复习：

数据库服务器 —— SQL

Web服务器：静态&动态 —— PHP

PHP访问MySQL数据库服务器：

$conn = mysqli\_connect('','','','',3306);

$sql = "";

$result = mysqli\_query($conn, $sql);

HTTP协议：

请求消息

响应消息

练习： 9：40

(1)编写SQL：数据库-meishichina，表名-dish( did, dname, pic, author, pubTime )

设置编码-删除-新建-进入-创建表-插入数据-查询

(2)编写PHP：dish\_add.php，接收客户端提交的dname, pic, author，插入到数据库中，返回SUCC/ERR

(3)编写HTML：dish\_add.html，显示一个表单，用于提交新菜谱的信息，实现菜谱添加功能

(4)编写PHP：dish\_select.php，在一个DIV列表中显示出所有的菜谱；每个菜谱右上角添加一个“删除”按钮，点击后提交给dish\_delete.php

(5)编写PHP：dish\_delete.php，接收客户端提交的did，执行删除操作

今日目标

(1)继续HTTP协议

(2)AJAX —— 重点&难点

1.补充：PHP常用函数

(1)require('x.php')：把指定文件中的内容包含在当前位置——服务器端页面包含

(2)mysqli\_fetch\_all($result, MYSQLI\_ASSOC)：一次性的抓取查询到的所有的记录行 PHP>=5.4

(3)header()：修改响消息头部

2.Content-Type：头部

用于指定当前消息的主体中“内容类型”：

**请求消息中可取值：**

(1)text/plain：请求主体内容是普通文本，未经编码——有些服务器拒绝接收这样的数据（如PHP服务器）

(2)application/x-www-form-urlencoded：请求主体数据时经过编码后的表单数据(encodeURIComponent())

(3)multipart/form-data：请求主体数据包含文件上传

**响应消息中可取值：**

(1)text/plain

(2)text/html

(3)text/css

(4)application/javascript

(5)application/xml

(6)application/json

(7)image/jpeg

(8)image/png

(9)audio/mpeg3

(....).... 100+ MIME多种

|  |
| --- |
| .html / .xhtml / .htm  .jpg / .jpeg  .mp3 / .mpeg3  MIME： Multipurpose Internet Mail Extension，定义了一批固定名称的文件类型，如image/jpeg、text/html、application/js.... |

3.如何修改响应消息的头部

**(1)修改Web服务器的配置文件，如httpd.conf**

问题：会影响到当前服务器上所有的网站、有些情况下没有权限修改

**(2)在HTML设置meta元标签**

<meta http-equiv="Cache-Control" content="no-cache">

问题：修改的并非真正的响应头部，可能是无效；仅对HTML文档可用

**(3)使用服务器端动态语言来修改，如PHP**

header('Content-Type: image/jpeg')

演示：使用PHP向客户端输出图片类型的响应消息——验证码图片

|  |
| --- |
| AJAX难学！  把之前所学的HTML、CSS、JS、DOM、jQuery、SQL、PHP、HTTP、XML、JSON揉在一起！ |

|  |
| --- |
| 面试题：谈一下对AJAX的理解 |

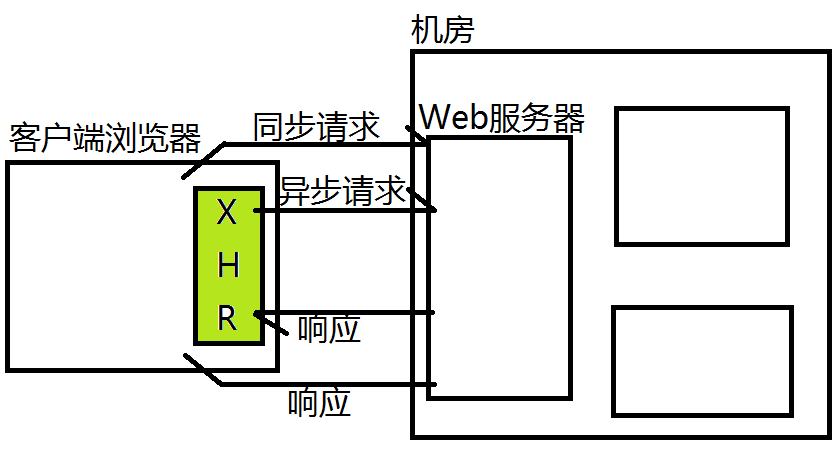
4.AJAX概述

AJAX：Asynchronous Javascript And XML，异步的JS和XML。2001，Google为了改进搜索的用户体验，提出了GoogleSugguest效果，正式提出了AJAX的概念。AJAX = HTML + CSS + JS + DOM + HTTP + XML，目标：在无刷新无提交的情况下，实现页面内容的局部更新——数据来自于服务器。

