必需禁用的两项服务：

Windows Firewall

Web部署代理服务 / IIS

点击Netstat按钮，确保如下三个端口未被占用：

1) 80

2) 443

3) 3306

删除c:/xampp/htdocs/下的所有内容

第一阶段： HTML(3) + CSS(5)

第二阶段：JS(7) + DOM(6)

第三阶段： AJAX(8.5) + HTML5(6)+ Bootstrap(5)

第四阶段： 框架

数据库服务器 —— SQL

Web服务器 —— PHP

动态网页 —— SQL + PHP

HTTP协议 —— 纯理论

原生AJAX —— 整合目前所有所学

1.服务器概述



2.数据库服务器概述

数据库的种类：

(1)树形数据库

(2)网状数据库

**(3)关系型数据库** —— 80年代至今

SQLite、MySQL、PostgreSQL、SQLServer、Oracle、DB2

(4)对象型数据库

(5)非关系型数据库 —— MongoDB

关系型数据库的数据结构：

Server(服务器) -> Database(数据库) ->Table(表) -> Row(行) -> Column(列)

|  |
| --- |
| XAMPP(服务器软件套装)= Apache + MySQL/MariaDB + PHP  下载：<https://www.apachefriends.org/zh_cn/index.html> |

3.关系型数据库MySQL的使用

最流行的开源数据库，目前有两个分支：MySQL 和 MariaDB

使用MySQL数据库系统的步骤：

(1)下载并安装MySQL服务器端程序

C:\xampp\mysql\bin\mysqld.exe

Deamon精灵、守护者、守护程序、服务器程序

(2)启动MySQL服务器端程序

双击mysqld.exe或者点击“启动”按钮，只要确保3306端口被mysqld.exe占用即可

-----------------------------------------------

(3)下载并安装MySQL客户端程序，可以选择下列三者之一

(3.1)命令行界面工具

C:\xampp\mysql\bin\mysql.exe -uroot**没有分号！！**

(3.2)桌面版本工具

(3.3)网页版本工具

<http://127.0.0.1/phpmyadmin/>

(4)发送SQL命令给服务器执行，例如：

show databases;  **SQL命令都以分号结尾！！**



4.SQL语言——重点

SQL：Strutured Query Language，一种专用于操作关系型数据库的语言，可以实现“**增删改查**”数据库服务器中的数据。

执行SQL语句的两种方式：

**(1)交互模式：**输入一行执行一行 —— 适用于临时少量操作数据

**(2)脚本模式：**把多条SQL命令写在一个文本文件中，一次性全部提交给服务器执行 ——适用于临时大量操作数据

**mysql -uroot < d:/jd.sql小于号表示“导入”**

练习：

(1)创建保存IBM公司部门和员工信息的数据库： 库名-ibm

(2)创建一个部门表-dept( did, dname, location)

(3)插入三个部门记录行，10-.... 20-.... 30-....

(4)创建一个员工表-emp( eid, ename, salary, hireDate, isOnDuty, deptId)

(5)为每个部门插入两个员工记录

(6)查询出所有的部门信息

(7)查询出市场部的编号

(8)查询出所有的员工信息

(9)查询出市场部的所有的员工信息

5.Web服务器

用于向客户端提供Web内容（网页、图片、音频视频、数据）的服务器。



Web服务器的两种类型：

**(1)静态Web服务器：**提供的内容任何人任何时间访问都是一成不变的

静态范畴的技术：HTML、CSS、JS、Flash、图片、音视频

**(2)动态Web服务器：**提供的内容不同人不同时间访问是可能变化的

动态范畴的技术：JSP、ASP.NET、PHP、Node.js

上述四门技术的本质类似：都是在HTML中嵌入一些动态编程语言（访问数据库、访问央行、访问证券交易所等...），让静态网页中可以显示动态的数据的

晚间练习：

(1)创建1.php，向客户端输出1个\*

(2)创建2.php，向客户端输出50个\*

(3)创建3.php，向客户端输出5行10列个\*

(4)创建4.php，向客户端输出“九九乘法表”

--------------------------------------------

(5)创建jd.sql文件，根据要求编写必需的SQL语句

1)设置SQL编码方式

2)删除数据库-jd，如何存在的话

3)创建数据库-jd，指定字符编码方式

4)开始使用数据库-jd

5)创建产品信息表-product(pid-编号, pname-名称, price-单价,isOnSale-是否特价, pic-产品图片文件路径)

6)向产品表中插入3行记录

7)创建产品评论表-comment(cid, userName-用户名, phone-用户联系电话, title-评论标题, content-评论内容, pubTime-发布时间, productId-所评论的产品编号)

