**StringBuffer类**

字符串缓冲区

StringBuffer是线程**安全**的**可变字符串**，StringBuilder是不同步不安全**效率高**的**可变字符串**，String的**内容和长度不可变**，用StringBuffer做拼接不会浪费空间

**构造方法：**

StringBuffer()

StringBuffer(int size) 指定容量的字符串缓冲区对象

StringBuffer(String str) 指定字符串内容的字符串缓冲区对象

Public int capacity() 返回**当前容量**

Public int length() 返回字符数**实际长度**

**方法：**

**添加功能**

public StringBuffer append(String str) 可以把任意类型数据追加到字符串缓冲区里面

public StringBuffer insert(int index,String sttr) 在指定位置把任意类型的数据插入到字符串缓冲区里面

**删除功能**

public StringBuffer deleteCharAt(int index) 删除指定位置的字符

public StringBuffer delete(int start,int end) 删除从指定位置开始到指定位置结束的内容

**替换功能**

public StringBuffer replace(int start,int end,String str) 从start位置开始到end用str替换

**反转功能**

public StringBuffer reverse() 把对象内容反转

**截取功能**

public String substring(int start) 从start位置开始到末尾返回该截取字符串

public String substring(int start,int end) 从start开始到end截取字符串并返回

1. **System类**

public static long currentTimeMillis() 返回当前时间（毫秒）

System.arraycopy(arr,int s,arr2,s1,int length) 把数组arr从下标s开始复制length个元素到数组arr2的下标s1后

System.exit(0) 非0表示异常终止

1. **Date类**

**方法：**

getTime() 获取当前毫秒时间

setTime(long time) 设置时间

1. **DateFormat抽象类的子类SimpleDateFormat**

实现日期到字符串的格式化和解析

**构造方法：**

SimpleDateFormat()

SimpleDateFormat(String pattern)

**用法：**

**将日期d格式化为字符串**

Date d=new Date();

SimpleDateFormat s=new SimpleDateFormat();

SimpleDateFormat sd=new SimpleDateFormat(“yyyy年MM月dd日:mm:ss”);

String s1=s.format(d); **//格式化，输出19-08-11 下午3：10**

String s2=sd.format(d); **//格式化，2019年08月10日:12:12**

**将字符串解析为日期**

String sr=”**2019-08-10 12:12:12**”;**//格式对齐**

SimpleDateFormat sd=new SimpleDateFormat(“**yyyy-MM-dd HH:mm:ss**”);

Date dd=sd.parse(sr);**//解析sr**

**Toolkit类**

**工具包类、抽象类**

**Toolkit.getDefaultToolkit() 用于获取一个子类对象**

**getImage(String directory) 根据图片的路径获取图片**

**getScreenSize() 获取屏幕的宽和高**

1. **UIManager类**

**UIManager.setLookAndFeel(String s) 设置窗体皮肤，在main方法第一条语句**

1. **Runtime类**

**单例设计模式的例子**

public Process exec(String command)

**Runtime.getRuntime().exec(“notepad”)**