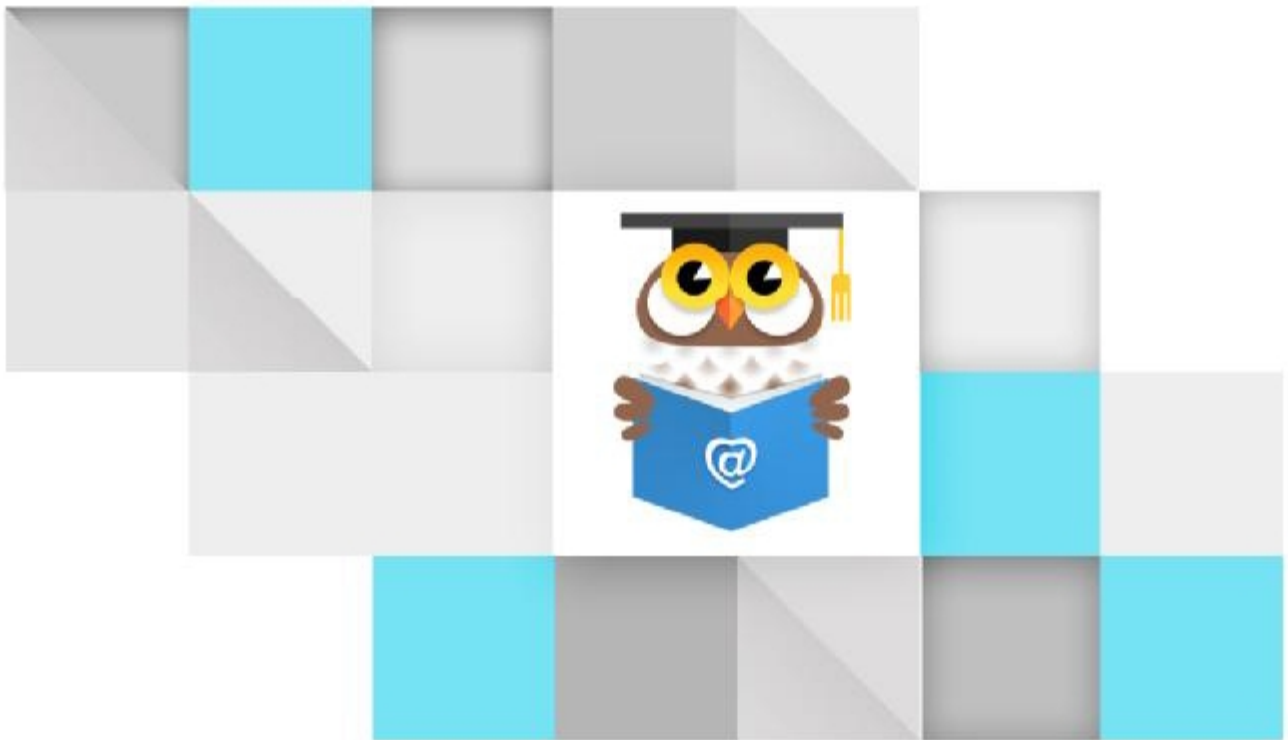


# 去哪玩旅游网（动态 WEB）

## 教学指导手册

### PRJ-WTP-JEE-002 监控请求与响应



# 目 录

一、场景说明 .....	1
1、实现效果 .....	1
2、业务描述 .....	1
3、实现思路 .....	1
4、核心组件 .....	2
二、实训技能 .....	2
1、重点演练 .....	2
2、相关技能 .....	2
3、相关知识点 .....	2
4、前置条件 .....	3
5、搭建环境 .....	3
三、场景任务 .....	4
任务 1、设置 Web 应用首页 .....	4
任务 2、技术体验：查看请求与响应 .....	6
任务 3、业务体验：未登录状态下访问站点 .....	10
任务 4、业务体验：登录状态下访问站点 .....	12
场景总结 .....	15

