

Java StringBuffer 和 StringBuilder 类

当对字符串进行修改的时候，需要使用 StringBuffer 和 StringBuilder 类。

和 String 类不同的是，StringBuffer 和 StringBuilder 类的对象能够被多次的修改，并且不产生新的未使用对象。

StringBuilder 类在 Java 5 中被提出，它和 StringBuffer 之间的最大不同在于 StringBuilder 的方法不是线程安全的（不能同步访问）。

由于 StringBuilder 相较于 StringBuffer 有速度优势，所以多数情况下建议使用 StringBuilder 类。然而在应用程序要求线程安全的情况下，则必须使用 StringBuffer 类。

Test.java 文件代码：

```
public class Test{
    public static void main(String args[]){
        StringBuffer sBuffer = new StringBuffer("菜鸟教程官网：");
        sBuffer.append("www");
        sBuffer.append(".runoob");
        sBuffer.append(".com");
        System.out.println(sBuffer);
    }
}
```

以上实例编译运行结果如下：

```
菜鸟教程官网：www.runoob.com
```

StringBuffer 方法

以下是 StringBuffer 类支持的主要方法：


序号	方法描述
1	public StringBuffer append(String s) 将指定的字符串追加到此字符序列。
2	public StringBuffer reverse() 将此字符序列用其反转形式取代。
3	public delete(int start, int end) 移除此序列的子字符串中的字符。
4	public insert(int offset, int i) 将 int 参数的字符串表示形式插入此序列中。
5	replace(int start, int end, String str)


使用给定 `String` 中的字符替换此序列的子字符串中的字符。

下面的列表里的方法和 `String` 类的方法类似：

序号	方法描述
1	<code>int capacity()</code> 返回当前容量。
2	<code>char charAt(int index)</code> 返回此序列中指定索引处的 <code>char</code> 值。
3	<code>void ensureCapacity(int minimumCapacity)</code> 确保容量至少等于指定的最小值。
4	<code>void getChars(int srcBegin, int srcEnd, char[] dst, int dstBegin)</code> 将字符从此序列复制到目标字符数组 <code>dst</code> 。
5	<code>int indexOf(String str)</code> 返回第一次出现的指定子字符串在该字符串中的索引。
6	<code>int indexOf(String str, int fromIndex)</code> 从指定的索引处开始，返回第一次出现的指定子字符串在该字符串中的索引。
7	<code>int lastIndexOf(String str)</code> 返回最右边出现的指定子字符串在此字符串中的索引。
8	<code>int lastIndexOf(String str, int fromIndex)</code> 返回 <code>String</code> 对象中子字符串最后出现的位置。
9	<code>int length()</code> 返回长度（字符数）。
10	<code>void setCharAt(int index, char ch)</code> 将给定索引处的字符设置为 <code>ch</code> 。
11	<code>void setLength(int newLength)</code> 设置字符序列的长度。
12	<code>CharSequence subSequence(int start, int end)</code> 返回一个新的字符序列，该字符序列是此序列的子序列。
13	<code>String substring(int start)</code> 返回一个新的 <code>String</code> ，它包含此字符序列当前所包含的字符子序列。

14	String substring(int start, int end) 返回一个新的 <code>String</code> , 它包含此序列当前所包含的字符子序列。
15	String toString() 返回此序列中数据的字符串表示形式。

 **3 篇笔记**

 **写笔记**