武汉纺织大学

Java应用开发课程设计

**基于HttpClient的生活百宝箱**

**学 院： 数学与计算机学院**

**班 级： 物联网11802**

**姓 名： 喻言**

**学 号： 1804240707**

**指导老师： 聂刚**

**成 绩：**

**完成日期： 2020年6月20日**

目 录

[1 需求分析 1](#_Toc2965)

[1.1用户登陆 1](#_Toc6465)

[1.2主菜单 1](#_Toc27274)

[1.3各项功能 1](#_Toc17655)

[1.3.1 天气预报查询 1](#_Toc21322)

[1.3.2 IP地址来源查询 1](#_Toc30258)

[1.3.3 中英文双向翻译 2](#_Toc28610)

[1.3.4 身份证归属查询 2](#_Toc14439)

[1.3.5 手机号归属查询 3](#_Toc27115)

[1.3.6 退出 3](#_Toc28851)

[2 系统设计 3](#_Toc15323)

[3 系统实现 9](#_Toc27433)

[4 系统测试 14](#_Toc19341)

[5 系统总结 14](#_Toc30615)

# 1 需求分析

设计一个生活百宝箱程序，实现查询“天气预报”、“IP地址”、“中英翻译”、“身份证归属地”、“手机号归属地”等功能，具体要求如下：

## 1.1用户登陆

当程序运行时，显示

生活百宝箱

1. 天气预报查询
2. IP地址来源查询
3. 中英文双向翻译
4. 身份证归属查询
5. 手机号码归属地查询
6. 退出

用户可自主选择需要查询的选项进行查询。

## 1.2主菜单

当用户名及密码正确时，进入系统主界面，如下：

生活百宝箱

1. 天气预报查询
2. IP地址来源查询
3. 中英文双向翻译
4. 身份证归属查询
5. 手机号码归属地查询
6. 退出

如果输入1-6以外的选择，则显示提示信息“输入无效，只能输入1-6”后，返回主界面。

## 1.3各项功能

### 1.3.1 天气预报查询

当选择1时，弹出“请输入查询地址”，然后输入回车后会显示当前天气以及明后两天的天气。

再次按回车后，返回主菜单。

### 1.3.2 IP地址查询

当选择2时，显示“请输入查询地址:”，输入想要查询的地址后，按回车键，即可出现查询结果。

再次按回车后，返回主菜单。

### 1.3.3 中英文双向翻译

当选择3时，显示“请输入翻译内容：”，输入内容后，按回车键，可以出现该内容的英文结果。

再次按回车后，返回主菜单。

### 1.3.4 身份证归属地查询

当选择4时，显示“请输入身份证号码：”，输入内容后，按回车键，可以出现该身份证的对应地址。

再次按回车后，返回主菜单。

1.3.5 手机号码归属地查询

当选择5时，显示“请输入手机号码：”，输入号码后，按回车键，可以出现该号码的归属地名称。

再次按回车后，返回主菜单。

1.3.6 退出

选择6时，显示“您确认退出系统吗（y/n）”,输入y，显示“欢迎下次继续使用”，退出程序，输入“n”，返回主菜单。

# 2 系统设计

# 2.1 Java HTTP网络编程

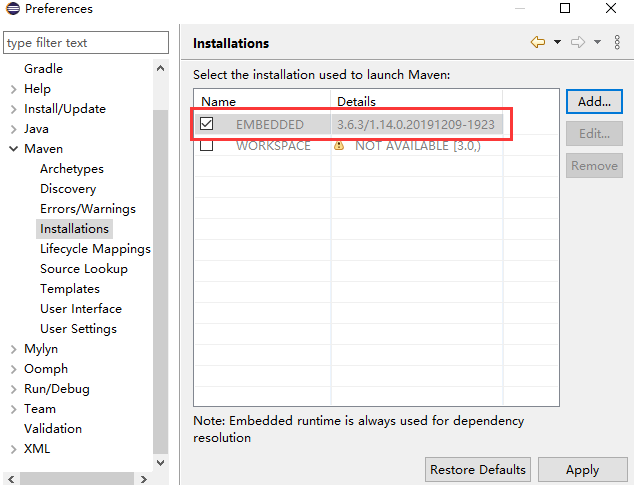
HTTP 协议是现在 Internet 上使用得最多、最重要的协议，越来越多的 Java 应用程序需要直接通过 HTTP 协议来访问网络资源。虽然在 JDK 的 java net包中已经提供了访问 HTTP 协议的基本功能，但是对于大部分应用程序来说，JDK 库本身提供的功能还不够丰富和灵活。HttpClient 是 Apache Jakarta Common 下的子项目，用来提供高效的、最新的、功能丰富的支持 HTTP 协议的客户端编程工具包。官方下载地址为<http://hc.apache.org/downloads.cgi> ，目前最新版本是HttpClient5.0。