# 一、场景说明

## 1、实现效果

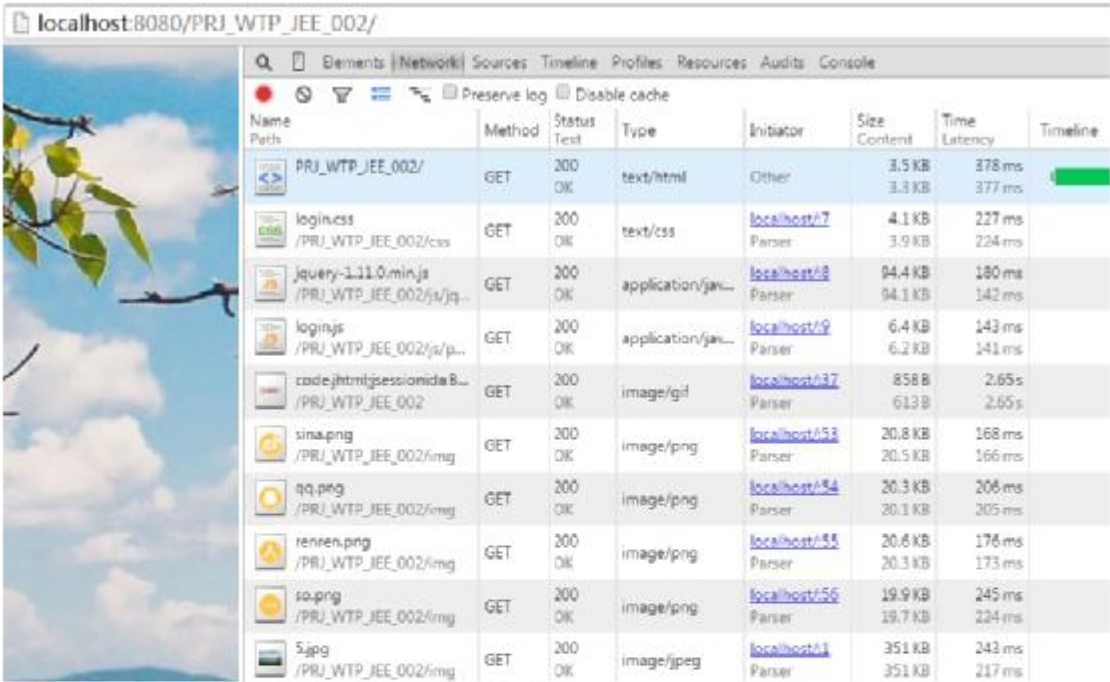


图 1-1-1

## 2、业务描述

本场景用于深入理解HTTP协议中请求和响应的数据结构，并会设置Web应用的首页。

- 2-1. 为Web项目：PRJ-WTP-JEE-002设置首页：login.jsp。
- 2-2. 通过Chrome浏览器的Network工具，监控HTTP协议中的请求数据结构。
- 2-3. 通过Chrome浏览器的Network工具，监控HTTP协议中的响应数据结构。
- 2-4. 按验证测试要求，熟悉《去哪玩旅游网》的业务流程。

