

2ª PARTE

Clase Java StringBuffer

La clase Java StringBuffer se usa para crear una cadena mutable (modificable). La clase StringBuffer en java es la misma que la clase String, excepto que es mutable, es decir, se puede cambiar.

Nota: La clase Java StringBuffer es segura para subprocesos, es decir, varios subprocesos no pueden acceder a ella simultáneamente. Por lo tanto, es seguro y dará como resultado un pedido.

Constructores importantes de la clase StringBuffer

Constructor	Descripción
StringBuffer ()	crea un búfer de cadena vacío con la capacidad inicial de 16.
StringBuffer (String str)	crea un búfer de cadena con la cadena especificada.
StringBuffer (int capacidad)	crea un búfer de cadena vacío con la capacidad especificada como longitud.

Métodos importantes de la clase StringBuffer

Modificador y Tipo	Método	Descripción
StringBuffer sincronizado público	agregar (String s)	se utiliza para agregar la cadena especificada con esta cadena. El método append () se sobrecarga como append (char), append (boolean), append (int), append (float), append (double) etc.
StringBuffer sincronizado público	insert (int offset, String s)	se usa para insertar la cadena especificada con esta cadena en la posición especificada. El método insert () se sobrecarga como insert (int, char), insert (int, boolean), insert (int, int), insert (int, float), insert (int, double) etc.
StringBuffer sincronizado público	reemplazar (int startIndex, int endIndex, String str)	se usa para reemplazar la cadena de startIndex y endIndex especificados.
StringBuffer sincronizado público	eliminar (int startIndex, int endIndex)	se usa para eliminar la cadena de startIndex y endIndex especificados.
StringBuffer sincronizado público	marcha atrás()	se usa para invertir la cadena.

int público	capacidad()	se usa para devolver la capacidad actual.
vacío público	sureCapacity (int minimalCapacity)	se utiliza para garantizar la capacidad al menos igual al mínimo dado.
char público	charAt (int int)	se usa para devolver el carácter en la posición especificada.
int público	longitud()	se utiliza para devolver la longitud de la cadena, es decir, el número total de caracteres.
Cadena pública	subcadena (int beginIndex)	se utiliza para devolver la subcadena desde el índice de inicio especificado.
Cadena pública	subcadena (int beginIndex, int endIndex)	se usa para devolver la subcadena del beginIndex y endIndex especificados.

¿Qué es una cadena mutable?

Una cadena que se puede modificar o cambiar se conoce como cadena mutable. Las clases StringBuffer y StringBuilder se usan para crear cadenas mutables.

Clase Java StringBuilder

La clase Java StringBuilder se usa para crear cadenas mutables (modificables). La clase Java StringBuilder es la misma que la clase StringBuffer, excepto que no está sincronizada. Está disponible desde JDK 1.5.

Constructores importantes de la clase StringBuilder

Constructor	Descripción
StringBuilder ()	crea un generador de cadenas vacío con una capacidad inicial de 16.
StringBuilder (String str)	crea un generador de cadenas con la cadena especificada.
StringBuilder (longitud int)	crea un generador de cadenas vacío con la capacidad especificada como longitud.

Métodos importantes de la clase StringBuilder

Método	Descripción
StringBuilder público append (String s)	se utiliza para agregar la cadena especificada con esta cadena. El método append () se sobrecarga como append (char), append (boolean), append (int), append (float), append

	(double) etc.
inserción pública de StringBuilder (int offset, String s)	se usa para insertar la cadena especificada con esta cadena en la posición especificada. El método insert () se sobrecarga como insert (int, char), insert (int, boolean), insert (int, int), insert (int, float), insert (int, double) etc.
Reemplazo público de StringBuilder (int startIndex, int endIndex, String str)	se usa para reemplazar la cadena de startIndex y endIndex especificados.
eliminar StringBuilder público (int startIndex, int endIndex)	se usa para eliminar la cadena de startIndex y endIndex especificados.
public StringBuilder reverse ()	se usa para invertir la cadena.
capacidad int pública ()	se usa para devolver la capacidad actual.
public void allowCapacity (int minimalCapacity)	se utiliza para garantizar la capacidad al menos igual al mínimo dado.
char público charAt (int int)	se usa para devolver el carácter en la posición especificada.
public int length ()	se utiliza para devolver la longitud de la cadena, es decir, el número total de caracteres.
subcadena de cadena pública (int beginIndex)	se utiliza para devolver la subcadena desde el índice de inicio especificado.
subcadena de cadena pública (int beginIndex, int endIndex)	se usa para devolver la subcadena del beginIndex y endIndex especificados.

Ejemplos de Java StringBuilder

Veamos los ejemplos de diferentes métodos de la clase StringBuilder.

