

陈新月-Phyllis

TA的个人主页 >

私信

关注

原创

粉丝

获赞

评论

访问

177

155

63

3786

11万+

等级: 4

周排名: 1868

积分: 1万+

总排名: 3117

勋章

最新文章

归档

展开

热门文章

最新评论

积分: 1万+

总排名: 3117

勋章

最新文章

归档

展开

热门文章

最新评论

原创 | String、StringBuffer和StringBuilder的区别

2019-06-16 10:31:19 陈新月-Phyllis 阅读量 6909 更多

版权声明：本文为博主原创文章，遵循 CC 4.0 BY-SA 版权协议，转载请附上原文出处链接和本声明。
本文链接：<https://blog.csdn.net/csxypr/article/details/92378336>

String、StringBuffer和StringBuilder的区别：

文章目录

- [String](#)
- [StringBuffer](#)
- [StringBuilder](#)
 - [StringBuffer是如何实现线程安全的呢？](#)
 - [Java9的改进](#)

String

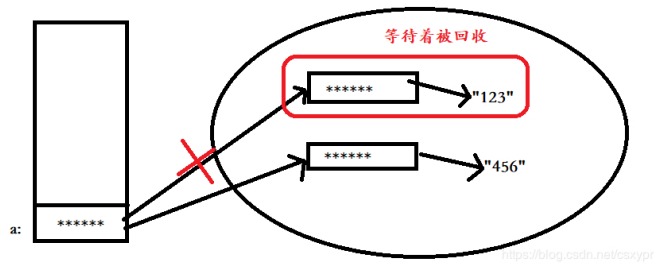
String类是不可变类，即一旦一个String对象被创建以后，包含在这个对象中的字符序列是不可改变的，直至这个对象被销毁。

这个是String类的解释，之前小威儿看到这个情况，不能理解上述的解释，如下

```
1 String a = "123";
2 a = "456";
3 // 打印出来的a为456
4 System.out.println(a)
```

看到这里，小威儿不明白了，这不是明明已经对他进行修改了吗？为什么还说他是一个不可变类呢？

经过小威儿和小伙伴们的学习，明白String类不可变在哪里体现出来的，接下来就看一张上述a对象的内存存储空间图



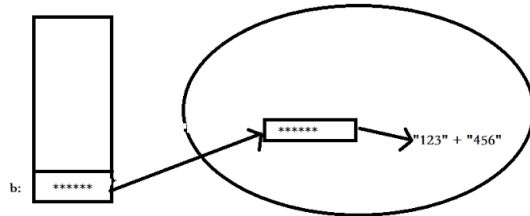
可以看出来，再次给a赋值时，并不是对原来堆中实例对象进行重新赋值，而是生成一个新的实例对象，并且指向“456”这个字符串，a则指向最新生成的实例对象，之前的实例对象仍然存在，如果没有被再次引用，则会被垃圾回收。

StringBuffer

StringBuffer对象则代表一个字符序列可变的字符串，当一个StringBuffer被创建以后，通过StringBuffer提供的append()、insert()、reverse()、setCharAt()、setLength()等方法可以改变这个字符串对象的字符序列。一旦通过StringBuffer生成了最终想要的字符串，就可以调用它的toString()方法将其转换为一个String对象。

```
1 StringBuffer b = new StringBuffer("123");
2 b.append("456");
3 // b打印结果为: 123456
4 System.out.println(b);
```

在看一下b对象的内存空间图：



<https://blog.csdn.net/csxypr>

所以说StringBuffer对象是一个字符序列可变的字符串，它没有重新生成一个对象，而且在原来的对象中可以连接新的字符串。

StringBuilder

StringBuilder类也代表可变字符串对象。实际上，StringBuilder和StringBuffer基本相似，两个类的构造器和方法也基本相同。不同的是：StringBuffer是线程安全的，而StringBuilder则没有实现线程安全功能，所以性能略高。

StringBuffer是如何实现线程安全的呢？

StringBuffer类中实现的方法：

```
@Override
public synchronized int length() { return count; }
```

```
@Override
public synchronized int capacity() { return value.length; }

@Override
public synchronized void ensureCapacity(int minimumCapacity) { super.ens
```

StringBuilder类中实现的方法：

```
@Override
public synchronized String replace(int start, int end, String str) {
    super.replace(start, end, str);
    return this;
}
```

由此可见，StringBuffer类中的方法都添加了**synchronized关键字**，也就是给这个方法添加了一个锁，用来保证线程安全。

Java9的改进

Java9改进了字符串（包括String、StringBuffer、StringBuilder）的实现。在Java9以前字符串采用char[]数组来保存字符，因此字符串的每个字符占2字节；而Java9的字符串采用byte[]数组再加一个encoding-flag字段来保存字符，因此字符串的每个字符只占1字节。所以Java9的字符串更加节省空间，字符串的功能方法也没有受到影响。

文章最后发布于: 2019-06-16 10:31:19

有 0 个人打赏

私信求助

想对作者说点什么

Hany_Ji

1周前

#15楼

感谢博主！！

苏晓Lucy

4个月前

#14楼

学习了，感谢分享

苏晓Lucy

4个月前

#13楼

非常优秀的博主

查看 17 条热评

python json java mysql pycharm android linux json格式 c# 增加元素 c# 控制台简单加法 c# 服务端框架 c# 判断事件是否注册 c#中is和has c# udp 连接超时 c#词典 c#实现排列组合 c# oss 上传 c#判断输入的是否为ip

没有更多推荐了，[返回首页](#)