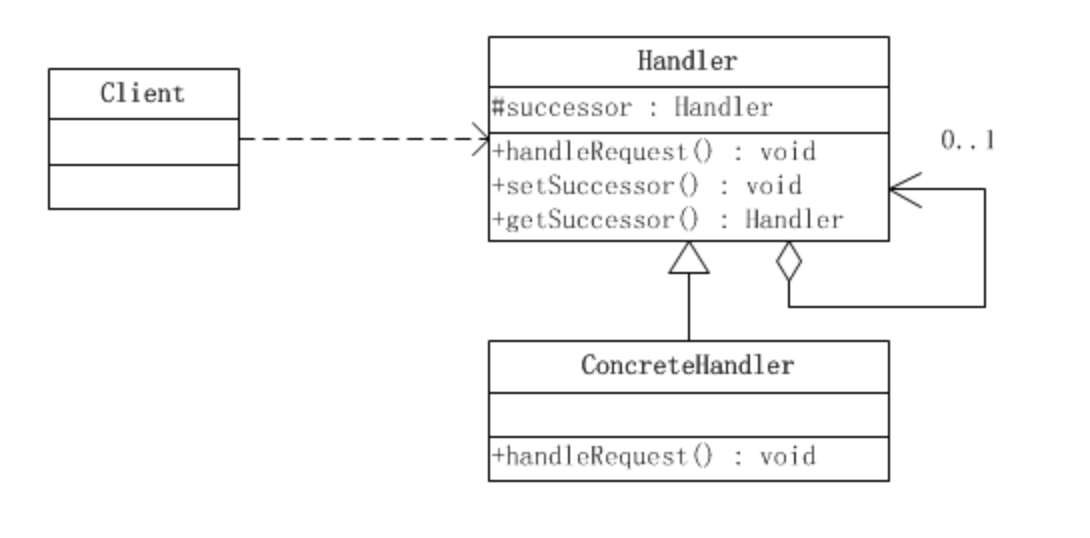
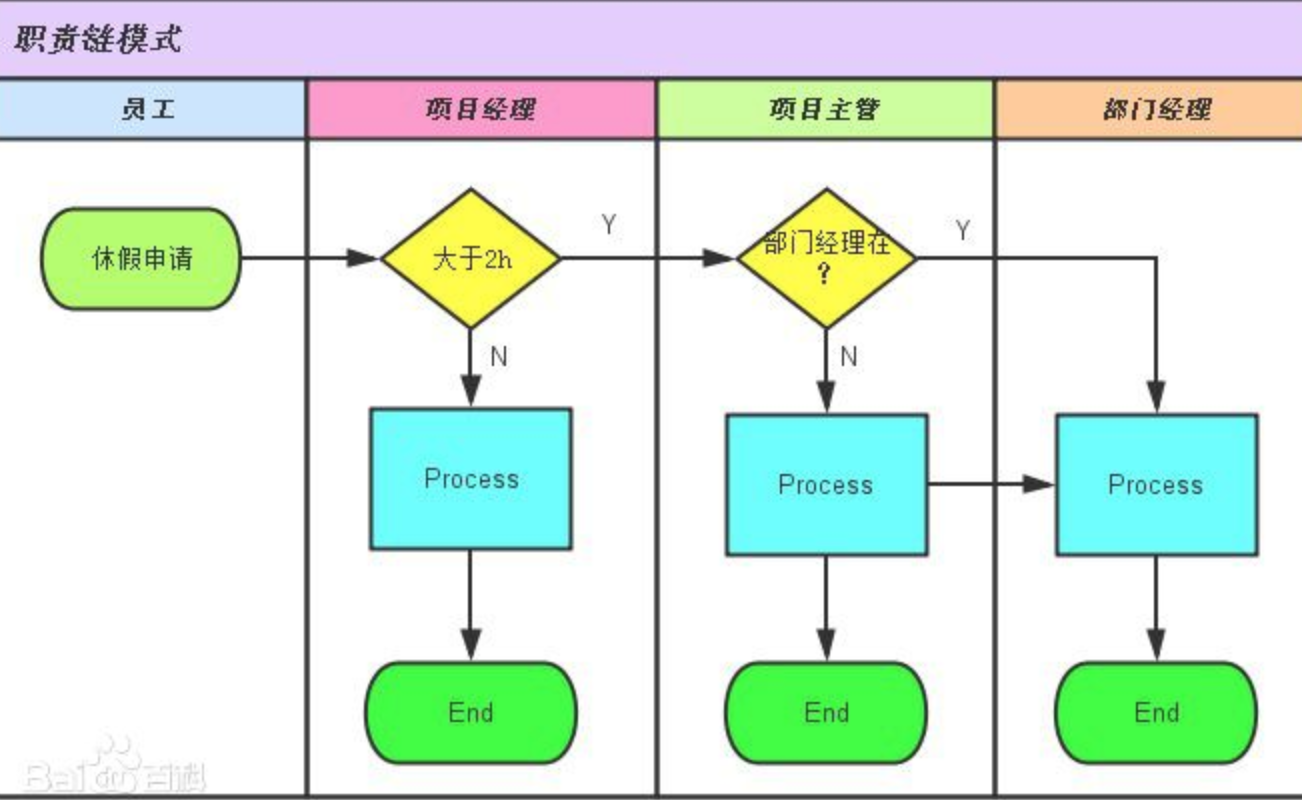
1. 责任链模式：将能够处理同一类请求的对象连成一条链, 所提交的请求沿着链传递, 链上的对象逐个判断是否有能力处理改请求, 如果能则处理, 如果不能则传递给链上的下一个对象
2. 类图



