# 概述

所为内部类，就是将一个类定义在另一个类的内部本文档记录不同类型的内部类的知识点。

内部类如下特性（摘自《Think in java》）：

* 内部类可以用多个实例，每个实例都有自己的状态信息，并且与其他外围对象的信息相互独立。
* 在单个外围类中，可以让多个内部类以不同的方式实现同一个接口，或者继承同一个类。
* 创建内部类对象的时刻并不依赖于外围类对象的创建。
* 内部类并没有令人迷惑的“is-a”关系，他就是一个独立的实体。
* 内部类提供了更好的封装，除了该外围类，其他类都不能访问。

# 作用

为什么要使用内部类？在《Think in java》中有这样一句话：使用内部类最吸引人的原因是：每个内部类都能独立地继承一个（接口的）实现，所以无论外围类是否已经继承了某个（接口的）实现，对于内部类都没有影响。

# 分类

## 成员内部类

成员内部类是外围类的一个成员，因此可以访问外部类的所有成员，访问方式采用如下方式：

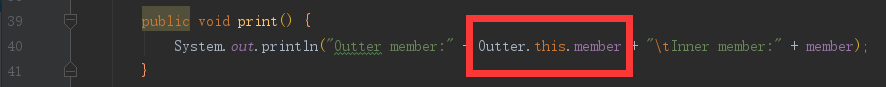


Figure 内部类访问外部类成员

通过外部类对象创建内部类对象的方式：

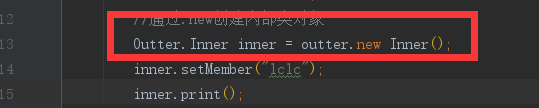


Figure 创建内部类对象方式

或者在外部类中定义一个方法，直接new一个内部类对象，如下：

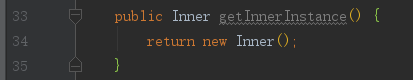


Figure 通过外部类方法创建内部类对象

注意事项：

* 成员内部类不能有static的方法和成员，因为static变量是类变量，隶属于类，而内部类只有依附于外部类存在，只有创建了外部类对象后才能访问到，不能单独通过内部类访问static变量和方法；

## 局部内部类

局部内部类是将一个类定义在一个另一个类的方法体或者作用域，局部内部类和成员内部类的区别是局部内部类的作用域仅仅限于该方法或者作用域。

注意事项

* 局部内部类在使用一个方法的形参时，该形参必须是final修饰的；

## 匿名内部类

匿名内部类，顾名思义就是没有类名的类。

匿名内部类是特殊的局部内部类。

创建格式如下：

new 父类构造器（参数列表）|实现接口（）

{

//匿名内部类的类体部分

}

由于匿名内部类没有构造方法，因此不能通过构造方法进行初始化，如果想要匿名内部类初始化，可以通过代码块进行初始化，如下所示：

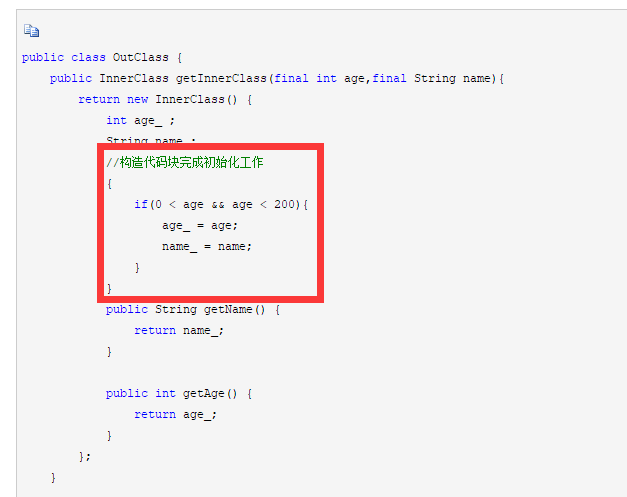


Figure 匿名内部类初始化

注意事项

* 由于没有名字，匿名内部类没有构造方法；
* 匿名内部类没有访问修饰符；
* 匿名内部类必须是继承某个类或者实现某个接口，并且只能继承或实现一个类或接口；
* 匿名内部类不能存在static变量或方法，因为没有类名，无法访问到成员；
* 局部内部类的限制条件都适用于匿名内部类；

## 静态内部类

# 参考

<https://www.cnblogs.com/xiao-chuan/p/6014752.html>

<https://www.cnblogs.com/chenssy/p/3388487.html>