString = String.format("%05d",  
str); System.out.println(formattedString); }}
```

Salida:

00088

Mostrando String, int, hexadecimal, float, char, valor octal usando el método format()

En el siguiente ejemplo, estamos utilizando diferentes especificadores de formato para mostrar valores de diferentes tipos. Aquí hemos mostrado algunos ejemplos de cómo un valor entero puede convertirse en un valor octal o hexadecimal utilizando el método format(). Después de este ejemplo, hemos compartido una lista de especificadores de formato disponibles.

```
public class JavaExample { public static void main(String[]
args) { String str1 = String.format("%d", 15); // Valor
entero String str2 = String.format("%s",
"BeginnersBook.com"); // String String str3 =
String.format("%f", 16.10); // Valor flotante String str4 =
String.format("%x", 189); // Valor hexadecimal String str5
= String.format("%c", 'P'); // Valor de carácter String
str6 = String.format("%o", 189); // Valor octal
System.out.println(str1); System.out.println(str2);
System.out.println(str3); System.out.println(str4);
System.out.println(str5); System.out.println(str6); } }
```

Especificadores de formato de cadena de Java

%c – Carácter

%d – Entero

%s – Cuerda

%o – Octal

%x – Hexadecimal

%f – Número flotante

%h – código hash de un valor

Mensajes relacionados:

Java – izquierda relleno una cadena con espacios y ceros

Java – Relleno derecho de una cadena con espacios y ceros