httpclient5-5.0.jar的运行依赖于**httpclient5-5.0.jar、httpcore5-h2-5.0.jar、slf4j-api-1.7.25.jar、commons-codec-1.13.jar**等包的支持。

#### 2.2 Maven介绍

Maven是基于项目对象模型(POM project object model)，可以通过一小段描述信息（配置）来管理项目的构建，报告和文档的软件项目管理工具。Maven 翻译为"专家"、"内行"，是 Apache 下的一个纯 Java 开发的开源项目。核心功能是合理叙述项目间的依赖关系，通俗点讲，*就是通过pom.xml文件的配置获取jar包，而不用手动下载添加jar包*。

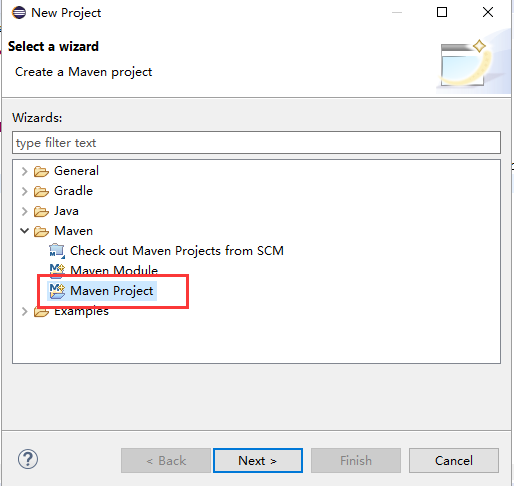
在最新的Eclipse中自带了maven，打开Windows->Preferences,可以看到下面的画面：



