计算机与IT入门实验讲义

# 实验项目：Ajax+Json讲义

实 验 手 册

大连理工大学软件学院

创新实践基地

2016年6月

# 实验说明

本实验文档为软件学院14级《计算机与IT入门》课程编写，主要内容为ajax应用、json数据格式的读取，css样式表。结合这几项技术设计出一个在静态HTML页面中，实现脚本控制案例。

# 试验1 AJAX的应用

# AJAX 简介

**AJAX 是一种在无需重新加载整个网页的情况下，能够更新部分网页的技术。**

## 您应当具备的基础知识

在继续学习之前，您需要对下面的知识有基本的了解：

* HTML / XHTML
* CSS
* JavaScript / DOM

## 什么是 AJAX ？

AJAX = 异步 JavaScript 和 XML。

AJAX 是一种用于创建快速动态网页的技术。

通过在后台与服务器进行少量数据交换，AJAX 可以使网页实现异步更新。这意味着可以在不重新加载整个网页的情况下，对网页的某部分进行更新。

传统的网页（不使用 AJAX）如果需要更新内容，必需重载整个网页面。

有很多使用 AJAX 的应用程序案例：新浪微博、Google 地图、开心网等等。

## Google Suggest

在 2005 年，Google 通过其 Google Suggest 使 AJAX 变得流行起来。

Google Suggest 使用 AJAX 创造出动态性极强的 web 界面：当您在谷歌的搜索框输入关键字时，JavaScript 会把这些字符发送到服务器，然后服务器会返回一个搜索建议的列表。

# AJAX 实例

## AJAX 实例解释

上面的 AJAX 应用程序包含一个 div 和一个按钮。

div 部分用于显示来自服务器的信息。当按钮被点击时，它负责调用名为 btnAdd () 单击事件的匿名函数：

<html>

<body>

<div id="divframe">

<div class="loadTitle">

<input type="button" value="添加数据" id="btnAdd"/>

<input type="button" value="移除数据" id="btnRemove"/>

</div>

<div id="jsonTip"></div>

</div>

</body>

</html>

接下来，在页面的 head 部分添加一个 <script> 标签。该标签中包含了这个 btnAdd (),单击事件的闭包（匿名）函数：

<head>

<script type="text/javascript">

$(function(){

$("#btnAdd").click(function() {

$.ajax({

type:"GET",//或者post

url:"list.json", //提交地址，等同表单的action=“list.json”

dataType:"json", //返回数据的格式

success:function(data){

//读取成功,返回数据的处理

}

});

}

}

</script>

</head>

# 试验2 JSON 数据的读取

# JSON 简介

## 类似 XML

* JSON 是纯文本
* JSON 具有“自我描述性”（人类可读）
* JSON 具有层级结构（值中存在值）
* JSON 可通过 JavaScript 进行解析
* JSON 数据可使用 AJAX 进行传输

## 相比 XML 的不同之处

* 没有结束标签
* 更短
* 读写的速度更快
* 能够使用内建的 JavaScript eval() 方法进行解析
* 使用数组
* 不使用保留字

## 为什么使用 JSON？

对于 AJAX 应用程序来说，JSON 比 XML 更快更易使用：

#### 使用 XML

* 读取 XML 文档
* 使用 XML DOM 来循环遍历文档
* 读取值并存储在变量中

#### 使用 JSON

* 读取 JSON 字符串
* 用 eval() 处理 JSON 字符串

**JSON 语法是 JavaScript 语法的子集。**

## JSON 语法规则

JSON 语法是 JavaScript 对象表示法语法的子集。

* 数据在名称/值对中
* 数据由逗号分隔
* 花括号保存对象
* 方括号保存数组

## JSON 名称/值对

JSON 数据的书写格式是：名称/值对。

名称/值对包括字段名称（在双引号中），后面写一个冒号，然后是值：

"firstName" : "John"

这很容易理解，等价于这条 JavaScript 语句：

firstName = "John"

