复习：

AJAX阶段项目

项目目标： 强化 SQL + PHP + AJAX 的使用

项目功能：

(1)弹出登录对话框，异步登录验证

(2)异步加载页头和页尾

(3)异步加载商品加载

(4)实现异步分页 —— 重点&难点

(5)异步添加商品到购物车

(6)异步修改购物车内容 —— 重点&难点

今日目标

(1)完成商品展示+分页效果 —— 重点&难点

(2)呈现购物车内容 —— 重点&难点

1.实现步骤：

(1)创建jd.sql，设计出必要的表结构

(2)编写产品列表页面，在jd.js中异步加载页头和页尾

(3)修改产品列表页面，显示一个登录弹出框，实现异步登录

(4)修改产品列表主体，异步加载前8条记录

data/1\_product\_select.php

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

(5)修改分页条，实现异步分页加载

data/1\_product\_select.php

(6)点击某个商品下的“添加到购物车”则在购物车详情表中添加新记录 3\_cart\_add.php?pid=?&uname=?

SQL1：根据用户名查询出用户编号

SQL2：查询该用户编号是否已有购物车

SQL3：若用户无购物车，则创建一个购物车

SQL4：查询购物车详情中是否已有指定商品编号

SQL5：若商品编号在购物车中不存在，则创建购买记录

SQL6：若商品编号在购物车中已存在，则更新购买数量

(7)点击查看“我的购物车”，跳转到购物车详情页面，使用cart.js

<a href="#">查看我的购物车</a>.onclick = function(){

location.href="shoppingcart.html?loginName="+loginName;

}

(8)在cart.js中，异步加载购物车详情页面的页头和页尾

使用location.search读取前一个页面传递的查询字符串——当前登录用户名

(9)在cart.js中，异步加载当前登录用户的购物车详情

4\_cart\_detail.php?uname=?

SQL1：根据登录用户名查询用户编号

SQL2：根据用户编号查询对应的购物车编号

SQL3：根据购物车编号到购物车详情表和产品表跨表查询购物车中的内容——小心防止出现笛卡尔积。

$.getJSON异步请求PHP页面，获取购物车数据，响应完成后，修改购物车内容

(10)用户修改完商品的购买数量，异步提交给服务器，修改数据库中购物车详情中的内容 did:购物车详情表中记录行的编号

5\_cart\_update.php?did=x&count=x

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

(11)购物车详情中，点击某个商品后的“删除”，则异步请求服务器页面，删除数据库中的对应记录

6\_cart\_detail\_delete.php?did=x

2.补充SQL知识点

(1)查询结果的排序

SELECT \* FROM jd\_product; #默认按插入顺序呈现

SELECT \* FROM jd\_product ORDER BY price; #升序

SELECT \* FROM jd\_product ORDER BY price DESC; #降序

(2)分页查询

SELECT \* FROM 表名 LIMIT start, count;

start：INT变量，指定从哪一行开始读取记录，第一行下标为0

count: INT变量，指定一次最多返回的记录行数

假设1页最多8条记录：

第1页：SELECT \* FROM jd\_product LIMIT 0, 8;

第2页：SELECT \* FROM jd\_product LIMIT 8, 8;

第3页：SELECT \* FROM jd\_product LIMIT 16, 8;

第4页：SELECT \* FROM jd\_product LIMIT 24, 8;

第5页：SELECT \* FROM jd\_product LIMIT 32, 8;

第n页：SELECT \* FROM jd\_product LIMIT (n-1)\*8, 8;

SELECT \* FROM jd\_product ORDER BY price DESC LIMIT 0,8;

(3)查询满足条件的记录条数

SELECT COUNT(pid) FROM jd\_product;

SELECT COUNT(\*) FROM jd\_product WHERE price<2000;

3.Web项目中基于JSON的分页查询——难点

基于JSON的分页查询中前端和后端传递的数据格式：

前端请求消息： **product.php?pno=1**

后端响应消息： **{recordCount:36, pageSize:8,...}**

{

recordCount: 36, //总的记录数

pageSize: 8, //页面大小，一页最多显示的记录数

pageCount: 5, //总页数

pno:1, //PageNumber，当前显示的页号

data: [{},{},{},{}] //当前页中的数据

}

start = (pno-1)\*pageSize;

count = pageSize;

SELECT \* FROM jd\_product LIMIT start, count;

1 2 3

1 2 3 4

1 2 3 4 5

2 3 4 5

3 4 5

4.SQL语句中的跨表查询

CREATE TABLE dept( did, dname );

10 研发部

20 市场部

CREATE TABLE emp( eid, ename, deptID);

1 Tom 10

2 Mary 10

3 John 20

SQL：查询出员工姓名和该用户对应的部门名称

SELECT ename,dname FROM dept,emp; //m\*n 笛卡尔积

SELECT ename,dname FROM dept,emp WHERE dept.did=emp.deptID;

结论：跨表查询必须有两个表的连接条件，否则就会得到笛卡尔积。

课下任务：

(1)继续完成购物车中商品的异步删除

(2)挑战性任务：产品列表页面，在“去购物车结算”超链接上显示当前购物车中产品的数量