一.责任链模式的现实场景

张三过生日邀请了很多朋友到KTV一起庆祝，为了增加欢乐的气氛，张三建议大家一起玩击鼓传花的游戏。由张三击鼓，大家依次往下传花，鼓声停下来时，花在谁手上就要喝酒并且还要唱一首歌。

大家都很踊跃地参加这个游戏，聚会的气氛很快热闹起来，张三也度过了一个很快乐的生日聚会。

在上面的这个场景中，击鼓传花的这个游戏过程类似于设计模式中的责任链模式，就是要事件不断的进行传递，直到事件被执行为止。

二.责任链模式（Chain of Respinsibility Pattern）的定义

使多个对象都有处理请求的机会，从而避免了请求的发送者和接收者之间的耦合关系。将这些对象串成一条链，并沿着这条链依次传递请求，直到有对象处理它为止。

三..责任链模式的两个角色

1.抽象处理者角色

该角色对请求进行抽象，并定义一个方法来设定和返回对下一个处理者的引用。

2.具体处理者角色

该角色接到请求后，可以选择将请求处理掉，或者将请求传递给下一个处理者。

由于具体处理者持有对下一个处理者的引用，因此，如果需要，具体处理者可以访问下一个处理者，把问题給下一个处理者，并以此类推。

四.责任链模式的优缺点

责任链模式中的优点：

1. 责任链模式将请求和处理分开，请求者不知道是谁处理，处理者可以不用知道请求者全貌。

2. 提高系统的灵活性

命令模式的缺点：

1.降低程序的性能。每个请求都是从链头遍历到链尾，当遇到链比较长的时候，性能就会大幅下降。

2.不易于调试，由于该模式采用了类似递归的方式，调试的时候逻辑比较复杂。

责任链中的节点数量需要控制，避免出现超长链的情况，这就需要设置一个最大的节点数量，一旦超过则不允许增加节点，避免责任链无意识地破化系统性能。

五.责任链模式的使用场景

责任链模式是一种常见的模式，其典型的应用场景如下：

1.一个请求需要一系列的处理工作。

2.业务流的处理，例如文件审批。

3.对系统进行扩展。