8)为每个产品添加两三条评论

9)查询所有产品

10)查询所有评论

11)查询出1号产品的所有评论

12)删除1号商品及所有评论

13)修改2号商品编号为200,同时修改其所对应的所有评论

第二天

复习：

数据库服务器：存储项目中的数据 —— SQL

Web服务器：

静态Web服务器 —— HTML/CSS/JS/Flash/音视频

动态Web服务器 —— JSP/ASP.NET/PHP/Node.js

使用MySQL的步骤：

(1)下载并安装MySQL服务器端

C:\xampp\mysql\bin\mysqld.exe

(2)启动MySQL服务器端启动

确保3306端口是否已经被mysqld.exe占用

------------------------------------

(3)在客户端下载并安装MySQL客户端程序

命令行版：C:\xampp\mysql\bin\mysql.exe

网页版：<http://127.0.0.1/phpmyadmin>

(4)使用客户端程序连接服务器端

交互模式：mysql -uroot

脚本模式：mysql -uroot < e:/xx.sql

练习：

(1)设置SQL语句编码方式

(2)试着删除数据库ifeng

(3)创建数据库ifeng，字符编码UTF8

(4)进入数据库ifeng

(5)创建新闻表news(nid, title, content, isOnTop, pubTime, viewCount)

(6)插入3条新闻纪录

(7)创建新闻评论表comment(cid, content, userName, phone, pubTime)

(8)插入6条新闻评论

(9)查询出所有的新闻

(10)查询出所有的评论

(11)查询出1号新闻所有的评论

注意：DATE类型的表示范围不同的数据库/编程语言不同的！真正的项目中使用BIGINT代替！

今日目标：

(1)PHP语法 —— 掌握

(2)PHP访问MySQL —— 重点

1.面试题： 如何自学一门新语言？

(1)了解背景：历史、现状、发展、特点、适用领域

(2)搭建开发运行环境，编写HelloWorld

(3)数据类型

(4)变量和常量

(5)运算

(6)逻辑结构

(7)函数和对象

(8)通用小程序，如九九乘法表

(9)常用工具、组件、框架

(10)实用小项目

2.动态Web服务器(PHP)的运行环境搭建



(1)下载并安装静态Web服务器，如ApacheHttpd、MS-IIS、NginX——“面包小姐”

C:\xampp\apache\bin\httpd.exe

(2)下载并安装PHP程序的解释器——“烤箱”

C:\xampp\php\php.exe

(3)修改Web服务器的配置文件，告诉它如果接收到.php请求，转交给PHP解释器来处理——“教面包小姐如何使用烤箱”

上述三步比较麻烦，直接使用XAMPP套装简化。

(4)服务器端编写网页，保存在一个特定的目录，如c:/xampp/htdocs——“面包货柜”

(5)启动Web服务器——"开门营业"

-------------------------------------------------------

(6)客户端通过浏览器，使用HTTP协议远程访问Web服务器上的页面

3.PHP中的数据类型——重点

**(1)值类型/原生类型**

string 1)拼接用点号 2)双引号字符串中可以包含变量名

boolean

int/integer

float/double

**(2)复合类型**

array 1)数组不是对象，没有toString()，不能直接输出 2)索引数组和关联数组 3)添加新元素$arr[]=值; 4)可以使用for或foreach遍历数组中的元素

|  |
| --- |
| $arr = array(10, 30, 50); //PHP5.4之前  $arr = [10,30,50]; //PHP5.4之后 |

object

**(3)其它类型**

NULL

resource（资源类型）

练习：使用PHP创建一个数组，保存５个员工的信息（ename/ sex / salary/birthday/ pic），把这五个员工数据输出在一个TABLE中

练习：使用PHP创建一个数组，保存四个商品的信息( pid, pname, price, pic )，把四个商品的信息输出到一个如下的列表中：



4.PHP中的变量和常量

定义变量：

$变量名 = 值;

使用变量：

echo $变量名; //输出变量的值

var\_dump($变量名); //查看变量的类型和值

定义常量：

const 常量名 = 值; //PHP5.4之后引入

define('常量名', 值); //PHP5.4之前

使用常量：

echo 常量名;

5.PHP中运算符

算术运算

比较运算

逻辑运算

三目运算

字符串拼接.

6.PHP中逻辑结构

选择结构： if...else ... switch...case...

循环结构： for while... do...while foreach($arr as $v)

7.PHP中如何声明函数

function add($num1, $num2){

$sum = $num1 + $num2;

return $sum;

}

8.通用小程序

......

9.常用函数和对象

查询参考手册，边学边查

10.PHP操作MySQL数据库的相关函数



PHP提供了两套函数库，操作MySQL数据库：

mysql\_xxx( ) //早期版本

mysqli\_xxx( ) //Improved，性能提高版

**PHP代码操作MySQL服务器的步骤(与命令行中连接的过程一样)——重点**

(1)连接到MySQL服务器

mysqli\_connect(...)

