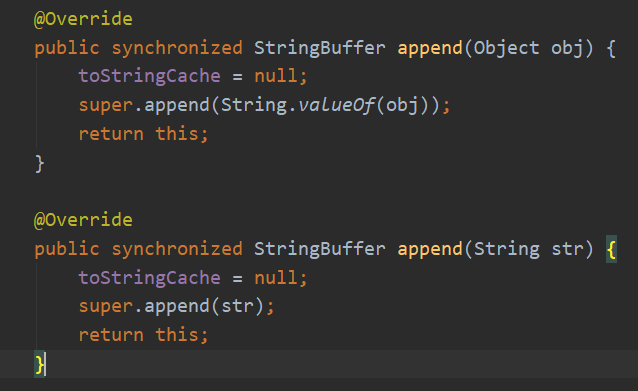
# Java 基础知识

1. JDK JRE JVM 的区别:
   1. 概念:
      1. JDK指的是Java 开发工具
      2. JRE指的是java运行环境
      3. JVM指的是java 虚拟机
   2. JDK 中包括JRE ,JRE中包括JVM
2. 基础知识
   1. 常量:
      1. 常量指的是不可以变的量
   2. 变量:
      1. 指的是内存中开辟的一块空间,空间会指向所对应的常量
   3. 注释:
      1. Java 中的注释分为 单行注释 多行注释 文档注释
         1. 单行注释: 使用 // 进行标注,
3. 基本数据类型
   1. Byte :
   2. Short :
   3. Int :
   4. Long :
   5. Float :
   6. Double :
   7. Char :
   8. Boolean:
4. String 类相关知识 (重要)
   1. String
      1. String 类型表示的是字符串
      2. String 类型的声明为final,不可以被继承
      3. 字符串的String 使用的是 Unicode 进行编码,一个字符占两个字节
      4. String实现了Serializable接口和Comparable接口
         1. Serializable接口：表示字符串是支持序列化的，说明可以在网络上传输.
         2. Comparable接口：表示String可以比较大小 (可以使用使用compareTo的方法进行比较)
      5. String类内部定义了final char[] value用于存储字符串数据
      6. 一定要注意：value是一个final类型，不可以修改：即value不能指向新的地址，但是单个字符内容是可以变化的。
      7. String代表不可变的字符序列，简称：不可变性
         1. 体现：
            1. 当对字符串重新赋值时，需要重新指定内存区域赋值，不能使用原有的value进行赋值。
            2. 当对现有的字符串进行连接操作时，也需要重新指定内存区域赋值，不能使用原有的value进行赋值。
            3. 当调用String的replace()方法修改指定字符或字符串时，也需要重新指定内存区域赋值，不能使用原有的value进行赋值。
      8. 通过字面量的方式(区别于new)给一个字符串赋值，此时的字符串值声明在字符串常量池中
      9. 字符串常量池中是不会存储相同内容的字符串的

总结: String 类型是一个表示字符串的最终类,它自己表示的是一个不可变的序列,通过将字面量进行赋值,会从原先的常量池地址间接/直接的指向一个新的常量池地址值.

* 1. StringBuffer (线程安全):
     1. StringBuffer 类是可变的字符串类
     2. 每个 StringBuffer 类的对象都能够存储指定容量的字符串，如果字符串的长度超过了 StringBuffer 类对象的容量，则该对象的容量会自动扩大
     3. 说StringBuffer 线程安全,是因为StringBuffer 的内部对长度添加了线程锁,多个线程中中能有一个线程进行执行.



* 1. StringBuilder (线程不安全)
     1. StringBuilder 是可变的字符串序列
     2. StringBuilder 非线程安全
  2. String StringBuffer StringBuilder 的区别:
     1. <https://blog.csdn.net/pf6668/article/details/108875324>
     2. <https://blog.csdn.net/lu202032/article/details/117047712>

1. 数组
   1. 创建数组的方式
      1. Int [] array = new int[]{11,12,13};
      2. Int [] array = {2,13,14};
      3. Int [] array = new int[4];
   2. 三种方式创建数组的区别
      1. 第一种方式 :属于动态创建,在这里声明并且创建了动态空间,并且为空间中的值进行赋予
      2. 第二种方式 :数据静态创建,直接声明并且进行赋值,创建空间
      3. 第三种方式 : 属于动态创建,在这里声明并且创建了内存空间,空间中的值为默认值,等待之后赋值

<https://blog.csdn.net/java_xiaoo/article/details/127190975>

1. 运算符
   1. 算数运算符
      1. + - \* / ++ --
   2. 逻辑运算符
      1. ! ^ && ||
   3. 赋值运算符
      1. = += -= \*= /=
   4. 位运算符
      1. & | ~ ^ << >> >>>
   5. 比较运算符
      1. > < >= <= !=
   6. 条件运算符
      1. 三元一次运算符 ? :
2. 表达式语句:
   1. 循环语句
      1. For循环
      2. 增强For循环(foreach)
   2. Switch语句
   3. If语句
3. 关键字
   1. Final:
   2. Break
   3. Return
   4. Continue
   5. Static
4. 类与对象
5. 面向对象
   1. 封装
   2. 继承
   3. 多态
6. 接口
7. 异常
8. 数据结构
9. 数组
   1. ArrayList
   2. LinkedList
   3. HashSet
   4. HashMap
10. 泛型
11. IO流对象
    1. File 类
    2. IO流对象
       1. 输入流
       2. 输出流
12. 序列化与反序列化
13. 多线程
    1. Thread
    2. Runnable
14. 网络编程
    1. Sockect