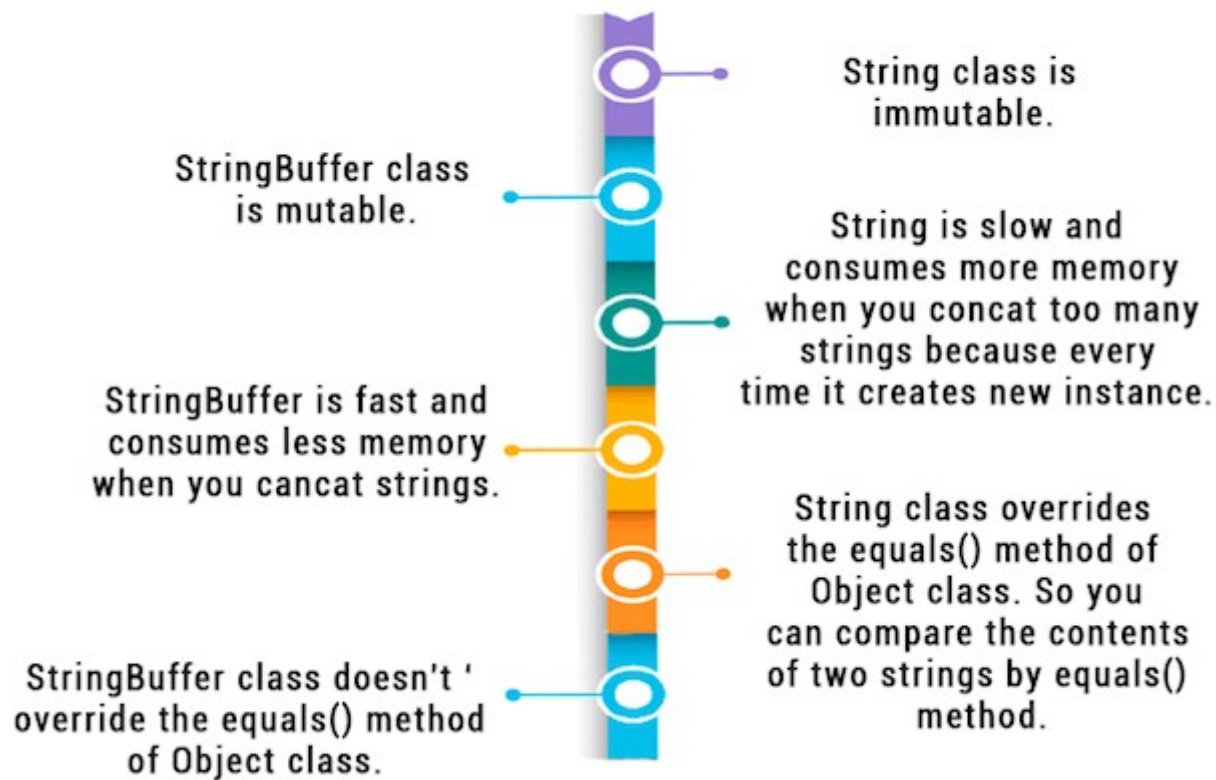
Diferencia entre String y StringBuffer

Hay muchas diferencias entre String y StringBuffer. A continuación se muestra una lista de diferencias entre String y StringBuffer:

No.	Cuerda	StringBuffer
-----	--------	--------------

1)	La clase de cadena es inmutable.	La clase StringBuffer es mutable.
2)	La cadena es lenta y consume más memoria cuando se combinan demasiadas cadenas porque cada vez que crea una nueva instancia.	StringBuffer es rápido y consume menos memoria cuando puede capturar cadenas.
3)	La clase de cadena anula el método equals () de la clase Object. Entonces puede comparar el contenido de dos cadenas por el método equals ().	La clase StringBuffer no anula el método equals () de la clase Object.

StringBuffer vs String



Prueba de rendimiento de String y StringBuffer

Diferencia entre StringBuffer y StringBuilder

Java proporciona tres clases para representar una secuencia de caracteres: String, StringBuffer y StringBuilder. La clase String es una clase inmutable, mientras que las clases StringBuffer y StringBuilder son mutables. Hay muchas diferencias entre StringBuffer y StringBuilder. La clase StringBuilder se introduce desde JDK 1.5.

A continuación se muestra una lista de diferencias entre StringBuffer y StringBuilder:

No.	StringBuffer	StringBuilder
1)	StringBuffer está <i>sincronizado</i> , es decir, es seguro para subprocesos. Significa que dos hilos no pueden llamar a los métodos de StringBuffer simultáneamente.	StringBuilder <i>no</i> está <i>sincronizado</i> , es decir, no es seguro para subprocesos. Significa que dos hilos pueden llamar a los métodos de StringBuilder simultáneamente.
2)	StringBuffer es <i>menos eficiente</i> que StringBuilder.	StringBuilder es <i>más eficiente</i> que StringBuffer.

StringBuffer

vs

StringBuilder

StringBuffer is synchronized i.e. thread safe. It means two threads can't call the methods of StringBuffer simultaneously.

StringBuilder is non-synchronized i.e. not thread safe. It means two threads can call the methods of StringBuilder simultaneously.

StringBuffer is less efficient than StringBuilder.

StringBuilder is more efficient than StringBuffer.

Método Java toString ()

Si desea representar cualquier objeto como una cadena, existe el **método toString ()** .

El método toString () devuelve la representación de cadena del objeto.

Si imprime algún objeto, el compilador de Java invoca internamente el método toString () en el objeto. Por lo tanto, anular el método toString (), devuelve el resultado deseado, puede ser el estado de un objeto, etc., depende de su implementación.

Ventaja del método Java toString ()

Al anular el método toString () de la clase Object, podemos devolver valores del objeto, por lo que no necesitamos escribir mucho código.