* 1. 抽象处理者(Handler)角色：定义出一个处理请求的接口。如果需要，接口可以定义 出一个方法以设定和返回对下家的引用。这个角色通常由一个Java抽象类或者Java接口实现。上图中Handler类的聚合关系给出了具体子类对下家的引用，抽象方法handleRequest()规范了子类处理请求的操作。
  2. 具体处理者(ConcreteHandler)角色: 具体处理者接到请求后，可以选择将请求处理掉，或者将请求传给下家

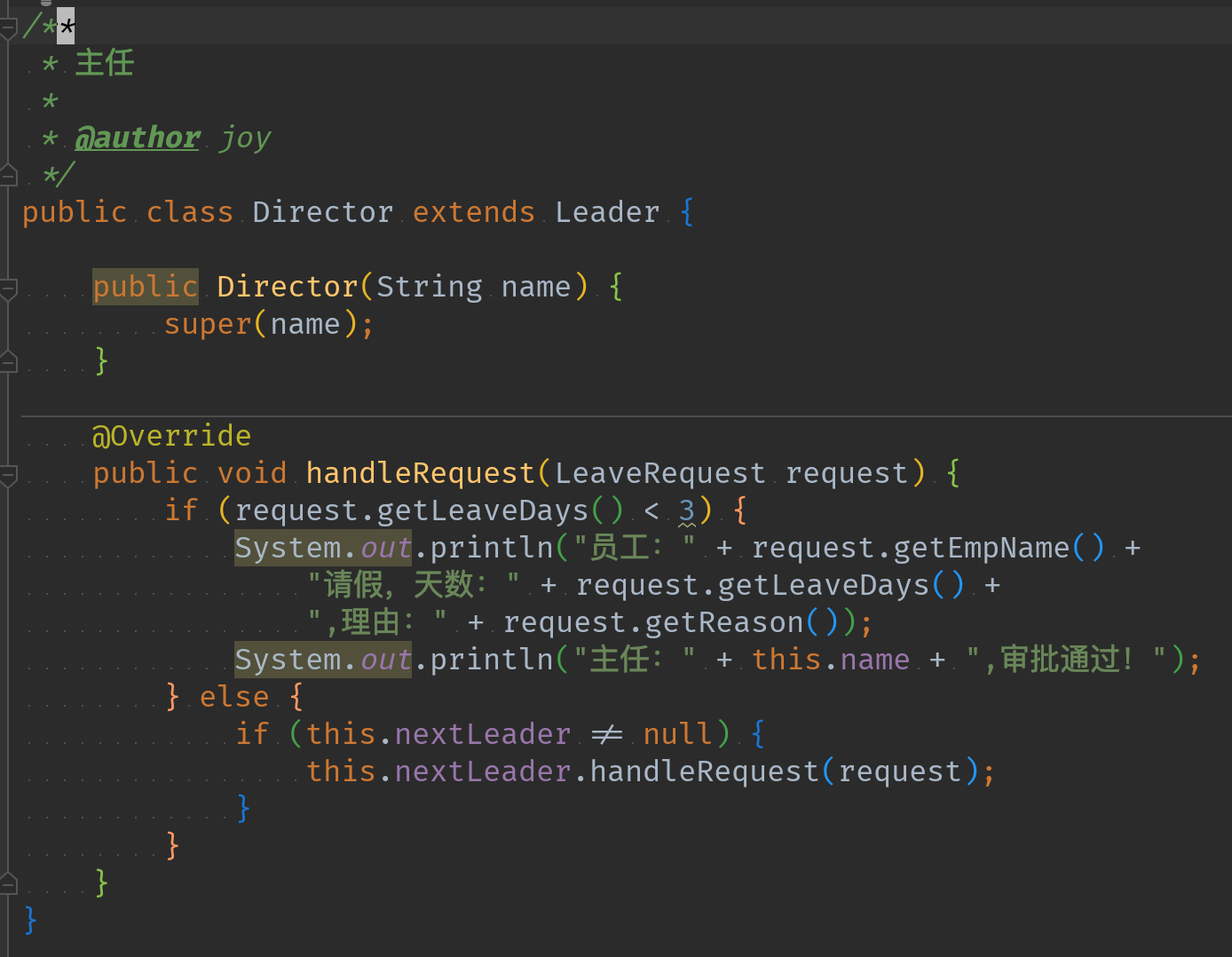
1. 责任链模式应用场合(场景)
   1. 请假审批流程
   2. 奖学金审批
   3. 接力赛跑
2. 简单例子—请假

