Manejo de String

Andrés Guzmán F

Qué son los String

- Los String son objetos de java, del tipo de referencia:

```
String nombre = new String("Andrés");
String tema = new String("Manejo de \"String\"");
```

- Aunque también los String se pueden representar con una sintaxis especialmente cómoda

Qué son los String

- El tipo String tiene una característica especial, permite crear objetos también en la literal entre comillas doble

```
String nombre = "Andrés";
String tema = "Manejo de \"String\"";
```

Qué son los String

- El tipo String tiene una característica especial, permite crear objetos también en la literal entre comillas doble

```
String nombre = "Andrés";
String tema = "Manejo de \"String\"";
```

- Para incluir el carácter comillas dobles, se debe escapar "\"".
- Los caracteres de un String se codifican usando Unicode
- Son inmutables

Concatenar String

- Sobre las cadenas se define la operación de concatenar con el operador de suma

```
String nombre = "Andrés";
String apellido = "Guzmán";

String nombreCompleto = nombre + " " + apellido;
```

Comparar String

- Con operador relacional de igualdad == compara por referencia
- Con el método equals() compara por valor

```
String str1 = "Hola Andres";
String str2 = new String("Hola Andres");

// Chequea si son el mismo objeto
System.out.println("Son el mismo objeto? " + (str1 == str2));

// Chequea si tienen el mismo valor
System.out.println("Tienen el mismo valor? " + str1.equals(str2));
```

Métodos de la clase String

- int length() : número de caracteres
- boolean equals(String b) : compara si ambas son iguales, por valor
- boolean equalsIgnoreCase(String b) compara si ambas son iguales, independientemente de mayúsculas o minúsculas.
- int compareTo(String b) Compara contra la cadena del argumento, devolviendo:
    - un valor negativo si la cadena es anterior a b
    - cero (0) si la cadena es igual a b
    - un valor positivo si la cadena es posterior a b
- String trim() Crea un nuevo objeto eliminado el espacio en blanco que pudiera haber al principio o al final.
- char charAt(int posicion) Extrae un carácter en la posición indicada.

Métodos de la clase String

- char[] toCharArray() Convierte la cadena en un arreglo de caracteres.
- String substring(int a, int b) Extrae la sub-cadena entre las posiciones a y b.
- String substring(int desde) Extrae la sub-cadena desde la posición indicada.
- int indexOf(String cadena) Indican en qué posición se encuentra el carácter (o cadena) indicado por primera vez, buscando desde el principio.
- int lastIndexOf(String cadena) Indica en qué posición se encuentra el carácter (o cadena) indicado por primera vez, buscando desde el final.
- boolean startsWith(String prefijo) Dice si la cadena comienza con el prefijo indicado.
- boolean endsWith(String sufijo) Dice si la cadena termina con el sufijo indicado.
- String[] split(String patron) Divide la cadena en varias subcadenas utilizando el patrón indicado como separador.