## 3、实现思路

无

## 4、核心组件

### 4-1. Network工具:

4-1.1. Network工具集成在Chrome浏览器之中，可通过F12快捷键访问。

4-1.2. Network工具可以方便地查看、监控网络请求中的常用参数与数据。

4-1.3. Network工具可以方便地获取服务端的响应参数与数据。

## 二、实训技能

### 1、重点演练

1-1. 为Web应用设置首页。

1-2. 深入理解HTTP协议中请求的数据结构。

1-3. 深入理解HTTP协议中响应的数据结构。

1-4. 体验完整的《去哪玩旅游网》的业务流程

1-4.1. 登录后，访问《去哪玩旅游网》的业务流程。

1-4.2. 不登录，访问《去哪玩旅游网》的业务流程

### 2、相关技能

- Web 开发环境搭建

### 3、相关知识点

- Web 应用目录结构

- 设置索引页
- HTTP 协议与 HTML
- HTTP 请求数据结构
- HTTP 响应数据结构

## 4、前置条件

4-1. 前置场景：PRJ-WTP-JEE-001 – 配置与优化Server

4-2. 前置技能：

4-2.1. Java开发工具（Eclipse）。

4-2.2. Web容器（Tomcat）。

## 5、搭建环境



## 三、场景任务

### 任务 1、设置 Web 应用首页

#### 1. 任务说明：

##### 1-1. 完成效果：

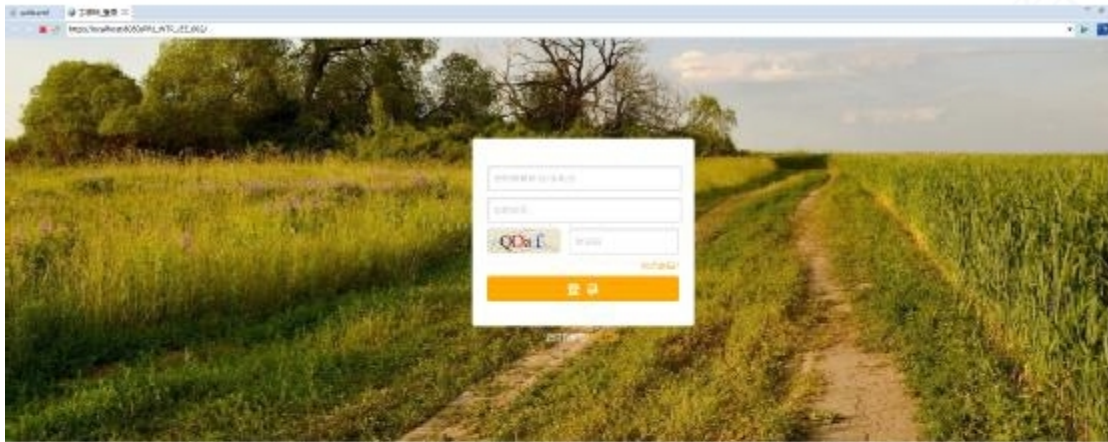


图 6-1

##### 1-2. 任务目标：

1-2.1. 为《去哪玩旅游网》设置站点首页。

##### 1-3. 任务要求：

1-3.1. 将登录页面（login.jsp）设置为《去哪玩旅游网》的首页。

1-3.2. 运行PRJ-WTP-JEE-002项目后见任务完成效果。

#### 2. 实现思路：

2-1. 修改web.xml文件中的<welcome-file>节点。

2-2. 将该节点内容设置为：login.jsp

#### 3. 推荐步骤：

3-1. 修改站点首页：按实现思路完成任务。

+ 提示:

1) web.xml文件的位置:

PRJ\_WTP\_JEE\_002/WebContent/WEB-INF/web.xml

#### 4. 验证与测试:

##### 4-1. 运行项目工程

4-1.1. 右键项目工程，选择【Run As -> Run on Server】

4-1.2. 在弹窗中选择好Tomcat后点击【Finish】

##### 4-2. 显示运行结果

4-2.1. 在Chrome地址栏输入: [http://localhost/PRJ\\_WTP\\_JEE\\_002/](http://localhost/PRJ_WTP_JEE_002/)。

