

Métodos de String

La clase **String** tiene varios métodos muy útiles como saber su longitud, trocear la cadena, etc. Recuerda que para invocar un **método** debemos escribir el nombre del **String**, un punto y el nombre del **método**, más sus **parámetros**. Es importante que si necesitas almacenar el valor devuelto, uses una **variable** para ello. Por ejemplo:

```
String cadena="americano";
```

Estos son los métodos más conocidos:

Método	Descripción	Parámetro	Tipo de dato devuelto
charAt	Devuelve el carácter indicado por parámetro	Un parámetro int	char
compareTo	Sirve para comparar cadenas, devuelve un número según el resultado. Recuerda que no sigue el alfabeto español, lo compara según la tabla ASCII.	Un parámetro String, la cadena a comparar.	int - Si devuelve un número mayor que 0: la primera cadena es mayor que la segunda. - Si devuelve un 0: las cadenas son iguales. - Si devuelve un número menor que 0: la primera cadena es menor que la segunda
compareToIgnoreCase	Es igual que el anterior, pero ignorando mayúsculas o minúsculas.	Un parámetro String, la cadena a comparar	int - Si devuelve un número mayor que 0: la primera cadena es mayor que la segunda. - Si devuelve un 0: las cadenas son iguales. - Si devuelve un número menor que 0:

			la primera cadena es menor que la segunda
concat	Concatena dos cadenas, es como el operador +.	Un parámetro String, la cadena a concatenar	Un nuevo String con las cadenas concatenadas.
endsWith	Indica si la cadena acaba con el String pasado por parámetro.	String	boolean
equals	Indica si una cadena es igual que otra.	String	boolean
equalsIgnoreCase	Es igual que el anterior, pero ignorando mayúsculas o minúsculas.	String	boolean
getBytes	Devuelve un array de bytes con el código ASCII de los caracteres que forman el String.	Ningún parámetro	Un array de bytes
indexOf	Devuelve la posición en la cadena pasada por parámetro desde el principio. -1 si no existe.	String o char	int
indexOf	Igual que el anterior, pero además le indicamos la posición desde donde empezamos a buscar.	String o char, el segundo parámetro es un int	int
lastIndexOf	Devuelve la posición en la cadena pasada por parámetro desde el final. -1 si no existe.	String o char	int
lastIndexOf	Igual que el anterior, pero además le indicamos la posición desde donde empezamos a buscar.	String o char, el segundo parámetro es un int	int
length	Devuelve la longitud de la cadena.	Ningún parámetro	int
matches	Indica si la cadena cumple con la expresión pasada como parámetro.	String	boolean
replace	Devuelve un String cambiando los caracteres que nosotros le indiquemos.	Dos parámetros char, el primero es el carácter que	String

		existe en el String y el segundo por el que queremos cambiar.	
replaceFirst	Devuelve un String intercambiando solo la primera coincidencia.	Dos parametros String, el primero son los caracteres que existe en el String y el segundo por el que queremos cambiar.	String
replaceAll	Devuelve un String intercambiando todas las coincidencias que se encuentren.	Dos parametros String, el primero son los caracteres que existe en el String y el segundo por el que queremos cambiar.	String
startsWith	Indica si la cadena empieza por una cadena pasada por parámetro.	String	boolean
substring	Trocea un String desde una posición a otra.	Dos parámetros int, indica desde donde empieza hasta donde acaba, este ultimo no se incluye.	String
toCharArray	Devuelve en un array de char, todos los caracteres de una String.	Ningún parámetro	Un array de char
toLowerCase	Convierte el String a	Ningún	String

	minúsculas.	parámetro	
toUpperCase	Convierte el String a mayúsculas.	Ningún parámetro	String
trim	Elimina los espacios del String.	Ningún parámetro	String
valueOf	Transforma una variable primitiva en un String. Para invocarse debe usarse con String. Por ejemplo, String.valueOf(variable)	Un parámetro, que puede ser un: boolean char double int float long Array de char Referencia a un objeto	String