## JSON 值

JSON 值可以是：

* 数字（整数或浮点数）
* 字符串（在双引号中）
* 逻辑值（true 或 false）
* 数组（在方括号中）
* 对象（在花括号中）
* null

## JSON 对象

JSON 对象在花括号中书写：

对象可以包含多个名称/值对：

{ "firstName":"John" , "lastName":"Doe" }

这一点也容易理解，与这条 JavaScript 语句等价：

firstName = "John"

lastName = "Doe"

## JSON 数组

JSON 数组在方括号中书写：

数组可包含多个对象：

{

"employees": [

{ "firstName":"John" , "lastName":"Doe" },

{ "firstName":"Anna" , "lastName":"Smith" },

{ "firstName":"Peter" , "lastName":"Jones" }

]

}

在上面的例子中，对象 "employees" 是包含三个对象的数组。每个对象代表一条关于某人（有姓和名）的记录。

## JSON 使用 JavaScript 语法

因为 JSON 使用 JavaScript 语法，所以无需额外的软件就能处理 JavaScript 中的 JSON。

通过 JavaScript，您可以创建一个对象数组，并像这样进行赋值：

## JSON 文件

* JSON 文件的文件类型是 ".json"
* JSON 文本的 MIME 类型是 "application/json"

## 把 JSON 文本转换为 JavaScript 对象

JSON 最常见的用法之一，是从 web 服务器上读取 JSON 数据（作为文件或作为 HttpRequest），将 JSON 数据转换为 JavaScript 对象，然后在网页中使用该数据。

为了更简单地为您讲解，我们使用字符串作为输入进行演示（而不是文件）。

## JSON 实例 - 来自字符串的对象

创建包含 JSON 语法的 JavaScript 字符串：

var txt = '{ "employees" : [' +

'{ "firstName":"Bill" , "lastName":"Gates" },' +

'{ "firstName":"George" , "lastName":"Bush" },' +

'{ "firstName":"Thomas" , "lastName":"Carter" } ]}';

由于 JSON 语法是 JavaScript 语法的子集，JavaScript 函数 eval() 可用于将 JSON 文本转换为 JavaScript 对象。

eval() 函数使用的是 JavaScript 编译器，可解析 JSON 文本，然后生成 JavaScript 对象。必须把文本包围在括号中，这样才能避免语法错误：

var obj = eval ("(" + txt + ")");

## JSON 解析器

**提示：**eval() 函数可编译并执行任何 JavaScript 代码。这隐藏了一个潜在的安全问题。

使用 JSON 解析器将 JSON 转换为 JavaScript 对象是更安全的做法。JSON 解析器只能识别 JSON 文本，而不会编译脚本。**（本次课程不做要求，课后自己完善安全性的功能。）**

JSON提供了json.js包，下载<http://www.json.org/json.js> 后，将其引入然后就可以简单的使用object.toJSONString()转换成JSON数据。

在浏览器中，这提供了原生的 JSON 支持，而且 JSON 解析器的速度更快。

较新的浏览器和最新的 ECMAScript (JavaScript) 标准中均包含了原生的对 JSON 的支持。

|  |  |
| --- | --- |
| **Web 浏览器支持** | **Web 软件支持** |
| * Firefox (Mozilla) 3.5 * Internet Explorer 8 * Chrome * Opera 10 * Safari 4 | * jQuery * Yahoo UI * Prototype * Dojo * ECMAScript 1.5 |

# JSON实例

## JSON 数据的读取

通过我们的编辑器，您可以在线编辑 JavaScript 代码，然后通过点击一个按钮来查看结果：

<html>

<body>

<h2>在 JavaScript 中创建 JSON 对象</h2>

<p>

Name: <span id="jname"></span><br />

Age: <span id="jage"></span><br />

Address: <span id="jstreet"></span><br />

Phone: <span id="jphone"></span><br />

</p>

<script type="text/javascript">

var JSONObject= {

"name":"Bill Gates",

"street":"Fifth Avenue New York 666",

"age":56,

"phone":"555 1234567"};