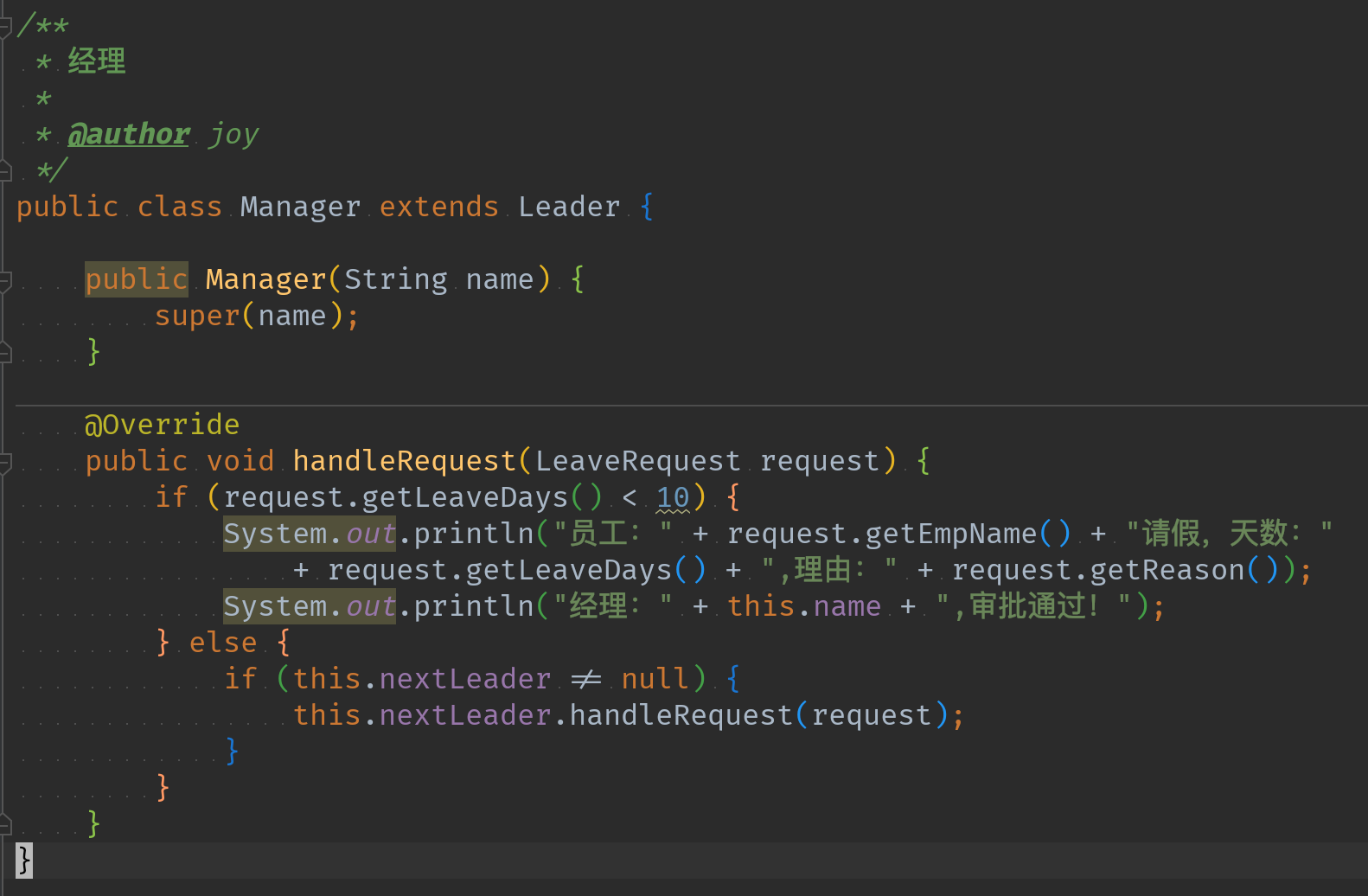


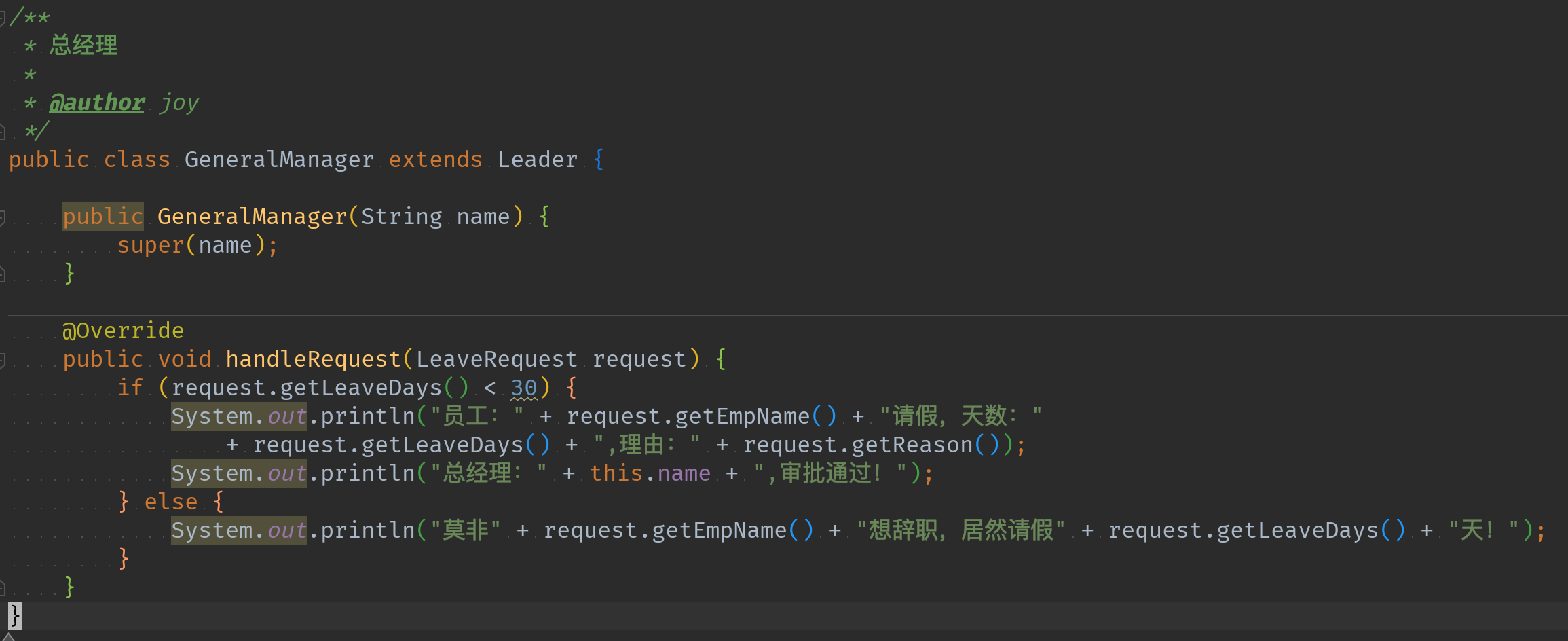
* 1. 抽象处理者



* 1. 具体处理者







* 1. Client



1. 开发中常见场景
   1. Java异常处理机制, 一个try对应多个catch
   2. Servlet开发中过滤器的链式调用