# Integer类

## 类比

Double、功能类似

## Integer类

Integer 类在对象中包装了一个基本类型 int 的值,

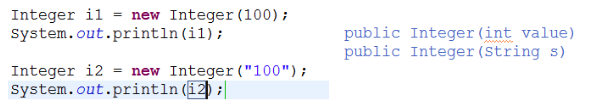
### 常量





### 构造方法

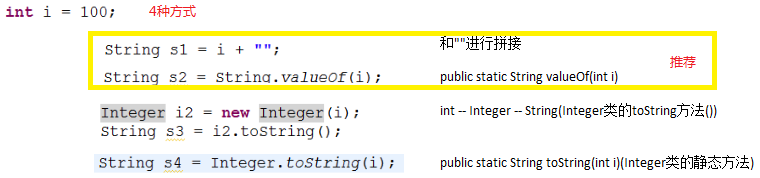
* 可以使用原始类型作为参数，实例化相应的包装类对象。



## String和int类型的相互转换

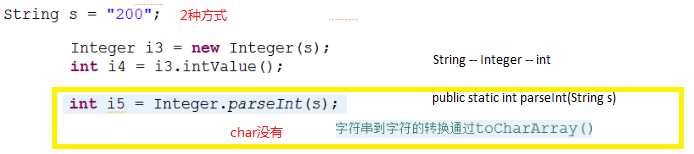
### int 转String(4种)

valueOf方法：每个包装类都有一个静态的valueOf方法，用于将字符串转换成相应包装类的对象。



### String 转 int

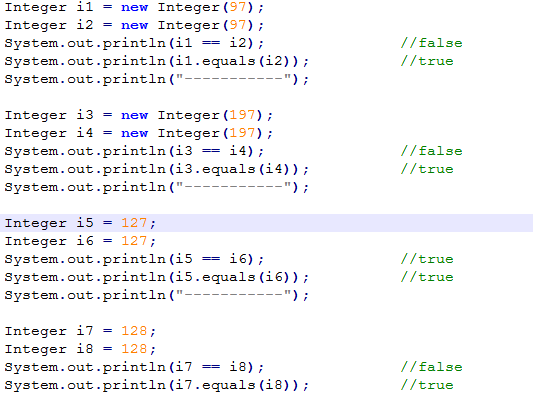
除了Boolean类和Character类以外，其它的包装类都有静态的parseXxx方法（Xxx指代具体的数据类型），用于将字符串转换成相对应的原始数据类型值。

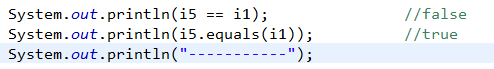


# 注意

## 为什么重写equals()方法为什么要重写hashCode()方法

https://www.jianshu.com/p/3819388ff2f4





-128到127是byte的取值范围,如果在这个取值范围内,自动装箱就不会新创建对象,而是从常量池中获取如果超过了byte取值范围就会再新创建对象

总结：

1.无论如何，Integer与new Integer不会相等。不会经历拆箱过程，i3的引用指向堆，而i4指向专门存放他的内存（常量池），他们的内存地址不一样，所以为false  
   2.两个都是非new出来的Integer，如果数在-128到127之间，则是true,否则为false  
  java在编译Integer i2 = 128的时候,被翻译成-> Integer i2 = Integer.valueOf(128);而valueOf()函数会对-128到127之间的数进行缓存  
   3.两个都是new出来的,都为false  
  4.int和integer(无论new否)比，都为true，因为会把Integer自动拆箱为int再去比