两者的区别：

1. string 对象时恒定不变的，stringBuider对象表示的字符串是可变的。stringBuilder是.net提供的动态创建string的高效方式，以克服string对象恒定性带来的性能影响。

2. 对于简单的字符串连接操作，在性能上stringBuilder并不一定总是优于string。因为stringBuider对象创建代价较大，在字符串目标连接较少的情况下，过度滥用stringBuilder会导致性能的浪费，只有大量的或者无法预知次数的字符串操作，才考虑stringBuilder来实现。事实上，一般连接次数设置100次以内，根本看不出两者的性能差别。

3. 当修改字符串信息时，此时不许创建对象，可以使用stringBuilder对象。

String 对象是不可改变的。每次使用 System.String 类中的方法之一时，都要在内存中创建一个新的字符串对象，这就需要为该新对象分配新的空间。

例如：   string a="a";a+="b";,每次在后面追加都会重新申请一个能放字符串的内存空间；

string Interning(字符串驻留)指的是通过维护一张表来存放字符串。CLR内部维护了一个哈希表（Hash Table）来管理其创建的大部分string对象，其中key为string本身，而value为分配给对应string的内存地址。

public static string Intern(string str);

public static string IsInterned(string str);

两者的处理机制都是在哈希表中查找是否存在str参数字符串，如果找到就返回已存在的string对象的引用，没有找到，Intern方法将该str字符串添加到哈希表，并返回引用；而IsInterned方法则不会向哈希表中添加字符串，而是返回null;

StringBuilder 对象是动态对象，允许扩充它所封装的字符串中字符的数量，但是您可以为它可容纳的最大字符数指定一个值，当修改 StringBuilder 时，在达到容量之前，它不会为其自己重新分配空间。当达到容量时，将自动分配新的空间且容量翻倍。可以使用重载的构造函数之一来指定 StringBuilder 类的容量。

例如： StringBuilder sb = new StringBuilder(); sb.Append("a")他不会频繁申请内存空间，他会自动向后扩展。

代码在Java的运行程序里面，这个是作为辅助说明用的文档。

以上代码在 NetBeans 5.0 IDE/JDK1.6 上编译通过,循环次数 ttime 为 10000 次的测试结果如下：   
操作 java.lang.String 类型使用的时间为： 4392 毫秒   
操作 java.lang.StringBuffer 类型使用的时间为： 0 毫秒   
操作 java.lang.StringBuilder 类型使用的时间为： 0 毫秒   
操作字符串[对象引用](https://www.baidu.com/s?wd=%E5%AF%B9%E8%B1%A1%E5%BC%95%E7%94%A8&tn=SE_PcZhidaonwhc_ngpagmjz&rsv_dl=gh_pc_zhidao" \t "https://zhidao.baidu.com/question/_blank)相加类型使用的时间为： 15 毫秒   
操作字符串相加使用的时间为： 0 毫秒   
  
好像还看不出 StringBuffer 和 StringBuilder 的区别，把 ttime 加到 30000 次看看：   
操作 java.lang.String 类型使用的时间为： 53444 毫秒   
操作 java.lang.StringBuffer 类型使用的时间为： 15 毫秒   
操作 java.lang.StringBuilder 类型使用的时间为： 15 毫秒   
操作字符串对象引用相加类型使用的时间为： 31 毫秒   
操作字符串相加使用的时间为： 0 毫秒   
  
StringBuffer 和 StringBuilder 的性能上还是没有太大的差异，再加大到 100000 看看，这里就不加入对 String 类型的测试了，因为对 String 类型这么大数据量的测试会很慢滴……   
操作 java.lang.StringBuffer 类型使用的时间为： 31 毫秒   
操作 java.lang.StringBuilder 类型使用的时间为： 16 毫秒   
  
能看出差别了，但其中有多次的测试结果居然是 StringBuffer 比 StringBuilder 快，再加大一些到 1000000 看看（应该不会当机吧？）：   
操作 java.lang.StringBuffer 类型使用的时间为： 265 毫秒   
操作 java.lang.StringBuilder 类型使用的时间为： 219 毫秒   
  
有些少区别了，而且结果很稳定，再大点看看， ttime = 5000000 ：   
  
······ Exception in thread "main" java.lang.OutOfMemoryError: Java heap space ······   
  
呵呵，算了，不去测试了，基本来说都是在性能上都是 StringBuilder > StringBuffer > String 的了。