JAVA.LANG.STRINGBUFFER.APPEND METHOD

Description

The **java.lang.StringBuffer.append*CharSequences, intstart, intend*** method appends a subsequence of the specified **CharSequence** to this sequence.
Characters of the argument **s**, starting at index **start**, are appended, in order, to the contents of this sequence up to the *exclusive* index **end**. The length of this sequence is increased by the value of **end - start**.

Declaration

Following is the declaration for **java.lang.StringBuffer.append** method

```
public StringBuffer append(CharSequence s, int start, int end)
```

Parameters

- **s** -- This is the sequence to append.
- **start** -- This is the starting index of the subsequence to be appended..
- **end** -- This is the end index of the subsequence to be appended.

Return Value

This method returns a reference to this object.

Exception

- **IndexOutOfBoundsException** -- if start or end are negative, or start is greater than end or end is greater than s.length.

Example

The following example shows the usage of java.lang.StringBuffer.append method.

```
package com.tutorialspoint;

import java.lang.*;

public class StringBufferDemo {

   public static void main(String[] args) {

   StringBuffer buff = new StringBuffer("tutorials ");
   System.out.println("buffer = " + buff);

   CharSequence cSeq = "tutspoint";
   // appends the CharSequence with start index 4 and end index 9
   buff.append(cSeq, 4, 9);

   // print the string buffer after appending
   System.out.println("After append = " + buff);
   }
}
```

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

```
buffer = tutorials
After append = tutorials point
```