4-2.2. 确认运行结果与任务完成效果截图一致。

##### 4-3. 停止服务器的运行

4-3.1. 在Eclipse的【Servers】标签中右键【Tomcat v7.0 Server at localhost】。

4-3.2. 选择【Stop】终止Tomcat服务器运行。（**IDEA这里不同，后续工程都一样**）

## 任务 2、技术体验：查看请求与响应

### 1. 任务描述：

1-1. 完成效果：无

1-2. 任务目标：

1-2.1. 体验Web站点中请求（request）与响应（response）的数据结构。

1-3. 任务要求：

1-3.1. 利用Network工具监控登录页（login.jsp）的请求和响应数据内容。

1-3.2. 深刻理解HTTP协议的请求和响应的数据结构。

### 2. 实现思路：

2-1. 请按【验证与测试步骤】完成本任务。

### 3. 推荐步骤：

无。

### 4. 验证与测试：

4-1. 运行项目工程

4-1.1. 右键项目工程，选择【Run As -> Run on Server】

4-1.2. 在弹窗中选择好Tomcat后点击【Finish】

4-2. 打开Chrome浏览器

4-3. 打开Network工具

4-3.1. 在Chrome浏览器中按下【F12】，并在弹出菜单窗口中选择【Network】标签

4-4. 测试1：查看浏览器缓存与网页显示效率：

4-4.1. 测试前准备，清除Chrome浏览器浏览历史记录（缓存）：

1) 点击浏览器顶部最右侧“三横线”图标。



- 2) 在下拉菜单中，选择历史记录。
- 3) 点击“清除浏览数据.....”按钮。
- 4) 勾选所有选项，清除时间段下拉框选择：全部。

4-4.2. 开始测试：地址栏输入【http://localhost/PRJ\_WTP\_JEE\_002/】后敲击回车

4-4.3. 通过Network工具，观察此次访问客户端向服务端发送的请求信息：

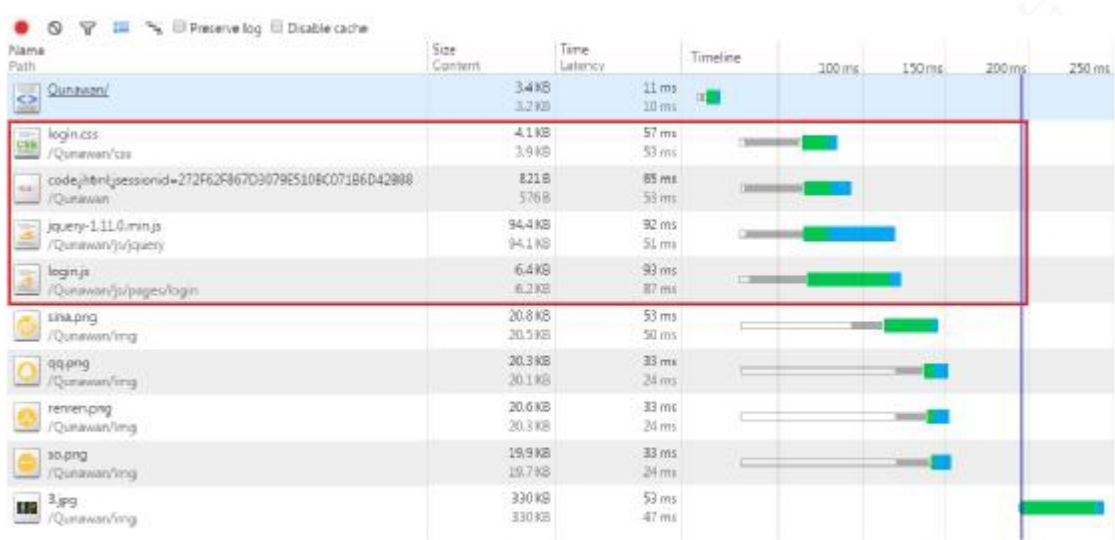


图 3-2-1

4-4.4. 再次在浏览器地址栏中敲击回车，观察第二次客户端发送请求的信息：

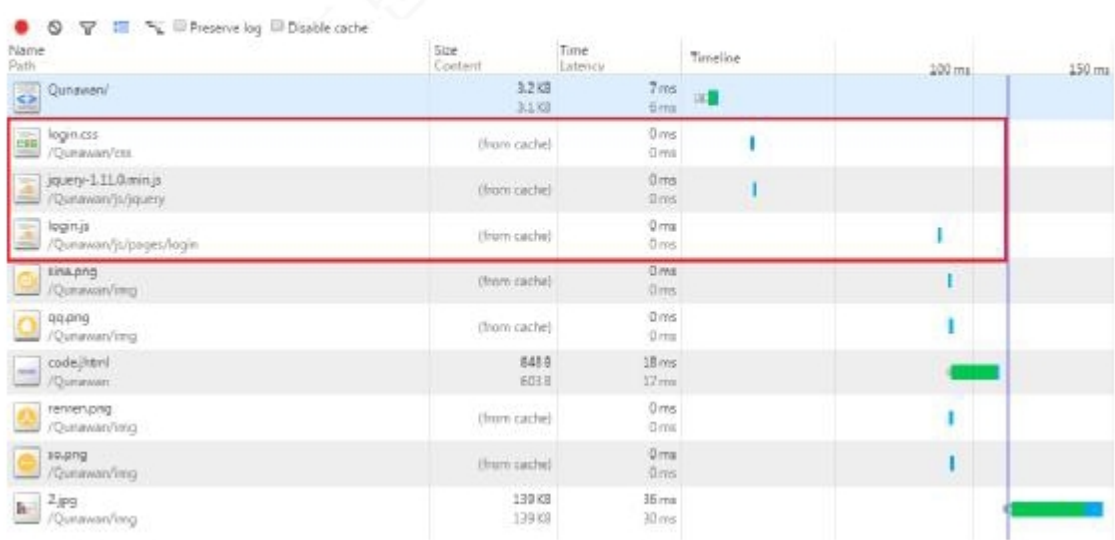


图 3-2-2

4-4.5. 对比两次访问结果可知：

1) 第一次访问某个网站页面之所以缓慢的原因是：网站所需的图片、js、css

等资源均需向服务端获取并下载，下载的时长视文件大小而定。

2) 以后每次访问网站页面之所以迅速的原因是：资源文件会被浏览器缓存

(cache)，不再需要从服务器端获取了。

4-5. 测试2：查看“响应头”与“响应体”的数据：

4-5.1. 在登录页面中分别输入：

1) 任意的用户名：18888888888。

2) 密码和验证码任意。

3) 点击登录按钮。

4-5.2. 在【Network】栏中点击请求链接【login.jhtml】：

1) Response Headers:

响应头中保存了大量服务器的基本信息，包括：Web容器的类型、响应时间、响应内容的数据MIME类型、响应内容的数据长度等。

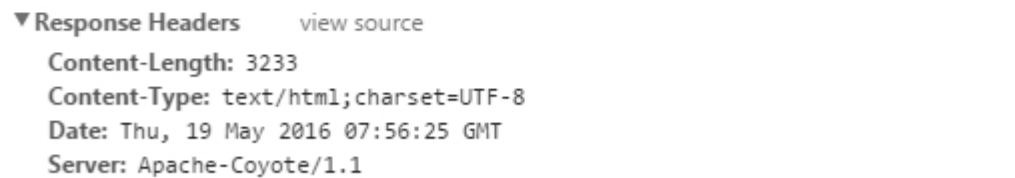


图 3-2-3

2) Response Body:

点击右侧“Response”选项卡，可以看到一份完整的HTML网页代码，这就是服务

器响应回客户端的完整网页数据，浏览器解析后就可以看到登录页面和报错信息。

```

1
2 <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
3 <html>
4 <head>