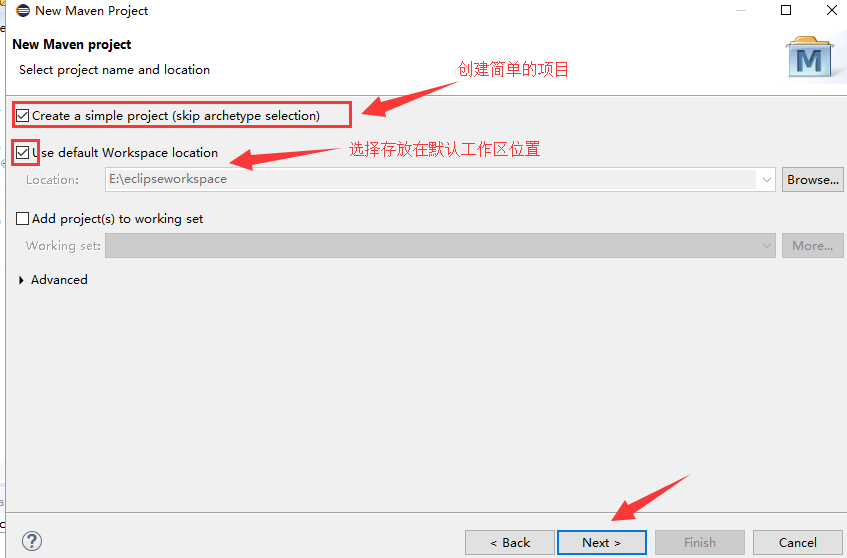
也可以点击Add按钮使用外部自己另外安装的maven版本。

#### 2.3 创建Maven项目

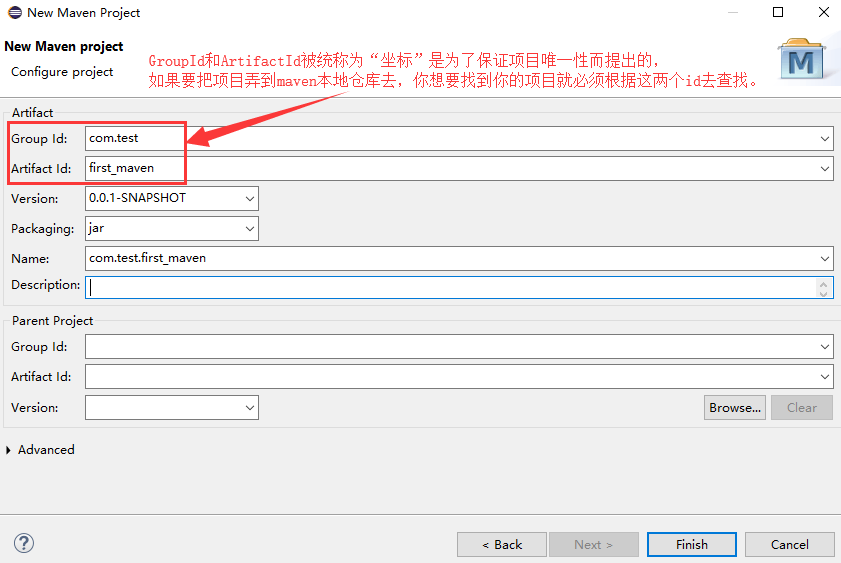
1、新建项目，选择Maven Project



2 选择创建简单的Maven项目，选择默认的工程位置



3、填写相关的groupId， artifactId， version等信息



maven的世界中拥有数量非常巨大的构件，也就是平时用的一些jar,war等文件。maven定义了这样一组规则：

世界上任何一个构件都可以使用*Maven坐标唯一标志*，maven坐标的元素包括GroupId,，ArtifactId，Version，Packaging等。

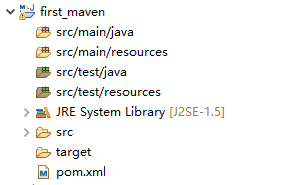
GroupId：一般分为多个段，第一段为域，第二段为公司名称。域又分为org、com、cn等等许多，其中org为非营利组织，com为商业组织。

例如apache公司的tomcat项目，这个项目的GroupId是org.apache，它的域是org（因为tomcat是非营利项目），公司名称是apache，ArtifactId是tomcat。

只要在pom.xml文件中配置好dependancy的groupId，artifact，verison等坐标元素,maven就会从仓库中寻找相应的构件供我们使用。maven内置了一个中央仓库的地址（*https://mvnrepository.com/*）,该中央仓库包含了世界上大部分流行的开源项目构件，maven会在需要的时候去那里下载。

4、点击完成就创建好了一个简单的maven工程。

maven的工程一般都是如下的结构：



pom.xml：用于定义或者添加jar包的依赖

src-main：用于存放java源文件

src-test：用于存放测试用例。

也许工程下还会出现target文件夹，这个是用来生成对应的class文件或发布的jar包。

# 3 系统实现

# 3.1 开发环境搭建

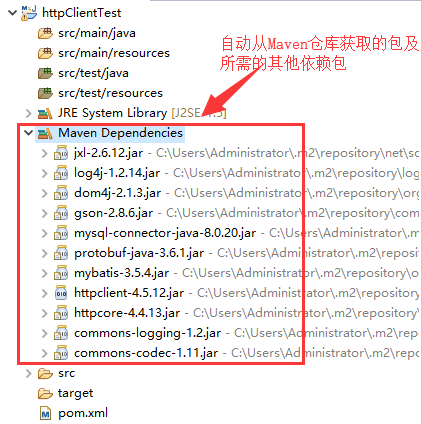
在Eclipse下新建Maven项目，Group Id输入edu.wtu，Artifact Id输入httpClientTest，打开pom.xml文件，加入项目开发所需的包（为演示起见，本例把前期所介绍的所有包jxl、dom4j、gson、mysql jdbc、myBatis、httpclient都通过Maven进行增加）。

在Maven中心仓库**https://mvnrepository.com/**查找到每个包的Maven依赖配置信息（*请注意查询结果的正确选择*），复制到pom.xml文件的*<dependencies>*标记下。





选择Maven-Update Project后，项目结构如下：



# 3.2 HttpClient 4.x基本功能的使用

HttpGet请求响应的一般步骤：

1. 创建HttpClient对象,可以使用HttpClients.createDefault()；
2. 如果是无参数的GET请求，则直接使用构造方法HttpGet(String url)创建HttpGet对象即可；

