## 模板方法模式

1. 定义与使用场景

定义：定义了操作中算法的框架，而将一些步骤延迟到子类中，使得子类可以不改变一个算法的结构即可重定义该算法的某些特定步骤

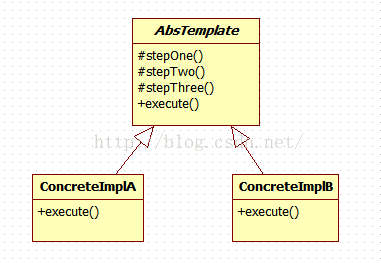
使用场景：

1）、多个子类有公有的方法，并且逻辑基本相同时

2）、重要、复杂的算法，可以把模板核心算法设计为模板方法，周边的相关细节功能则由各个子类实现

3）、重构时，模板方法模式是一个经常使用的模式，把相同的代码抽取到父类中，然后通过抽象方法约束其行为

1. UML类图



AbsTemplete：抽象类，定义了一套算法框架

ConcreteImplA：具体实现类A

ConcreteImplB：具体实现类B

1. 简单实现

模板方法实际上是封装一个固定的流程，就像是一套执行模板一样，第一步该做什么，第二步该做什么都已经在抽象类中定义好。而子类可以有不同的算法实现，在框架不被修改的情况下实现某些步骤的算法替换。

1. 源码实现

AsynTask、Activity的生命周期

1. 实战与总结

流程封装