5 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
6 <title>去哪儿 登录</title>
7 <link type="text/css" rel="stylesheet" href="css/login.css" />
8 <script type="text/javascript" src="js/jquery/jquery-1.11.0.min.js"></script>
9 <script type="text/javascript" src="js/pages/login/login.js"></script>
10 </head>
11 <body>
12 <!-- 背景 开始 -->
13 <div class="backgrounds" id="backgrounds">
14 <!-- 表单框 -->
15 <div class="content" id="div_position">
16
17 <div class="login_framework">

```

图 3-2-4

#### 4.6. 测试3：查看“请求头”与“请求体”的数据：

##### 1) Request Headers:

请求头中保存了大量客户端的基本信息，包括：请求URL、请求的类型（GET/POST）、

当前正在使用的浏览器、操作系统、客户端IP地址等。

```

▼ Request Headers view source
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.8
Cache-Control: max-age=0
Connection: keep-alive
Content-Length: 40
Content-Type: application/x-www-form-urlencoded
Cookie: JSESSIONID=F48185D26A48D320F7FE28067808F486
Host: localhost:8080
Origin: http://localhost:8080
Referer: http://localhost:8080/WEB_INF/007/
User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/41.0.2228.89 Safari/537.36

```

图 3-2-5

##### 2) Form Data（请求体）：

从下图可知，用户在浏览器中的所输入的所有数据，都是通过请求体发送到Web容器

中的，请求体的格式是：控件的name属性 = 用户输入的数据。

```

▼ Form Data view source view URL encoded
name: 188888888888
password: 123456
code: test

```

图 3-2-6

## 任务 3、业务体验：未登录状态下访问站点

### 1、任务说明：

1-1. 完成效果：无

1-2. 任务目标：

1-2.1. 体验《去哪玩旅游网》的实际业务和相关界面。

1-3. 任务要求：

1-3.1. 在未登录状态下访问并体验《去哪玩旅游网》的相关页面。

1-3.2. 页面包括：首页、搜索、详情。

### 2、实现思路：

2-1. 请按【验证与测试步骤】完成本任务。

### 3、推荐步骤：

无。

### 4、验证与测试：

4-1. 访问首页

4-1.1. 运行该项目，并在浏览器地址栏输入【[http://localhost/PRJ\\_WTP\\_JEE\\_002/index.jhtml](http://localhost/PRJ_WTP_JEE_002/index.jhtml)】后敲击回车，进入首页：