如果是带参数GET请求，可以采取url地址拼接的形式，如http://a.b.c?a=1&b=2，然后使用构造方法HttpGet(String url)创建HttpGet对象；也可以先使用URIBuilder(String url)创建URIBuilder对象，再调用addParameter(String param, String value)，或setParameter(String param, String value)来设置请求参数，并调用build()方法构建一个URI对象。使用构造方法HttpGet(URI uri)来创建HttpGet对象。

1. 创建HttpResponse，调用HttpClient对象的execute(HttpUriRequest request)发送请求，该方法返回一个HttpResponse。调用HttpResponse的getAllHeaders()、getHeaders(String name)等方法可获取服务器的响应头；调用HttpResponse的getEntity()方法可获取HttpEntity对象，该对象包装了服务器的响应内容。程序可通过该对象获取服务器的响应内容。通过调用getStatusLine().getStatusCode()可以获取响应状态码。
2. 释放连接。

HttpPost请求响应的一般步骤：

Post请求包含两部分：**请求头**（header）和**请求体**（body）

Post常见的请求体（body）有三种传输内容类型Content-type：**application/x-www-form-urlencoded**、**application/json**、**multipart/form-data**，当然还有其他的几种，不过不常用，常用的就是这三种。

application/x-www-form-urlencoded：是Post请求最早支持的一种数据传输方式，这种也是key和value形式，将我们的参数类似于GET方式那样拼接成一个字符串，例如：key1=value1&key2=value2，这种形式，然后将这个参数字符串进行urlencode编码，放到Body里进行发送请求数据。

application/json：也就是告诉我们的服务器我们的消息体内容类型是序列化的JSON字符串，例如：{ "name": "value1", "pwd": "value2" }。获取到这个body直接解析Json格式字符串即可拿到参数数据。

multipart/form-data：这种格式主要用来进行文件上传

具体步骤如下：

1. 创建HttpClient对象,可以使用HttpClients.createDefault()；
2. 直接使用构造方法HttpPost(String url)创建HttpPost对象；

如果是带参数POST请求，构建UrlEncodedFormEntity 对象并设置请求参数，创建时必须是NameValuePair类型的对象作为他的构造参数。然后调用setEntity()将请求内容设置到HttpPost对象。

List<BasicNameValuePair> parameters = new ArrayList<BasicNameValuePair>();

parameters.add(new BasicNameValuePair("key1", "value1"));

// 构造一个form表单式的实体

UrlEncodedFormEntity formEntity = new UrlEncodedFormEntity(parameters,"utf-8");

httpPost.setEntity(formEntity); // 将请求实体设置到httpPost对象中

如果是直接发送JSON字符串，构建StringEntity对象，然后调用setEntity（）将JSON字符串放在HttpPost对象请求体内。

String jsonStr="\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

StringEntity se = new StringEntity(jsonStr,"UTF-8");

se.setContentType("application/json"); //设置请求头部Content-Type

se.setContentEncoding("utf-8"); //设置请求头部Content-Encoding

httpPost.setEntity(se);

1. 创建HttpResponse，调用HttpClient对象的execute(HttpUriRequest request)发送请求，该方法返回一个HttpResponse。调用HttpResponse的getAllHeaders()、getHeaders(String name)等方法可获取服务器的响应头；调用HttpResponse的getEntity()方法可获取HttpEntity对象，该对象包装了服务器的响应内容。程序可通过该对象获取服务器的响应内容。通过调用getStatusLine().getStatusCode()可以获取响应状态码。
2. 释放连接。

使用HttpClient4遇到的一些问题及解决方法

1中文参数乱码解决：

Get方式提交的乱码处理：对中文参数使用URLEncoder.encode(value,"UTF-8");来编码；

Post方式提交的乱码处理：

UrlEncodedFormEntity entity = new UrlEncodedFormEntity(formparams, "UTF-8");

或

StringEntity se = new StringEntity(jsonStr,"UTF-8");

2 相应内容中文乱码处理：

EntityUtils.toString(entity,"utf-8");

3 超时设置

httpClient.getParams().setIntParameter(CoreConnectionPNames.CONNECTION\_TIMEOUT,5000);//连接超时

