

La clase java.lang.String

Detalles (I): características relevantes

- **Referencia:** [Documentación de la clase String - API de Java](#)
- **Clase Tipo de Datos del estándar de Java:** representa cualquier **cadena de caracteres** y sus características más importantes son:
 1. Además de vía constructor, se puede **crear un String por asignación directa de un literal** del tipo.



BlueJ: ejercicios - Tema 3

```
String st1 = "Esto es un ejemplo de String";  
String st2 = new String("Esto es un ejemplo de String");  
String st3 = ""; // st3 es un String vacío, que NO null!!  
String st4 = new String(); // Ídem para st4
```

1. **¡Un objeto de tipo String es inmutable!** (una vez creado no es posible alterar su contenido añadiendo, eliminando o cambiando caracteres)
➔ Los métodos que devuelven un String, devuelven un String nuevo
3. **El primer carácter de un String no vacío ocupa su posición 0**
4. Usando el **operador + se concatenan String**, lo que crea un nuevo String a partir de los concatenados

La clase java.lang.String

Detalles (II): métodos relevantes

Método	Significado
length()	Devuelve la longitud (número de caracteres) de un String
trim()	Devuelve el String que resulta de eliminar los espacios en blanco del principio y final de un String
charAt(n)	Devuelve el carácter que ocupa la posición n de un String
substring(i, f) substring(i)	Devuelve el String comprendido entre las posiciones i y f - 1 de un String; sin segundo parámetro, devuelve hasta el final del String
toUpperCase() toLowerCase()	Devuelve el String que resulta de pasar las letras de un String a mayúsculas/minúsculas
indexOf(st) lastIndexOf(st)	Devuelve la posición de la primera/última aparición de st en un String o -1 si no existe
startsWith(prefijo) endsWith(sufijo)	Comprueba si un String empieza/termina por prefijo/sufijo
equals(otro)	Comprueba si un String es igual a otro
compareTo(otro)	Devuelve un nº (int) negativo si un String es menor que otro; uno positivo si un String es mayor que otro; 0 si son iguales (según el orden lexicográfico)

La clase java.lang.String

Ejemplos relevantes (I): igualdad y comparación



BlueJ: ejercicios - Tema 3

```
String s1 = "Ho1a", s2 = "He1lo", s3;  
boolean iguales;  
  
iguales = s1 == s2;           // iguales vale false  
iguales = s1.equals(s2);      // iguales vale false  
  
s3 = s1;  
iguales = s3 == s1;           // iguales vale true  
iguales = s3.equals(s1);      // iguales vale true  
  
int comp1 = s3.compareTo(s1); // comp1 es 0  
int comp2 = s3.compareTo(s2); // comp2 es > 0  
int comp3 = s2.compareTo(s3); // comp3 es < 0
```

La clase java.lang.String

Ejemplos relevantes (II)



BlueJ: ejercicios - Tema 3

```
String st1 = "Ejemplo 1";
String mayus = st1.toUpperCase(); // mayus es "EJEMPLO 1"
String minus = st1.toLowerCase(); // minus es "ejemplo 1"
int longitud = st1.length(); // longitud vale 9
char caracter = st1.charAt(1); // caracter vale 'j'
String sub = st1.substring(3, 5); // sub es "mp"
String st = st1.concat(" y 2"); // st es "Ejemplo 1 y 2"
boolean b = st1.startsWith("Eje"); // b vale true
boolean c = st1.endsWith("Eje"); // c vale false
int inici = st1.indexOf("mp1"); // inicio vale 3
int desde = st1.indexOf("mp1", 2); // desde vale 3
String st2 = " Ejemplo 2 ";
String noBlanco = st2.trim(); // noBlanco es "Ejemplo 2"
int ultima = st2.lastIndexOf(" "); // ultima vale 11
```