(2)编写SQL命令，提交给MySQL服务器执行

$sql = "INSERT/DELETE/UPDATE/SELECT....";

mysqli\_query(...)

(3)查看执行结果

......

(4)关闭数据库连接(可选)

11.演示：使用PHP提交INSERT语句——添加新闻

|  |
| --- |
| HTTP协议规定如何给一个服务器端的页面提交请求数据：  [http://127.0.0.1/x.phpHYPERLINK "http://127.0.0.1/x.php?uname=tom&upwd=123456"?uname=tomHYPERLINK "http://127.0.0.1/x.php?uname=tom&upwd=123456"&HYPERLINK "http://127.0.0.1/x.php?uname=tom&upwd=123456"upwd=123456](http://127.0.0.1/x.php?uname=tom&upwd=123456) |

12.练习：使用PHP实现新闻删除功能

(1)创建SQL，编写必需的news表

(2)创建PHP，news\_delete.php，接收客户端提交的新闻编号：nid，连接数据库，提交DELETE，输出删除成功或失败

(3)创建HTML，news\_delete.html，包含一个表单，“请输入要删除的新闻的编号：”,点击“提交”按钮，把用户输入提交给news\_delete.php实现新闻删除

day03

复习：

数据库服务器——SQL

Web服务器：静态和动态——PHP

php语法：

(1)了解背景

(2)搭建环境

(3)数据类型

值类型：string boolean int float

复合类型：array object

其它类型：null resource

(4)变量和常量： $变量名=值; const 常量名=值;

(5)运算 . .=

(6)逻辑结构foreach($arr as $k=>$v){}

(7)函数和对象 function 函数名(形参){ }

(8)通用小程序

(9)常用函数、对象、组件、框架

(10)实用小项目

练习：

(1)编写SQL：创建数据库dangdang，包含表book(bid, pic, bname, price, pubDate, hasStock-是否有货)，试着插入三行记录，查询出所有记录

(2)编写PHP：创建book\_add.php，接收客户端提交的请求数据($\_REQUEST[''])，执行INSERT，插入到数据库，返回SUCC/ERR

(3)编写HTML：创建book\_add.html，显示一个表单，将用户的输入提交给PHP，完成添加

(4)编写PHP：创建book\_delete.php，接收客户端提交的书籍编号bid，执行DELETE，从数据库中删除指定的书籍，返回SUCC/ERR

(3)编写HTML：创建book\_delete.html，显示一个表单，将用户的输入的待删除的书籍编号提交给PHP，完成删除

今日目标：

(1)PHP执行数据库操作——重点

(2)HTTP协议——重点

1.补充：SQL语句的分类

**(1)DDL：Data Define Language**，数据定义语言——定义表的列结构

CREATE、DROP、ALTER、TRUNCATE

**(2)DML：Data Manipulate Language**，数据操作语言——操作表的记录行

INSERT、DELETE、UPDATE

**(3)DQL：Data Query Language**，数据查询语言——不影响表的行和列

SELECT

**(4)DCL：Data Control Language**，数据控制语言——控制用户的权限

GRANT、REVOKE

**PHP中的$result = mysqli\_query($conn, $sql)的返回值：**

1)无论什么语句，只要执行失败(如SQL语法错误)，一律返回false；

2)如果是DML执行成功，返回true；

3)如果是DQL执行成功，返回查询结果集；

2.补充：PHP常用函数

(1) @ 放在一行最前面，用于压制错误消息的输出

(2)die() 终止当前页面的执行，可以输出一个错误消息

(3) strtotime() 把一个形如'yyyy-MM-dd'格式在日期时间转换为一个表示秒数整数值

(4) mysqli\_affected\_rows($conn)返回刚刚执行的DML语句影响的行数

(5) mysqli\_fetch\_rows($result)把一行记录返回为一个索引数组

(6) mysqli\_fetch\_assoc($result)把一行记录返回为一个关联数组

练习：读取所有的书籍记录，输出在一个TABLE元素中，使用循环抓取

while( true ){

//抓取一行

//读取内容为null则退出循环

}

午间练习：完成书籍的修改功能

提示：

当用户点击每个书籍记录后面的“修改”链接后，需要跳转到book\_update\_select.php页面，执行SELECT语句，根据客户端提交的书籍编号，查询出书籍信息，输出在一个类似“添加书籍”的表单中；

用户修改完成后，再点击“保存修改按钮”，把修改后的数据提交给book\_update.php页面，执行UPDATE语句。

day04

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

复习：

数据库服务器 —— SQL

Web服务器：静态&动态 —— PHP

HTTP协议：

**请求消息的格式：**

POST /x.php HTTP/1.1

Host: [www.tmooc.cn](http://www.tmooc.cn/)

Connection: keep-alive

Cache-Control: no-cache

Content-Length: 33

Content-Type: application/x-www-form-urlencoded

uname=%BC%38%4A&upwd=123

**响应消息格式：**

HTPP/1.1 200 OK

Date:

