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。

3.学习HTTP协议的两个目的——重要！

(1)对网站访问进行优化设计

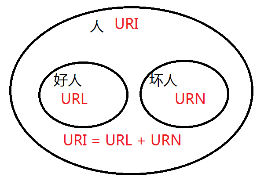
(2)调试AJAX请求中的“看不见摸不着”的错误

4.URL

URL：Unified Resource Locator：统一的资源定位符

URN：Unified Resource Naming：统一的资源命名符

URI：Unified Resource Identifier：统一的资源识别符



URL:

<a href="http://baidu.com/1.php">

<a href="ftp://baidu.com/2.mp3">

<a href="img/3.jpg">

URN:

<a href="mailto: admin@tmooc.cn">

<a href="tel: 1350123457">

<a href="javascript: void(0)">

URL地址最完整的形式：

**<scheme>://<user>:<pwd>@<host>:<port>/<path>;<params>?<query>#<frag>**

5.HTTP协议的历史

IETF制定了HTTP协议的详细内容——规定了客户端浏览器和Web服务器通信所必需的内容格式。

RFC: Request For Comment，意见征求稿

HTTP/1.0： 1996年 RFC1945

HTTP/1.1： 1999年 RFC2616

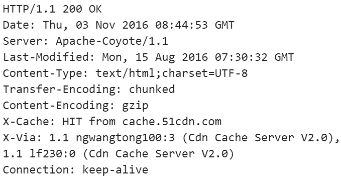
|  |
| --- |
| 三次握手 -> 发请求 -> 四次挥手  面试题：HTTP/1.1相比1.0有哪些改进？  (1)支持持久连接 不是永久连接  Connection: keep-alive  (2)支持虚拟主机  Host: www.baidu.com  (3)支持代理连接  Poxy: xxx |

6.HTTP协议详解

HTTP请求(Request)消息： 客户端发送给服务器的消息



HTTP响应(Response)消息： 服务器返回给客户端的消息



|  |
| --- |
| 面试题：  **(1)浏览器什么情况下发起GET请求？POST请求？**  **GET请求**：地址栏输入URL回车、超链接跳转、JS跳转、GET方式提交表单、AJAX-GET请求  **POST请求**：POST方式提交表单、AJAX-POST请求  **(2)GET和POST请求的区别？**  GET把请求数据追加在请求URI后面，以?开头;——由于浏览器和服务器对URI总长度有限制，一般请求数据不要超过几十个汉字。GET请求没有请求主体。GET的语义表“获取”。  POST把请求数据放在请求主体中，不以?开头;——请求主体的长度没有限制，可以传输大量数据，尤其是文件上传。必需请求主体。POST的语义表“传递/提交”。  **(3)GET和HEAD请求的区别？**  HEAD请求不会获得响应主体  **(4)你在什么项目中使用到了PUT和DELETE请求吗？**  RESTful项目 |

课后练习：

HTML中base标签的含义是什么？

**base> 标签为页面上的所有链接规定默认地址或默认目标。**

**通常情况下，浏览器会从当前文档的 URL 中提取相应的元素来填写相对 URL 中的空白。**

**使用 <base> 标签可以改变这一点。浏览器随后将不再使用当前文档的 URL，而使用指定的基本 URL 来解析所有的相对 URL。这其中包括 <a>、<img>、<link>、<form> 标签中的 URL。-+**

(2)阶段小项目：留言板

功能：

1)发表留言： 用户在一个表单中提交自己的姓名、电话、以及留言内容，点击提交按钮即可发表。

2)浏览所有留言： 可以查看到编号、姓名、电话、内容、发布时间。

3)删除某个留言： 点击某个留言右上角的X删除该留言。

创建SQL->创建PHP->创建HTML

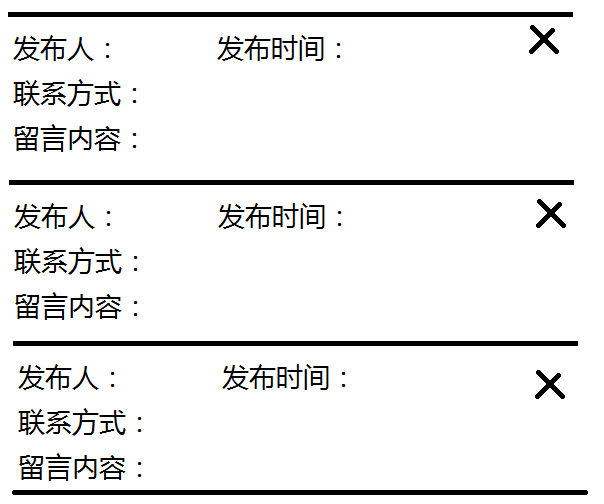
需要的文件：

1)编写一个tarena.sql，创建一个留言表 msg，包含留言编号mid、发布人姓名uname、联系电话phone、发布时间pubTime、留言内容content

2)发表留言： msg\_add.php msg\_add.html

3)浏览所有留言： msg\_select.php

4)删除留言： msg\_delete.php



<a href="delete.php?mid=1">&times;</a>

<a href="delete.php?mid=2">&times;</a>

<a href="delete.php?mid=3">&times;</a>

进阶要求：点击“删除”先弹出一个确认框，如果用户选择“是”，才真正跳转到删除页面，否则不跳转。