# Object类

## 概述

面向对象的核心思想:“找合适的对象，做适合的事情”。

合适的对象：

1. 自己描述类，自己创建对象。
2. sun已经描述了好多常用的类，可以使用这些类创建对象。

API（Application Program Interface）

sun定义的那么多类的终极父类是Object。object类是所有java类的根基类。所有类都直接或者间接的继承自该类。

:构造方法

\* public Object()

\* 回想面向对象中为什么说：

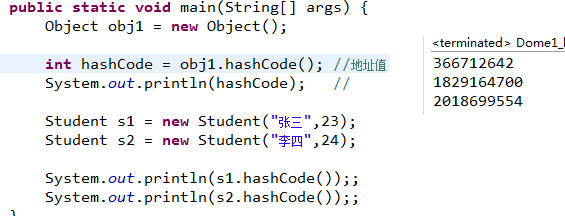
\* 子类的构造方法默认访问的是父类的无参构造方法

## hashCode()方法

public int hashCode()

返回该对象的哈希码值。默认情况下，该方法会根据对象的地址来计算。

不同对象的，hashCode()一般来说不会相同。但是，同一个对象的hashCode()值肯定相同。



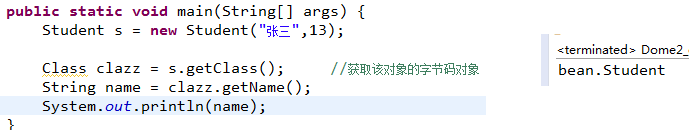
## getClass()

public final Class getClass()

返回此 Object 的运行时类。

可以通过Class类中的一个方法，获取对象的真实类的全名称。

public String getName()



## toString方法

public String toString()

a:返回该对象的字符串表示。

public Stirng toString() {

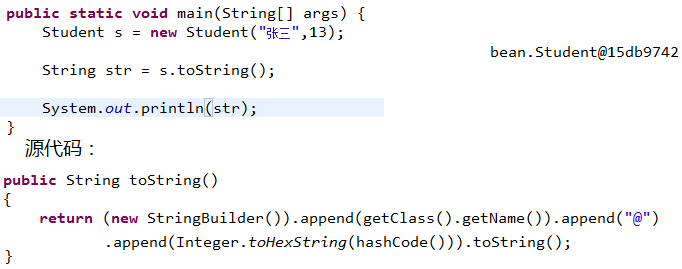
return name + "," + age;

}

b:它的值等于：

getClass().getName() + "@" + Integer.toHexString(hashCode())

c:由于默认情况下的数据对我们来说没有意义，一般建议重写该方法。



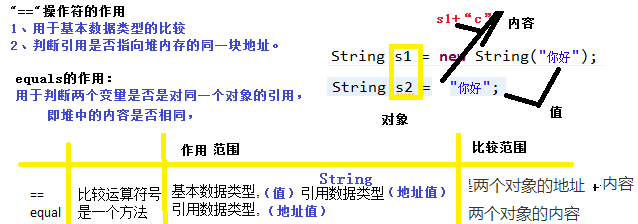
## equals方法

a:指示其他某个对象是否与此对象“相等”。

b:默认情况下比较的是对象的引用是否相同。

c:由于比较对象的引用没有意义，一般建议重写该方法。

补充：==号和equals方法的区别



# Scanner类--从命令行输入、输出数据

## 创建输入对象：

Scanner reader = **new** Scanner(System.*in*);

## reader对象

### next×××()

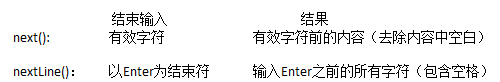
读取基本类型数据：eg: **int** obj = reader.nextInt();

nextBoolean()、nextByte()、nextShort()、nextInt()、nextLong()、nextFloat()、nextDouble()、next()

读取String（实体类、文件类型、任意类型）数据：

public String next()，public String nextLine()

next() 与 nextLine() 区别



next():

1、一定要读取到有效字符后才可以结束输入。

2、对输入有效字符之前遇到的空白，next() 方法会自动将其去掉。

3、只有输入有效字符后才将其后面输入的空白作为分隔符或者结束符。next() 不能得到带有空格的字符串。

nextLine()：

1、以Enter为结束符,也就是说 nextLine()方法返回的是输入回车之前的所有字符。

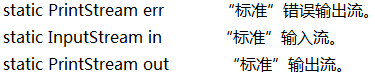
2、可以获得空白。

### hasNext×××()

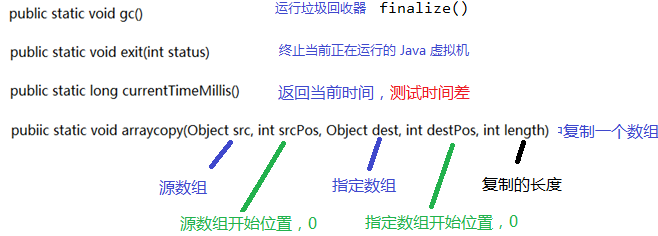
判断使用的类型—boolean

# System类

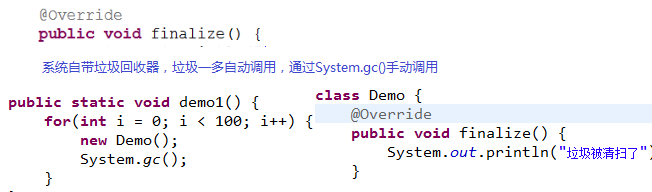
## 字段

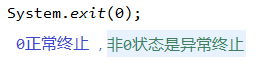


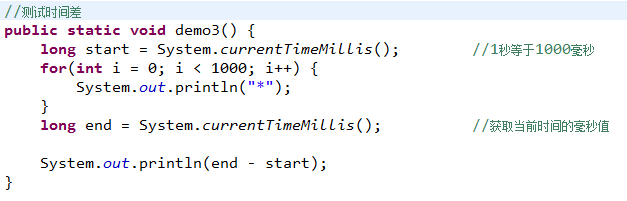
## 常用方法

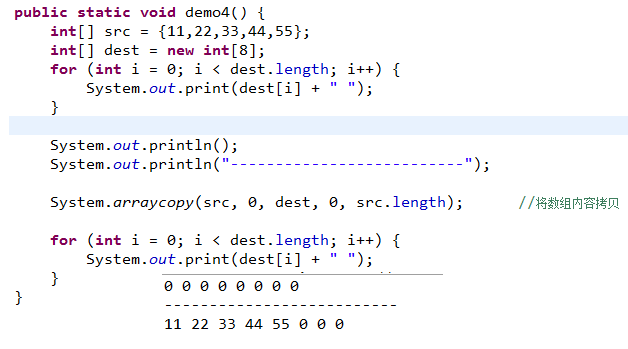


eg:









# Runtime

Runtime 类主要描述的是应用程序运行的环境。