Server:

Last-Modified:

Content-Length: 1345

Content-Type: text/css

<html><head>......</html>

AJAX：异步的JS和XML，作用：无刷新无提交的情况下，实现页面局部更新——数据来自于服务器。核心对象：XMLHttpRequest —— 发起异步HTTP请求，接收服务器返回的响应消息。

使用XHR的四步：

//1 var xhr = new XMLHttpRequest();

//2 xhr.onreadystatechange = function(){ }

//3 xhr.open(method, url, isAsync)

//4 xhr.send( body )

练习：

**(1)编写SQL**：数据库-jd，数据表-product( pid, pname, pic, price, produceDate )，插入三行数据，查询出所有记录

**(2)编写PHP**：product\_add.php，接收客户端提交的pname、pic、price、produceDate等数据，执行INSERT，保存入数据库，返回succ或err

**(3)编写HTML**：product\_add.html，没有FORM元素，仅提供四个输入框，和一个A元素(“保存产品记录”)，点击后读取用户的输入，发起异步请求，提交给PHP页面，执行添加操作，根据服务器返回的结果，alert()出提示内容。

day05

今日目标：

(1)补充HTTP小知识——了解

(2)使用XHR发起两种请求 —— GET/POST

(3)使用XHR接收五种请求 —— 重点



1.小知识：HTTP和HTTPS

HTTP: 80 消息都是明文传输

HTTPS: 443 消息发送之前要加密，接收之前需要解密

HTTP over SSL

2.小知识：通过HTTP协议进行网站的访问优化

(1)域名解析

尽可能减少域名解析次数——减少跨站外部资源的引用

(2)创建连接

努力减少连接创建次数——使用持久连接，避免重复连接

(3)发送请求

尽力减少请求次数——合理设置缓存时间、资源合并

(4)等待响应

提高服务器端运行速度——提高数据运算及查询速度

(5)接收响应

尽可能减小响应数据长度——缩小文件体积，启用压缩

3.ES6新特性——``字符串

使用``括起来的字符串，可以换行，还可以包含变量的拼接${ }

var ename = 'tom';

var age = 20;

var data = `ENAME: ${ename} AGE: ${age}`;

var data = `ENAME: ${ename.toUpperCase()}

AGE: ${age>18?'成年人':'未成年'}`;

4.使用XHR接收响应消息——text/plain

**服务器端：**

header('Content-Type: text/plain');

echo 'succ';

**客户端：**

if( xhr.responseText === 'succ'){ }

5.使用XHR接收响应消息——text/html

**服务器端：**

header('Content-Type: text/html');

echo '<li>xx</li><li>yy</li> '; //服务器返回的不是完整的HTML文档，只是一段HTML片段

**客户端：**

ul.innerHTML = xhr.responseText ;

效果演示：搜索建议 —— 面试题

实现步骤：

(1)编写SQL：创建数据库jd，产品表product，插入若干数据

(2)编写PHP：product\_search.php，接收客户端提交的搜索关键字kw，执行模糊查询，把包含搜索关键字的记录全部查询出来，以HTML片段形式返回，形如： <li>Java从入门到精通</li><li>JS从入门到精通</li>....

(3)编写HTML：product\_search.html，输入框中当键盘弹起时(onkeyup)，发起异步请求，询问服务器搜索结果

|  |
| --- |
| 扩展SQL知识点：模糊查询  SELECT \* FROM product WHERE pname='入门';  SELECT \* FROM product WHERE pname LIKE '%入门%';  SELECT \* FROM product WHERE pname LIKE '%入门%' OR pname LIKE '%指南%';  %匹配任意多个任意字符 |

6.使用XHR接收响应消息——application/javascript

**服务器端：**

header('Content-Type: application/javascript');

echo 'alert(111);alert(222);f1(9999)'; //JS在PHP看来就是字符串

**客户端：**

var str = xhr.responseText ; //客户端接收到字符串

eval( str ); //JS中执行字符串表示的代码

练习：编写一个PHP，向客户端输出一段JS字符串；编写一个HTML，点击一个按钮，发起异步请求，读取服务器返回的JS字符串并执行

演示：实现了国际化( i18n )的欢迎消息

过程：

(1)编写PHP：从请求头部读取客户端的首选语言，根据此值返回不同形式的欢迎消息以及显示方式。

(2)编写HTML，点击一个按钮，异步获取欢迎消息并执行出来。

7.服务器如何向客户端返回批量的复合数据

服务器端：

$data = [

['name'=>'xx', 'pic'=>'xx', 'price'=>xxx],

['name'=>'xx', 'pic'=>'xx', 'price'=>xxx],

....

