第七章实验报告

**实验名称: 学习使用AJAX实现一个简单的应用实例**

**学 院:计算机科学与工程学院/软件学院**

**专 业: 计算机科学与技术**

**年 级: 2021级**

**学 号: 202110803009**

**学生姓名: 欧阳俊杰**

**任课教师: 黄汉民**

实验目的：

学习使用AJAX实现一个简单的应用示例。

实验步骤：

1. 创建HTML页面，设置基本结构，包括一个按钮和一个用于显示数据的div。

2. 编写JavaScript代码，实现以下功能：

a. 当用户点击按钮时，触发一个名为ajaxRequest的函数。

b. 在ajaxRequest函数中，创建一个XMLHttpRequest对象，用于与服务器交互。

c. 设置XMLHttpRequest对象的onreadystatechange事件，以便在请求状态发生变化时执行相应操作。

d. 发送HTTP请求到服务器，请求一个名为data.php的文件，并设置请求类型为GET。

e. 当服务器响应就绪时，根据响应内容更新div中的文本内容。

3. 编写服务器端代码（data.php）：

a. 接收来自客户端的请求。

b. 读取请求参数，例如：用户名和密码。

c. 模拟数据库查询，获取用户信息。

d. 将用户信息编码为JSON格式，并发送回客户端。

实验代码：

HTML：

```html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>AJAX应用示例</title>

</head>

<body>

<button id="submitBtn">点击获取数据</button>

<div id="result"></div>

<script src="script.js"></script>

</body>

</html>

```

JavaScript（script.js）：

```javascript

function ajaxRequest() {

var xhr = new XMLHttpRequest();

xhr.onreadystatechange = function() {

if (xhr.readyState == 4 && xhr.status == 200) {

var resultDiv = document.getElementById("result");

resultDiv.innerHTML = xhr.responseText;

}

};

var submitBtn = document.getElementById("submitBtn");

var userName = "testUser";

var password = "testPassword";

var requestUrl = "data.php?userName=" + encodeURIComponent(userName) + "&password=" + encodeURIComponent(password);

xhr.open("GET", requestUrl, true);

xhr.send();

}

submitBtn.addEventListener("click", ajaxRequest);

```

服务器端代码（data.php）：

```php

<?php

$userName = $\_GET['userName'];

$password = $\_GET['password'];

// 模拟数据库查询，获取用户信息

$userInfo = array(

"userName" => "testUser",

"password" => "testPassword",

"email" => "test@example.com"

);

// 将用户信息编码为JSON格式

$json = json\_encode($userInfo);

// 发送响应

header("Content-Type: application/json; charset=utf-8");

echo $json;

?>

```

实验总结：

通过本实验，我们学习了如何使用AJAX技术实现一个简单的应用示例。实验中，我们创建了一个HTML页面，并通过JavaScript和服务器端代码实现了一个简单的登录验证功能。当用户点击按钮时，客户端发送请求到服务器，服务器返回用户信息，客户端根据响应内容更新页面内容。通过这个示例，我们了解了AJAX的基本工作原理和实际应用场景，为以后开发更加复杂的Web应用奠定了基础。