图 3-3-1

## 4.2. 访问搜索页

4-2.1. 点击首页顶部的【搜索】按钮（无需输入任何关键字），进入搜索页：

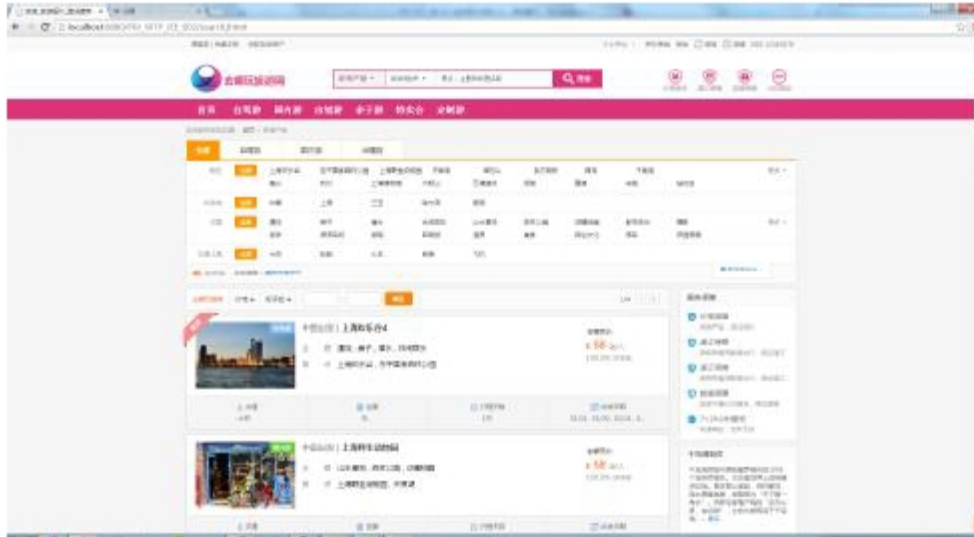


图 3-3-2

## 4.3. 访问详情页

4-3.1. 点击任意【产品图片】，进入行程产品详情页：



图 3-3-3

## 任务 4、业务体验：登录状态下访问站点

### 1、任务说明：

1-1. 完成效果：无

1-2. 任务目标：

1-2.1. 体验《去哪儿旅游网》的实际业务和相关界面。

1-3. 核心业务：

1-3.1. 体验《去哪儿旅游网》的实际业务和相关界面。

1-3.2. 在登录状态下访问并体验《去哪儿旅游网》的相关页面。

1-3.3. 页面包括：个人中心、下单、评论。

### 2、实现思路：

2-1. 请按【验证与测试步骤】完成本任务。

### 3、推荐步骤：

无。

### 4、验证与测试：

4-1. 用户登录

4-1.1. 运行项目，在登录页中输入用户名【18701721202】，密码【123456】进行登录。

4-2. 访问个人中心

4-2.1. 把鼠标放到首页顶部的【个人中心】处，点击下拉列表的【个人资料】则可进入个人资料设置页面：

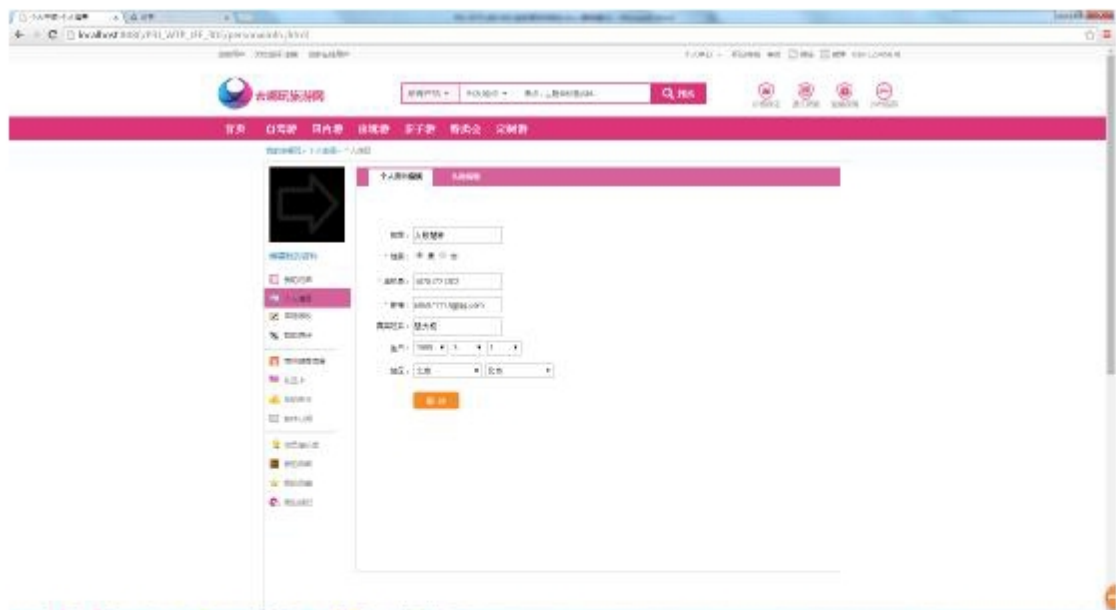


图 3-4-1

4-2.2. 点击左侧菜单的【我的约单】、【密码修改】、【我的点评】和【常用游客信息】

则可进入相应页面进行体验。

#### 4-3. 访问下单页面

4-3.1. 点击【首页】 - 【上海一日游】进入旅游产品详情页。

4-3.2. 在详情页选择好“出行日期”，上下翻动日历选择有价格的日期、选择人数，点

击开始预订，跳转到下单页面：



图 3-4-2

4-3.3. 选择紧急联系人、勾选旅游游玩人，点击去付款按钮，进入模拟付款页面。

4-3.4. 点击“确认付款”按钮，进入订单完成页面。

4-3.5. 点击底部“查看订单详情”进入订单页面，可以看到之前创建的旅游订单。

4-3.6. 再次进入“个人空间”，点击“我的评论”进入评论页面，选择最后一页。

4-3.7. 点击右侧评论图标，可以开始评论（如下图）：

订单号: 10521810  
好评: 100.0%  