['name'=>'xx', 'pic'=>'xx', 'price'=>xxx]

];

//HTTP协议要么传图片/音视频，要么传字符数据，不能传数组！

PHP必须把批量数据转换为某种格式的字符串，再发送给客户端

**可用于描述批量复合数据的字符串格式：**

(1)自定义字符串格式

"金士顿32G#56#img/1.jpg@闪迪32G#55#img/2.jpg@三星64G#78#img/3.jpg"

问题：不够灵活，不易识别，容易出错

(2)HTML片段字符串格式

"<div><img src="img/1.jpg"><h3>金士顿32G</h3><p>56</p></div>

<div><img src="img/2.jpg"><h3>闪迪 32G</h3><p>56</p></div>

<div><img src="img/3.jpg"><h3>三星32G</h3><p>56</p></div>"

问题：数据格式比较复杂、数据显示已经固定，不够灵活，前后端工作混杂在一起

(3)XML字符串格式 —— 了解

(4)JSON字符串格式 —— 重点掌握

8.XML字符串格式语法要求

eXtensible Markup Language，可扩展的标记语言，是一种标签语言，但没有预定义任意一个标签——所有的标签都是自定义的

(1)根标签有且只能有一个

(2)标签严格区分大小写

(3)开始标签和结束标签必须一样

(4)标签可以声明任意的属性，属性必须有值，值必须用'或"括起来

<img>

<br>

<hr>

<li>AAA

<li>BBB

<p>AAA

<p>BBB

总结：XML和HTML的区别？

1)XML标签可以任意指定，但HTML标签是固定的；

2)XML语法非常严格，HTML语法非常松散；

3)XML用于描述和存储数据——类似于数据库；HTML用于展现网页的结构，只用于浏览器；

8.使用XHR接收响应消息——application/xml

**服务器端：**

header('Content-Type: application/xml');

$data = [ [ ],[ ],[ ] ];

echo "<productList><p></p></productList>";

**客户端：**

xhr.responseXML ;

|  |
| --- |
| DOM标准分为三部分：  **核心DOM：**处理任意的树型文档——HTML文档和XML文档  **HTMLDOM**：处理HTML文档  **XMLDOM：** |
| 客户端JS不要使用xhr.responseText来处理XML数据——太麻烦。  使用xhr.responseXML来处理——它是一个DOM树的根元素：document |

案例演示：异步请求表格中更多的数据

(1)编写SQL，创建数据库-jd，表-product

(2)编写PHP，查询出更多的记录，保存在二维数组中，以XML格式输出给客户端

(3)编写HTML，点击按钮，异步请求更多的记录行

练习内容：

功能要求：

(1)编写SQL，数据库名: tarena，员工信息表emp(eid,ename,pic, salary,deptId)，保存多个员工的信息；

(2)编写PHP，emp\_select.php，根据客户端提交的部门编号(deptId)，返回该部门下的员工信息；

(3)编写HTML，emp\_select.html，下拉框中选择不同的部门，则下方的员工列表显示出该部门的员工信息；

(4)点击“删除”，发起异步请求（emp\_delete.php）删除该员工信息，服务器返回删除成功信息后，前端页面将表格中的当前行删除（从tbody中删除tr）。



day06

复习：

AJAX：是客户端技术，目标：(1)在无刷新无提交的情况下，页面内容的局部更新 (2)实现前端后端工作分工

day01:使用XHR发起两种请求：

var xhr = new XMLHttpRequest()

xhr.onreadystatechange = function(){}

xhr.open('GET','x.php?k=v', true)

xhr.send(null);

-------------------------------------

var xhr = new XMLHttpRequest()

xhr.onreadystatechange = function(){}

xhr.open('POST','x.php', true)

xhr.setRequestHeader('Content-Type','application/x-www-form-urlencoded');

xhr.send('k=v');

day02:使用XHR接收五种响应：

1)普通文本

header('Content-Type: text/plain');

echo 'succ';

--------------------

if(xhr.responseText==='succ'){}

2)html片段

header('Content-Type: text/html');

echo '<li></li><li></li>';

--------------------

ul.innerHTML = xhr.responseText

3)javascript语句

header('Content-Type: application/javascript');

echo 'alert(111);alert(222)';

--------------------

eval(xhr.responseText)

4)XML格式字符串

header('Content-Type: application/xml');

echo '<productList><product></product></productList>';

