

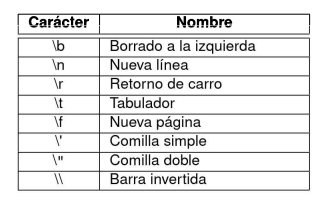
**Tema: Investigación String**

**Profesor: Javier Faus**

**Alumnos: Joaquín, Jetse, Francisco**

**Curso: 1ºDAM**

**Asignatura: Programación.**



Un carácter precedido de una barra invertida (\) se conoce como secuencia de escape.

Se utiliza para poner caracteres especiales.

Java dispone de varios métodos en su librería básica para poder manipular datos de tipo texto, y se los conoce como del tipo ***String***.

Los ***String*** en Java son objetos y tienen una sintaxis especial para representar cadenas de caracteres.

El objeto ***String***, una vez creado su contenido es inmutable, o sea que no se puede modificar.

Cualquier valor de una variable se puede convertir a una cadena mediante el uso de la función String()

Las cadenas son inmutables. Eso significa que una vez que haya creado la cadena, si otro proceso puede volcar la memoria, no hay manera

**¿Para qué se utilizan los Strings en Java?**

Cuando trabajamos con cadenas de texto muchas veces precisamos manipularlas, modificarlas, buscar y extraer una subcadena, sacar informes, convertir a mayúsculas o minúsculas, entre otras muchas cosas.

Con el método ***String*** en Java podemos obtener la ubicación de una letra dentro de la cadena, la longitud que tiene un grupo de caracteres, una porción de una cadena o palabra, ordenarlas de menor a mayor y viceversa, y muchas otras tareas.

## **¿Cómo se utiliza la clase String en Java?**

La forma más común de aplicar ***String*** es declarando el contenido de esta manera:

\*\*String [nombrevariable] = new String (“cadenadecaracteres”);\*\*

Cabe mencionar que si bien un objeto ***String*** es inmutable, existen dos clases derivadas del mismo que se las conoce como **StringBuilder** y **StringBuffer** que permiten crear objetos que pueden ser modificados sin necesidad de crear nuevos.

**StringBuilder** y **StringBuffer**, disponen de los métodos append, replace e insert con los que se pueden manipular cadenas de caracteres.

## **Métodos de la clase** ***String***

Los métodos predeterminados que posee ***String*** nos dan la posibilidad de realizar muchas cosas, aquí presentamos algunos de ellos:

|  |  |
| --- | --- |
| **MÉTODO** | **DESCRIPCIÓN** |
| **length()** | **Nos sirve para conocer la longitud de la cadena** |
| **indexOf(‘caracter’)** | **Nos entrega la ubicación, el índice, de la primera aparición del carácter buscado.** |
| **lastIndexOf(‘caracter’)** | **Nos entrega la ubicación, el índice, de la última aparición del carácter buscado.** |
| **charAt(n)** | **Nos muestra qué carácter se encuentra en la posición solicitada entre paréntesis.** |
| **substring(n1,n2)** | **Este método nos devuelve la subcadena que se encuentra entre las posiciones n1 y n2-1** |
| **toUpperCase()** | **Convierte toda la cadena a mayúsculas** |
| **toLowerCase()** | **Convierte toda la cadena a minúsculas.** |
| **equals(«cad»)** | **Compara dos cadenas y nos informa True si son iguales** |
| **equalsIgnoreCase(«cad»)** | **Compara dos cadenas y nos informa True si son iguales y no considera mayúsculas y minúsculas** |
| **compareTo(OtroString)** | **Uno de los métodos más utilizados, nos devuelve 0 si ambas cadenas son iguales. <0 si la primera es alfabéticamente menor que la segunda ó >0 si la primera es alfabéticamente mayor que la segunda.** |
| **String trim()** | **Devuelve la copia de la cadena, elimina los espacios en blanco en los extremos, pero no afecta los espacios en blanco intermedios.** |
| **valueOf(N)** | **Nos sirve para convertir el valor N a String. N puede ser de cualquier tipo.** |
| **String replace (char oldChar, char newChar)** | **Devuelve una nueva cadena al reemplazar todas las ocurrencias de oldChar con newChar** |

¿El método String trim() elimina los espacios del principio o el final?

Elimina los espacios de ambas.

Elimina solo los espacios del principio

Elimina solo los espacios del final.

No elimina ningún espacio

¿Qué sucede si comparas un string con ”==” con otro string?

Devuelve error la forma de comparar con string es con el método equals().

Devuelve true si son iguales.

Devuelve error la forma de comparar es con compareTo().

Ninguna es correcta.

¿Qué es el charAt?

Seleccionar un carácter

Coger dos caracteres

La 1 y 2 son correctas

Todas son falsas