产品名称: 《上海松江1天1夜》住上海金山大酒店4晚·免费游金山森林公园  
出行时间: 2016-06-01

景点 (1) ★★★★★  
酒店 (1) ★★★★★  
餐饮 (1) ★★★★★  
交通 (1) ★★★★★

精华秘籍: 超200字+美图+原创内容丰富实用, 满足以上即有机会被设为精华哦!  
旅途中的喜闻乐见, 美景美食美人, 都记录下来吧~

不知道写什么? 查看示例: [查看全部](#) 输入20-1000字

图片大小1M以下, 支持上传12张, 格式支持: jpg, jpeg

开始上传

图 3-4-3



## 场景总结

### Q1. 请您谈谈HTTP协议中请求的类型、数据组成和作用。

#### 1、 请求分为两种类型：POST请求、GET请求

1-1. POST请求： POST是一种相对安全但效率较低请求方式，所有客户端用户数据都会

通过请求体发送到服务器端，POST请求被大量运用在表单数据提交的场合。

例如：登录页面中，用户输入的“用户名”、“密码”、“验证码”都应该通过POST方式

提交请求，因为这种提交方式相对安全。

1-2. GET请求： GET请求是一种效率较高但安全级别较低请求方式，所有客户端用户数

据都会通过URL“明文”发送到服务器端，GET请求被大量运用在页面间相互跳转的场合。

#### 2、 请求数据分别存储于“请求头”与“请求体”内。

2-1. 请求头存放了客户端的信息数据，包括用户的IP地址、浏览器类型与版本、操作系统

类型与版本、浏览器解析字符集、操作系统默认语言等。

2-2. 请求体存放了用户在网页控件中填写的数据，以name=value的形式存储。

例如：页面有一个文本框存放于表单中，name为uid，当用户输入了123并提交请求时，请

求体内的数据为：uid=123。

**注意：只有POST请求会将用户数据保存于请求体中，GET请求的请求体始终为空。**

### Q2. 请您谈谈HTTP协议中响应的数据组成和作用。

#### 1、 响应数据分别存储于响应头与响应体内

1-1. 响应头存放了服务器端的信息数据，包括服务器的类型、协议类型与版本、响应数据

类型和字符集、响应数据大小与长度、响应资源的最后修改时间。

1-2. 响应体存放了完整的服务器响应结果，例如：用户可以看到完整网页HTML代码。