--------------------

var document = xhr.responseXML

今日目标：

day03:使用jQuery封装的AJAX操作函数：

练习： 批量复合数据

**1)编写SQL**：数据库-mymovie，表-movie( mid, title, pic, showDate, typeId(1-喜剧片，2-动作片，3-战争片) )，插入8部电影记录

**2)编写PHP**，movie\_select.php，接收客户端提交的typeId,返回该类型下的影片信息，以XML格式

<movieList>

<movie mid="">

<title>XXXX</title>

</movie>

</moveList>

**3)编写HTML**，movie\_select.html，显示下拉列表，三种影片类型，用户选项一改变，异步请求该类别有哪些影片，显示在下面的TABLE中

1.描述批量的复合数据的字符串格式——JSON字符串

官网：<http://json.org/json-zh.html>

JSON(JavaScript Object Notation) 是一种轻量级的字符串数据交换格式。 易于人阅读和编写。同时也易于机器解析和生成。语法借鉴了ECMAScript的一部分内容。JSON格式的数据可以被常见的编程语言所支持。

JSON作用等同于XML作用的一部分——描述批量复合数据；

XML属于重量级数据格式；JSON属于轻量级数据格式，更加简单容易使用。Web应用中，JSON可以完全取代XML格式。

JSON字符串语法：

(1)JSON字符串只能有一个根元素，要么是一个数组： [ ]，要么是一个对象：{ }

(2)JSON数组必须是多个并列的值:

[ 值1, 值2, 值3 ]

(3)JSON对象中必须是键值对集合

{ "k1":值1, "k2":值2 }

(4)所有的键必须用双引号！值是字符串的话用双引号！

(5)值的类型可以是number、string、boolean、null；没有其他类型了。

2.使用XHR接收响应数据之五——json

**服务器端：**

header('Content-Type: application/json; charset=UTF-8');

$list = ...; //PHP数组数据

$str = json\_encode($list); //PHP中把数组编码为JSON字符串

echo $str;

**客户端：**

var arr = JSON.parse( xhr.responseText ); //JS中把JSON字符串解析为JS对象或数组

3.jQuery中对AJAX操作的封装函数

jQuery提供了6个简化AJAX操作的函数，每个都可以代替元素AJAX中的四步代码！

(1)$('xxx').load() jQuery对象函数

(2)$.get() jQuery全局函数

(3)$.post() jQuery全局函数

(4)$.getScript() jQuery全局函数

(5)$.getJSON() jQuery全局函数

(6)$.ajax() jQuery全局函数

4. jQuery中对AJAX操作的封装函数之一——load

**用法：**

$('选择器').load( 'x.php' ,[data], [fn] )

**作用：**

使用XHR发起异步的请求，获取服务器端返回的html片段，把xhr.responseText设置为当前选定元素的innerHTML。响应成功完成后，会自动调用第三个回调函数。

不足：*服务器返回必须是HTML片段！响应内容会替换掉当前选定元素中已有的内容！*

演示：异步加载全站的页头和页尾

**服务器端：**

创建header.php，只包含HTML片段：页头内容

创建footer.php，只包含HTML片段：页尾内容

**客户端：**

创建index.html:

<body>

<div id="header">空</div>

<div id="main">首页中的主体内容</div>

<div id="footer">空</div>

</body>

待整个页面加载完成后，异步请求页头，放在#header中；再异步请求页尾，放在#footer中

5. jQuery中对AJAX操作的封装函数之二——$.get

**用法：**

$.get( 'x.php' ,[data], function(txt, msg, xhr){ } )

$.get( 'x.php' ,'uname=tom&age=20', function(txt, msg, xhr){ } )

$.get( 'x.php' ,{uname:'tom', age:20} , function(txt, msg, xhr){ } )

**作用：**

创建XHR发起异步的GET请求，第二个参数是可选的请求数据(键值对形式或者对象形式)，获取服务器端的响应消息，调用第三个回调函数（相当于之前的doResponse），在此回调函数中对响应主体进行处理。

练习：页面加载完成后，异步请求表格中的批量数据

(1)编写SQL：数据库-mymovie，表-movie

(2)编写PHP：movie\_select\_all.php，以JSON格式向客户端返回所有的电影记录 '[{},{},{}]'

(3)编写HTML：movie\_select\_all.html，当页面加载完后，异步请求所有的电影记录，拼接在TABLE中 $.get('x.php', doResponse)

6. jQuery中对AJAX操作的封装函数之三——$.post

**用法：**

$.post( 'x.php' ,data, function(txt, msg, xhr){ } )

$.post( 'x.php' ,'uname=tom&age=20', function(txt, msg, xhr){ } )

$.post( 'x.php' ,{uname:'tom', age:20} , function(txt, msg, xhr){ } )

**作用：**

创建XHR发起异步的POST请求，第二个参数是必需的请求数据(键值对形式或者对象形式)，获取服务器端的响应消息，调用第三个回调函数（相当于之前的doResponse），在此回调函数中对响应主体进行处理。

练习：电影数据加载完成后，当用户点击某个电影的“删除”，异步提交要删除的影片编号，实现删除功能

步骤：

(1)编写SQL

(2)编写PHP：movie\_delete.php，接收客户端提交的mid，执行删除，从数据中删除该电影，返回'{"msg":"succ","affectedRows":1}' 或 '{"msg":"err", "sql":"DELETE...."}'

