**说明文档**

**一、项目结构**



其中各包和其作用如下：

1. **Constant包**

常量包，用来存放通用常量值

1. **http包**

包含请求处理线程类（RequestHandlerThread）、服务器类（Server）和容器类（Container），其中服务器类利用ServerSocket接收浏览器发送的请求创建一个Socket，并且新建一个请求处理线程来处理这次请求。容器类还在完善中，计划将Session及服务器相关数据存放在此

1. **Middleware包**

中间件包，用来对响应response添加必要的header信息，其中ContentLenMiddleware类向response添加Content-Length: xxx头，DateMiddleware类向response添加Date: xxx头，ServerMiddleware类向response添加Server: xxx头

这里有点问题，我不太确定一个http response到底需要包含哪些必要的头信息

1. **Request 包**

请求包，其中包含关于Http request相关的类，Request接口用来封装从输入流（InputStream）中得到的Http request报文，并解析它，分离header和body，提取传递的参数等信息，便于上层应用方便的使用request对象。RequestWrap是Request接口的实现类。Cookie类封装了Http request报文中关于Cookie的信息，Session则封装了会话信息（还未实现）

1. **Response包**

响应包，其中包含关于Http response相关的类，Response接口用于封装即将写入输出流（OutputStream）传输到客户端的Http response报文，分为header和body两部分。ResponseWrap是Response的基本实现类。Rsxxx类是特殊响应类，针对某一个具体的响应要求，如RsFile用来向客户端返回一个文件响应，RsNotFound向客户端返回一个404 Not Found响应，RsText向客户端返回一个纯文本的响应，今后还会添加更多类型的响应，如RsJson用来返回Json格式的响应，RsImg用来返回图片响应等

1. **Route包**

路由包，用来根据Http request请求中的Url地址和用户自定义的映射Map<String, Action>来决定客户端本次访问由哪一个Aciton实现来进行处理，Route接口定义了route方法，RouteWrap为Route简单实现类，RtRegex则是Route的正则表达式实现，用户可以定义一个< "^/user", someAction>，则浏览器访问xxx/user、xxx/users、xxx/user/123都将被这个someAction处理，并且可以收集Url上传递的数据，如定义一个< "^/user/(?P <userid>[0-9]+)"$, someAction>，当浏览器访问xxx/user/123时，这个123可以通过request.getParameter("userid")得到（类似于Python的web框架Django）

1. **Util包**

工具包，FileHelper中的getContent方法可以获得一个文件的内容，如果文件不存在则返回null

1. **View包**

视图包，在这里准备实现包含特殊代码（逻辑控制语句、循环语句等）的html视图的渲染，将其渲染成一个标准的html文件，便于封装进Http response的body中返回客户端。目前正在完善中（准备实现成类似于Python的Web框架Django中模板引擎的写法）

1. **Action接口**

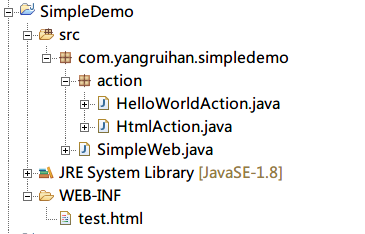
Action接口定义了一个方法



接受一个request对象，返回一个response对象，由上层应用用户自己实现，中间包含处理逻辑，类似于Java EE中的Servlet

**二、简单演示**

简单演示项目结构：



其中SimpleWeb类为整个程序的入口，action包里存放用户自定义的Action类，用来处理用户想要的逻辑

1. **SimpleWeb.java代码：**



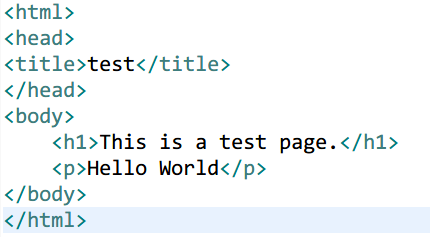
1. **HelloWorldAction.java代码：**



1. **HtmlAction.java代码**

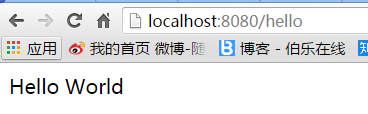


1. **Test.html代码**

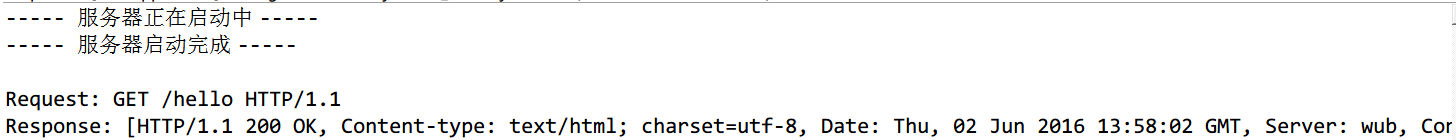


**运行结果：**

浏览器显示：



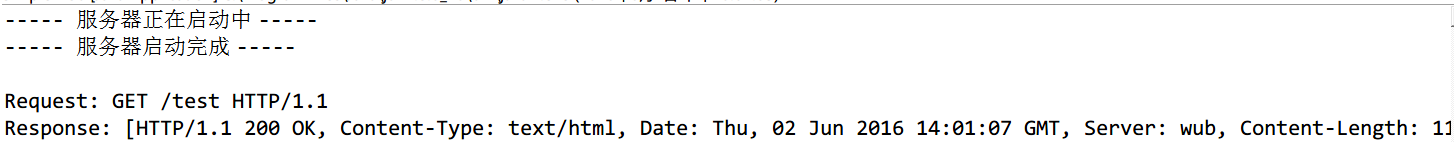
控制台显示：



浏览器显示：



控制台显示：



整个项目的源代码放在github上：https://github.com/yangruihan/wub