# Day：Java基础

## final关键字的理解

1、final的意思是【无法改变的】、【最终的】，可以修饰非抽象类、非抽象类的成员变量和成员方法

2、final类不能被继承，没有子类，其中的方法默认是final的

3、final方法不能被重写，但可以被继承（final不能用于修饰构造方法）

4、final成员变量表示常量，只能被赋值一次，赋值后不再改变。

5、使用final声明基础数据类型时，数值恒定不变；使用final声明对象引用时，引用的对象恒定不变，但对象的数据可变；使用final声明数组类型时，引用的数组恒定不变，但数组内的数据可变。

优点：

1、编译器遇到final方法时会转入内嵌机制，提高效率。

2、可以安全的在多线程环境下共享，不需要额外的同步开销。

## 什么是匿名内部类

1、匿名内部类即没有名字的内部类，只可被使用一次

2、使用匿名内部类前提条件：必须继承自一个父类或实现一个接口

3、语法格式为 new 父类构造器(参数列表)|实现接口(){}

4、直接将抽象类的方法或者接口方法在大括号中实现，可以省略一个类的书写

5、匿名内部类中不能定义构造方法，但可以使用初始化语块代替构造方法

## 为什么匿名内部类使用局部引用要用final

1、匿名内部类属于一种局部内部类。

2、编译后局部内部类中会有一个成员变量，是对外部局部变量的引用的拷贝，在局部内部类使用外部局部变量值时都是通过这个引用进行的。

3、为避免这个成员变量的值（引用的对象）被外部类的方法修改，导致内部类在使用时得到的值不一样，需要使用final让该成员变量不可变。

4、局部变量位于方法内部，在虚拟机的栈上，意味着这个变量无法进行共享，匿名内部类无法直接访问，只能通过值传递的方式，传递给匿名内部类。

5、而类的成员变量在虚拟机的堆上，内部类可以直接获取这个变量，故类的成员变量不需要声明为final内部类就可访问。

## 值传递与引用传递

1、Java总是采用按值调用，方法得到的是参数值的一个拷贝。

2、一个方法不能改变其参数的值，如果是基本类型参数不能改变参数的值，如果是对象参数不能改变参数的引用（但可以改变）

## 引用分类

1、强引用：是使用最普遍的引用。如果一个对象具有强引用，那垃圾回收器绝不会回收它。

2、软引用：如果内存空间足够，垃圾回收器就不会回收软引用对象；如果内存空间不足，就会回收软引用对象。

3、弱引用：弱引用对象比软引用对象生命周期更短，垃圾回收器扫描时一旦发现弱引用对象，就会回收它。垃圾回收器优先级不高，所以不一定很快发现弱引用对象。

4、虚引用：虚引用对象就和没有被引用一样，任何时候都可能被垃圾回收器回收。一般用于跟踪对象被垃圾回收器回收的过程，如开源库LeakCanary。