**提示：PHP中的关联数组会被json\_encode()编码为JSON对象**

(3)修改影片列表页，点击某个“删除”，异步提交要删除的影片编号，实现影片删除

**提示：为DOM树上后添加的元素绑定事件监听必须用“事件代理**”

7. jQuery中对AJAX操作的封装函数之四——$.getScript

**用法：**

$.getScript( 'x.php' ,[data], [fn])

**作用：**

创建XHR发起异步的GET请求，第二个参数是可选的请求数据(键值对形式或者对象形式)，第三个参数，回调函数可选的，表示响应成功之后的回调。要求服务器端返回的必须是application/javascript类型的响应，该方法会*自动调用eval(xhr.responseText)进行*执行。

演示：页面加载完成后，异步请求服务器端的实现了国际化的欢迎消息(i18n)

|  |
| --- |
| $.getScript()不论响应消息内容类型声明什么，都会调用eval(xhr.responseText) |
| $.get()只有当响应消息内容类型声明为application/javascript，才会调用eval(xhr.responseText) |

8. jQuery中对AJAX操作的封装函数之五——$.getJSON

**用法：**

$.getJSON( 'x.php' ,[data],function(result, msg, xhr){ })

**作用：**

创建XHR发起异步的GET请求。要求服务器端返回的必须是application/json类型的响应，该方法会*自动调用JSON.parse(xhr.responseText)进行*解析，再调用第三个参数——响应成功的回调函数处理响应结果。

|  |
| --- |
| $.getJSON()不论响应消息内容类型声明什么，都会调用JSON.parse(xhr.responseText) |
| $.get()只有当响应消息内容类型声明为application/json，才会调用JSON.parse(xhr.responseText) |

|  |
| --- |
| 面试题：使用jQuery的AJAX函数时，如何处理失败的响应消息？ |

9. jQuery中对AJAX操作的封装函数之五——$.ajax —— 重点！

**用法：**

$.ajax( {

type: 'GET', //POST/PUT/DELETE/HEAD

url: 'x.php',

data: 'k=v'或{k:v} //要提交的请求数据

beforeSend: fn, //请求发送之前的回调函数

success: fn, //响应成功的回调函数

error: fn, //响应失败的回调函数

complete: fn //响应完成的回调函数-不论成败

} )

|  |
| --- |
| var xhr = new XMLHttpRequest()  xhr.onreadystatechange = function(){  if(xhr.readyState===4){  if(xhr.status===200){  success();  }else {  error();  }  complete();  }  }  xhr.open('GET','x.php?k=v', true)  beforeSend();  xhr.send(null); |
| 响应成功回调顺序： beforeSend => success => complete |
| 响应失败回调顺序： beforeSend => error => complete |

**作用：**

此函数是jQuery中万能的AJAX调用函数！前面5个方法都是该方法的简化版本。

创建XHR发起异步的GET请求。要求服务器端返回的必须是application/json类型的响应，该方法会*自动调用JSON.parse(xhr.responseText)进行*解析，再调用第三个参数——响应成功的回调函数处理响应结果。

课后练习：仿beijing.huimaiche.com中的车辆展示

lt8 lt15 lt30 suv



汽车详情：



(1)编写SQL，数据库-huimaiche，表-car（cid,cname,price,bigPic,thumbPic, count, type）

(2)编写PHP，创建car\_select\_bytype.php，根据客户端提交的车辆类别type，返回该类别下所有汽车，以JSON格式

(3)编写HTML，创建car\_select.html，用户鼠标悬停在不同的车辆类别上，异步请求该类别下所有的汽车

(4)编写PHP，创建car\_select\_byid.php，根据客户端提交的cid，向客户端输出该车辆的所有信息，以JSON格式

(5)修改car\_select.html，当用户点击某个车辆的图片时，在下方显示出该车辆的全部信息

day07

复习：

见思维导图



今日目标

(1)PPT小知识拾漏

(2)面试题：跨域请求——掌握

(3)AJAX阶段项目——重点

1.补充小知识点：JSON和JS对象间的转换

JSON字符串解析为JS对象：

var str = '{"ename":"tom", "age":20}';

var obj = JSON.parse(str);

obj.ename

JS对象编码为JSON字符串：

var obj = {ename: 'Mary', age: 18};

var str = JSON.stringify(obj);

str

2.补充小知识点：JSON和PHP数组的转换

JSON字符串解析为PHP数组：

$str = '{"ename":"tom", "age":20}';

$list = json\_decode(str);

PHP数组编码为JSON字符串：

$list = [ ];

$str = json\_encode($list);

3.面试题：跨域请求

Cross Domain Request：从某个域名下的资源发起请求，获取另一个域名下的资源。只要两个域名的协议名、域名、端口之中有一项不同，就是跨域。即使是从127.0.0.1请求localhost下的资源，也算跨域！



