

JAVA.LANG.STRINGBUILDER.ENSURECAPACITY METHOD

http://www.tutorialspoint.com/java/lang/stringbuilder_ensurecapacity.htm

Copyright © tutorialspoint.com

Description

The **java.lang.StringBuilder.ensureCapacity** method ensures that the capacity is at least equal to the specified minimum. If the current capacity is less than the argument, then a new internal array is allocated with greater capacity. The new capacity is the larger of:

- The **minimumCapacity** argument.
- Twice the old capacity, plus 2.

If the **minimumCapacity** argument is nonpositive, this method takes no action and just returns.

Declaration

Following is the declaration for **java.lang.StringBuilder.ensureCapacity** method

```
public void ensureCapacity(int minimumCapacity)
```

Parameters

- **minimumCapacity** -- This is the minimum desired capacity.

Return Value

This method does not return any value.

Exception

- NA

Example

The following example shows the usage of java.lang.StringBuilder.ensureCapacity method.

```
package com.tutorialspoint;

import java.lang.*;

public class StringBuilderDemo {

    public static void main(String[] args) {

        StringBuilder str1 = new StringBuilder("tuts point");
        System.out.println("string1 = " + str1);

        // returns the current capacity of the string Builder 1
        System.out.println("Old Capacity = " + str1.capacity());
        /* increases the capacity, as needed, to the specified amount in the
        given StringBuilder object */
        // returns twice the capacity plus 2
        str1.ensureCapacity(28);
        System.out.println("New Capacity = " + str1.capacity());

        StringBuilder str2 = new StringBuilder("compile online");
        System.out.println("string2 = " + str2);
        // returns the current capacity of string Builder 2
        System.out.println("Old Capacity = " + str2.capacity());
        /* returns the old capacity as the capacity ensured is less than the
        old capacity */
        str2.ensureCapacity(29);
        System.out.println("New Capacity = " + str2.capacity());
```

```
}  
}
```

Let us compile and run the above program, this will produce the following result:

```
string1 = tuts point  
Old Capacity = 26  
New Capacity = 54  
string2 = compile online  
Old Capacity = 30  
New Capacity = 30
```

Loading [MathJax]/jax/output/HTML-CSS/jax.js