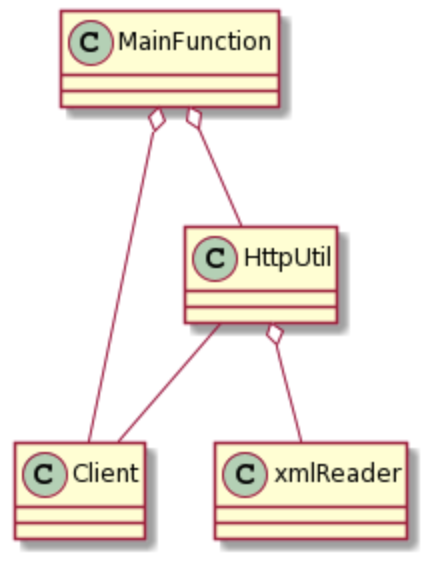
## 设计思路



主函数内启动了10个client线程，client调用相同的HttpUtil对象进行post请求，请求实体为xmlReader返回的xml entity

Post请求核心封装在HttpUtil类中，设计思路如下：

1. 构造器传入client自定义最大链接数量及超时等待时间参数
2. 使用request config对象封装这些http parameters
3. 为了使多线程访问资源，使用连接池PoolingHttpClientConnectionManager进行管理
4. 考虑到session有一定长度，引入keepAliveStrategy使连接存活一定时间，默认存活时间为300秒
5. 多线程并发访问有可能会出现资源请求失败抛出异常，引入StandardHttpRequestRetryHandler处理异常。
6. 引入监视线程IdleConnectionMonitorThread定期监控并清除idle / expired thread，默认时间为一分钟
7. 关闭连接池可调用shutDownAllConnections方法，client thread由于被调用interrupt方法会捕捉到InterruptedException，打印连接结束信息
8. 传入url和xml来调用post方法，方法会添加相应的header和config并将xml文档的内容写入到post entity中。
9. post请求返回的response及session信息会被logger记录到日志中
10. Client 及 HttpUtil类中使用log4j输出相关日志

Client类为定义具体请求的入口，本次demo使用间隔10s进行一次post请求的设定

xmlReader使为准备xml文档相关的参数，此次仅使用默认上传同一份xml文档的设定。

## 编译设定

1.相关依赖放置在src同级目录中

2.src下log4j.properties中可改变具体日志输出位置，当前为/Users/luozhou/Desktop/log.log

3.邮件中python简易服务器代码无法处理post请求，我新建了一个python小程序提供get/post简易请求能返回200 OK的服务器，具体代码请查阅python程序，默认为localhost:8080

4.编译通过及session记录日志附在project中