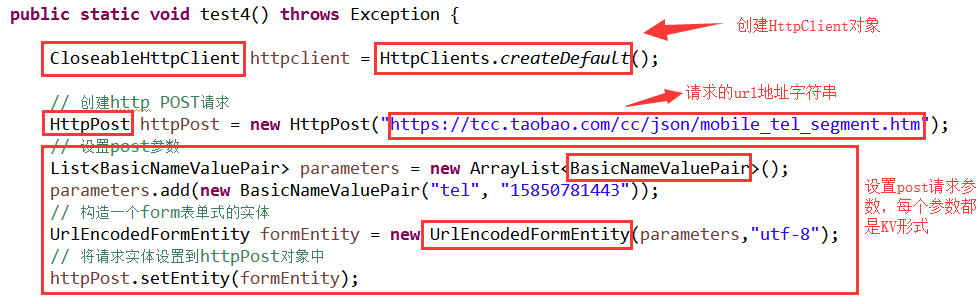
httpclient.getParams().setIntParameter(CoreConnectionPNames.SO\_TIMEOUT,5000);/读取超时

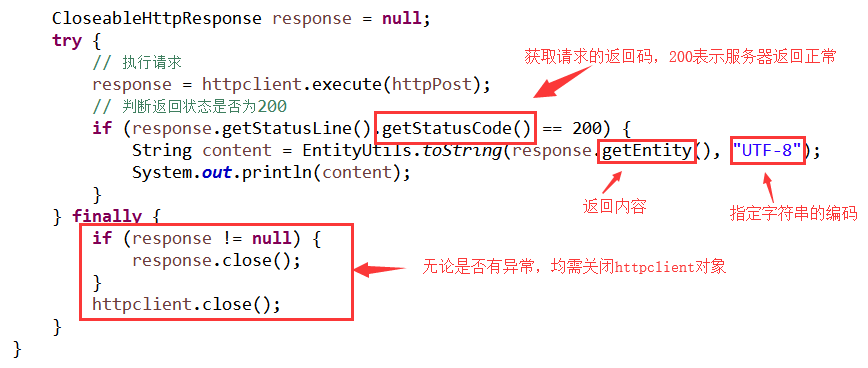
# 4 系统测试

向淘宝提供的HTTP API接口发送get请求，查询指定手机号码的所属运营商和所在地

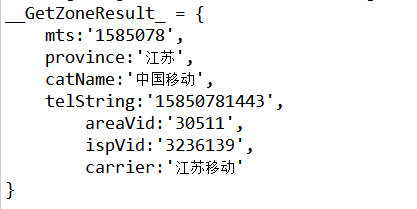


向淘宝提供的HTTP API接口发送post请求，查询指定手机号码的所属运营商和所在地。

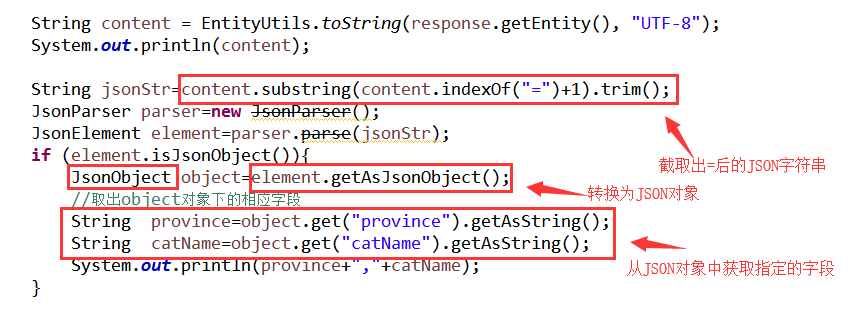




API接口返回以下JSON格式的结果



获取服务器返回的结果后，利用前面所讲的JSON解析方法，从中解析出province和catName字段内容。



# 5 系统总结

在观看老师讲解视频以及教学文件后，我尝试自己运用所学知识来完成这一任务。我大概将任务分为menu制作和各功能制作。在完成过程中了解了javahttp的各方面知识，以及各种api的查询与运用。但是由于自身的实践能力不够，无法将学到的知识点完整的表达出来。所以只能努力的将各块程序完成，无法融会贯通成为一个整体。到最后也是书到用时方恨少，察觉了自己实际操作能力的不足。以后一定多打些代码，增强自己的实践能力。