document.getElementById("jname").innerHTML=JSONObject.name

document.getElementById("jage").innerHTML=JSONObject.age

document.getElementById("jstreet").innerHTML=JSONObject.street

document.getElementById("jphone").innerHTML=JSONObject.phone

</script>

</body>

# 试验3 CSS 样式表

# CSS 简介

## CSS 概述

* CSS 指层叠样式表 (**C**ascading **S**tyle **S**heets)
* 样式定义**如何显示** HTML 元素
* 样式通常存储在**样式表**中
* 把样式添加到 HTML 4.0 中，是为了**解决内容与表现分离的问题**
* **外部样式表**可以极大提高工作效率
* 外部样式表通常存储在 **CSS 文件**中
* 多个样式定义可**层叠**为一

## 样式解决了一个普遍的问题

HTML 标签原本被设计为用于定义文档内容。通过使用 <h1>、<p>、<table> 这样的标签，HTML 的初衷是表达“这是标题”、“这是段落”、“这是表格”之类的信息。同时文档布局由浏览器来完成，而不使用任何的格式化标签。

由于两种主要的浏览器（Netscape 和 Internet Explorer）不断地将新的 HTML 标签和属性（比如字体标签和颜色属性）添加到 HTML 规范中，创建文档内容清晰地独立于文档表现层的站点变得越来越困难。

为了解决这个问题，万维网联盟（W3C），这个非营利的标准化联盟，肩负起了 HTML 标准化的使命，并在 HTML 4.0 之外创造出样式（Style）。

所有的主流浏览器均支持层叠样式表。

## 样式表极大地提高了工作效率

样式表定义如何显示 HTML 元素，就像 HTML 3.2 的字体标签和颜色属性所起的作用那样。样式通常保存在外部的 .css 文件中。通过仅仅编辑一个简单的 CSS 文档，外部样式表使你有能力同时改变站点中所有页面的布局和外观。

由于允许同时控制多重页面的样式和布局，CSS 可以称得上 WEB 设计领域的一个突破。作为网站开发者，你能够为每个 HTML 元素定义样式，并将之应用于你希望的任意多的页面中。如需进行全局的更新，只需简单地改变样式，然后网站中的所有元素均会自动地更新。

## 多重样式将层叠为一个

样式表允许以多种方式规定样式信息。样式可以规定在单个的 HTML 元素中，在 HTML 页的头元素中，或在一个外部的 CSS 文件中。甚至可以在同一个 HTML 文档内部引用多个外部样式表。

### 层叠次序

**当同一个 HTML 元素被不止一个样式定义时，会使用哪个样式呢？**

一般而言，所有的样式会根据下面的规则层叠于一个新的虚拟样式表中，其中数字 4 拥有最高的优先权。

1. 浏览器缺省设置
2. 外部样式表
3. 内部样式表（位于 <head> 标签内部）
4. 内联样式（在 HTML 元素内部）

因此，内联样式（在 HTML 元素内部）拥有最高的优先权，这意味着它将优先于以下的样式声明：<head> 标签中的样式声明，外部样式表中的样式声明，或者浏览器中的样式声明（缺省值）。

# CSS实例

## 设置DIV的样式（class、id）

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<title>getJSON获取数据</title>

<style type="text/css">

#divframe{

border:1px solid #999;

width:500px;

margin:0 auto;

}

.loadTitle{

background:#CCC;

height:30px;

}

</style>

</head>

<body>

<div id="divframe">

<div class="loadTitle">

<input type="button" value="添加数据" id="btnAdd"/>

<input type="button" value="移除数据" id="btnRemove"/>

</div>

<div id="jsonTip">

</div>

</div>

</body>

</html>

参照：<http://www.w3school.com.cn/example/csse_examples.asp>

# 综合案例

## AJAX+JSON模拟登录功能的简单实现



## AJAX+JSON读取数据的简单实现



# 作业

### 登录失败如何清空用户名和密码文本框的数据。

### 2.请学生自由发挥美化页面。