常见应用：实时数据获取(如股票走势图)、搜索建议、聊天室、SPA

AJAX应用依赖于浏览器底层提供的核心对象：

**XMLHttpRequest：用于向Web服务器发起异步请求，并接收响应消息**。



|  |
| --- |
| 浏览器发起的HTTP请求的类型：  (1)同步请求： 超链接跳转、JS跳转、表单提交  (2)异步请求： 使用XHR发起 |

XMLHttpRequest对象的常用成员：

|  |
| --- |
| **成员属性：**  readyState: 0，就绪状态，描述XHR请求-响应过程进行到哪一步了，可取值为：  0 - UNSENT 表示请求尚未发送  1 - OPENED 表示XHR已经打开了到服务器的连接  2 - HEADERS\_RECEIVED 表示XHR已经接收到了响应消息头部  3 - LOADING 表示XHR正在加载响应消息主体  4 - DONE 表示XHR已经加载完成响应主体    status：0，表示响应消息的状态码， readyState变为2才有值  statusText：""，表示响应消息的原因短句，readyState变为2才有值  responseText: ""，表示响应消息的主体数据，readyState变为3开始有值，变为4时值稳定下来 |
| **成员方法：**  open(method, url, isAsync) 打开到服务器的连接  send( null/body ) 发送请求消息  setRequestHeader() 设置请求消息头部  getResponseHeader() 得到响应消息头部 |
| **成员事件：**  onreadystatechange：每次xhr.readyState属性的值发生改变都会触发该事件，监听此事件可以知道请求-响应进行到哪一步了 |

使用XHR发起异步请求步骤：

(1)创建XHR对象 —— 每个XHR只能发一个请求

var xhr = new XMLHttpRequest();

(2)绑定监听函数，处理XHR的每一次状态改变

xhr.onreadystatechange = function(){ }

(3)打开到Web服务器的连接

xhr.open('GET','9.php', true);

(4)发送请求消息主体

xhr.send(null);

5.使用XHR发起异步的GET请求

演示：异步验证用户名是否存在

实现步骤：

(1)编写SQL：创建数据库sohu，包含表users(uid,uname,upwd)， 插入三行记录，查询出所有记录

(2)编写PHP：check\_uname.php，接收客户端提交的uname，执行SELECT语句，返回cunzai或bucunzai

SELECT \* FROM users WHERE uname='?'

(3)编写HTML：user\_register.html，表单中当输入框失去焦点时，获取用户的输入，发起异步请求，验证该用户名是否存在

var xhr = new XMLHttpRequest();

xhr.onreadystatechange = function(){

if(xhr.readyState===4){

if(xhr.status===200){ //响应完成且成功

}else { //响应完成但有问题

}

}

}

xhr.open('GET', 'x.php?k1=v1&k2=v2&k3=v3', true);

xhr.send(null);

6.使用XHR发起异步的POT请求

//1 var xhr = new XMLHttpRequest();

//2 xhr.onreadystatechange = function(){

if(xhr.readyState===4){

if(xhr.status===200){ //响应完成且成功

}else { //响应完成但有问题

}

}

}

//3 xhr.open('POST', 'x.php', true);

**//3.5 xhr.setRequestHeader('Content-Type', 'application/x-www-form-urlencoded'); //修改请求消息头部**

//4 xhr.send('k1=v1&k2=v2&k3=v3');

练习内容：

(1)使用XHR对象发起异步的HTTP请求——成绩录入系统

<h1>成绩录入系统</h1>

学生姓名： <input id="stuName">

语文成绩： <input id="chinese">

数学成绩： <input id="math">

<input type="button" value="保存成绩">

要求：点击“保存成绩”按钮，发起异步HTTP请求，把数据提交给服务器端的PHP页面，INSERT到MySQL数据库，返回'succ'或'err'提示文字。客户端弹出一个alert("保存成功/失败")。用户点击确定后，清空输入框内容，继续录入下一个学生成绩。

实现步骤：

(1)编写SQL：创建数据库-tarena，包含表-score( sid、stuName、chinese、math )；插入三行测试数据，查询所有记录行。

(2)编写PHP：创建score\_add.php，接收客户端提交的stuName / chinese / math数据，执行INSERT，把数据插入到数据库，返回succ或err

(3)编写HTML：创建score\_add.html，没有FORM,显示三个输入框及提交按钮，一点击发起异步请求，将数据提交给服务器端PHP页面

提示：先用GET方式提交数据；再用POST方式提交数据——注意3.5！