**浏览器哪些情形下允许跨域请求：**

IMG、LINK、SCRIPT、IFRAME等标签都可以跨域请求

**浏览器哪些情形下禁止跨域请求：**

XHR请求 —— 浏览器处于安全考虑，禁止XHR的跨域访问，不是服务器的问题。

如何解决浏览器XHR禁止跨域的问题：—— baidu

(1)方案1：

(2)方案2：

(3)方案3：

(4)方案4：

(5)方案5：

(6)方案6：

(7)方案7：在服务器响应消息中声明允许跨域源

header('Access-Control-Allow-Origin: [http://127.0.0.1](http://127.0.0.1/)');

(8)方案8：JSONP手段 —— 非常巧妙！

JSON是一种字符串数据格式。—— 猪肉/鸡肉

JSONP: JSON with Padding，填充式JSON，与JSON完全不一样！是一种手段，一种解决浏览器禁止XHR跨域访问的方案。—— 红烧/清炖

JSONP基本原理：

动态添加一个SCRIPT标签，请求PHP页面返回的动态数据——必须填充为JS函数的调用，SCRIPT标签会自动执行该函数——**在客户端执行**。JSONP方案中服务器端返回的数据形如：

**doResponse(** {"ename":"Tom", "age":20} **)**

其中的数据是服务器的动态数据，而JS函数是定义在客户端的。

day08

AJAX阶段项目——京东购物车

1.Web应用中的分页查询——前后端都是难点&重点



当数据库中的记录条数比较多，一次无法全部显示，需要使用分页显示。

**分页查询中前端的请求消息形如：**

GET /data/product\_select.php?pageNum=3

**分页查询中后台的响应消息形如：**

{

recordCount: 36, //符合条件的总记录数

pageSize: 8, //页面大小，每页的记录数

pageCount: 5, //总的页数

pageNum: 3, //当前的页号

data: [{},{}...{}] //当前页中的数据

}

服务器端如何获得上述数据：

(1)SQL中如何读取满足条件的总记录行数

SELECT COUNT(\*) FROM product WHERE ...

(2)如何计算总页数

ceil(recordCount / pageSize)

(3)如何设定当前显示的页号——页号必须是整数

intval( $\_REQUEST['pageNum'] )

(4)如何读取指定“页”中的记录

SELECT \* FROM proudct LIMIT (pageNum-1)\*pageSize, pageSize;

1 2 3

1 2 3 4

1 2 3 4 5

2 3 4 5

3 4 5

|  |
| --- |
| 扩展SQL知识——分组函数  SELECT MAX(price) FROM jd\_product;  SELECT MIN(price) FROM jd\_product;  SELECT SUM(price) FROM jd\_product;  SELECT COUNT(price) FROM jd\_product;  SELECT AVG(price) FROM jd\_product; |
| 扩展SQL知识——分页函数  SELECT \* FROM jd\_product [WHERE ...] LIMIT start, count;  start: 指定从哪一行开始读取，从0开始....  count: 一次最多读取多少行  第1页： ... LIMIT 0, 8;  第2页： ... LIMIT 8, 8;  第3页： ... LIMIT 16, 8;  第4页： ... LIMIT 24, 8;  第5页： ... LIMIT 32, 8;  第pageNum页： ... LIMIT (pageNum-1)\*pageSize, pageSize; |

SELECT \* FROM jd\_product LIMIT 0,8;

2.扩展SQL语句——跨表查询

一个结果集中的数据来自于多个表：

CREATE TABLE dept ( did INT , dname VARCHAR(32) );

INSERT INTO dept VALUES

(10, '市场部'),

(20,'研发部');

CREATE TABLE emp ( eid INT , ename VARCHAR(32), deptId INT );

INSERT INTO emp VALUES

(1, 'Tom', 10),

(2,'Mary', 10),

(3, 'John', 20);

请查询出员工姓名以及其所在部门的名称：

SELECT ename,dname FROM emp,dept;

 

跨表查询不注意就可能产生“笛卡尔积”问题！

SELECT ename,deptId,dname,did

FROM emp,dept

WHERE emp.deptId = dept.did; //没有指定此关联条件就会产生笛卡尔积问题

3.扩展SQL知识——对查询结果排序

SELECT \* FROM jd\_product

[WHERE ....]

ORDER BY price;

[LIMIT ....]

-----------------------------------

SELECT \* FROM jd\_product

[WHERE ....]

ORDER BY price DESC;

[LIMIT ....]

课下可选功能：

(1)产品列表中按特定的列排序

(2)添加购物车后，在页面上方显示出购物车中商品的数量

(3)添加购物车成功后，客户端的商品添加动画：“飞入”购物车中

(4)实现购物车条目的“批量删除”