

小白入门项目——设计论坛系统数据库

1. 系统概述

论坛又名 BBS，全称为 Bulletin Board System(电子公告板)或者 Bulletin Board Service (公告板服务)。它是 Internet 上的一种电子信息服务系统。它提供一块公共电子白板，每个用户都可以在上面书写，可发布信息或提出看法。

论坛是一种交互性强，内容丰富而及时的电子信息服务系统。用户在 BBS 站点上可以获得各种信息服务、发布信息、进行讨论、聊天等等。像日常生活中的黑板报一样，论坛按不同的主题分为许多版块，版面的设立依据是大多数拥护的要求和喜好，用户可以阅读别人关于某个主题的看法，也可以将自己的想法毫无保留地帖到论坛中。随着计算机网络技术的不断发展，BBS 论坛的功能越来越强大，目前 BBS 的主要功能有以下几点：

- (1) 供用户自我选择阅读若干感兴趣的专业组和讨论组内的信息。
- (2) 可随意检查是否有新消息发布并选择阅读。
- (3) 用户可在站点内发布消息或文章供他人查阅。
- (4) 用户可就站点内其他人的消息或文章进行评论。
- (5) 同一站点内的用户互通电子邮件，设定好友名单

2. 系统功能

论坛管理系统重要功能是管理论坛帖子的基本信息。通过本管理系统，可以提高论坛管理员的工作效率。本节将详细介绍本系统的功能。

论坛系统主要分为 5 个管理部分，包括用户管理、管理员管理、板块管理、主贴管理和回复贴管理。本系统的功能模块的详细介绍如下。

- (1) 用户管理模块：实现新增用户，查看和修改用户信息功能。
- (2) 管理员管理模块：实现新增管理员，查看、修改和删除管理员信息功能。
- (3) 板块管理模块：实现对管理员、对管理的模块和管理的评论赋权功能。
- (4) 主贴管理模块：实现对主贴的增加、查看、修改和删除功能。
- (5) 回复贴管理模块：实现有相关权限的管理员对回复贴的审核和删除功能。

3. 数据库设计和实现

本系统所有的表都放在 bss 数据库下。创建和选择 bss 数据库的 SQL 代码如下：

```
CREATE DATABASE bbs;
```

```
USE bbs;
```

在这个数据库下总共存放 5 张表，分别是 user、admin、section、topic 和 reply。

1. user 表

user 表中存储用户 ID、用户名、密码和用户 Email 地址等，所以 user 表设计了 10 个

字段。user 表每个字段的信息如表 1 所示。

表 1 user 表的内容

列名	数据类型	允许 NULL 值	说明
uID	INT	否	用户编号
userName	VARCHAR(20)	否	用户名称
userPassword	VARCHAR(20)	否	用户密码
userEmail	VARCHAR(20)	否	用户 Email
userBirthday	DATE	否	用户生日
userSex	BIT	否	用户性别
userClass	INT	否	用户等级
userStatement	VARCHAR(150)	否	用户个人说明
userRegDate	TIMESTAMP	否	用户注册时间
userPoint	INT	否	用户积分

根据表 1 的内容创建 user 表。创建 user 表的 SQL 语句如下：

```
CREATE TABLE user(  
    uID INT PRIMARY KEY UNIQUE NOT NULL,  
    userName VARCHAR(20) NOT NULL,  
    userPassword VARCHAR(20) NOT NULL,  
    sex varchar(10) NOT NULL,  
    userEmail VARCHAR(20) NOT NULL,  
    userBirthday DATE NOT NULL,  
    userSex BIT NOT NULL,  
    userClass INT NOT NULL,  
    userStatement VARCHAR(150) NOT NULL,  
    userRegDate TIMESTAMP NOT NULL,  
    userPoint INT NOT NULL  
);
```

2. admin 表

管理员信息表（admin）主要用来存放用户账号信息，如表 2 所示。

表 2 admin 表的内容

列名	数据类型	允许 NULL 值	说明
adminID	INT	否	管理员编号
adminName	VARCHAR(20)	否	管理员名称
adminPassword	VARCHAR(20)	否	管理员密码

根据表 2 的内容创建 admin 表。创建 admin 表的 SQL 语句如下：

```
CREATE TABLE admin(  
    adminID INT PRIMARY KEY UNIQUE NOT NULL,  
    adminName VARCHAR(20) NOT NULL,  
    adminPassword VARCHAR(20) NOT NULL  
);
```

3. section 表

板块信息表（section）主要用来存放板块信息，如表 3 所示。

表 3 section 板块信息表的内容

列名	数据类型	允许 NULL 值	说明
sID	INT	否	版块编号
sName	VARCHAR(20)	否	版块名称

sMasterID	INT	否	版主编号
sStatement	VARCHAR	否	版块说明
sClickCount	INT	否	版块点击次数
sTopicCount	INT	否	版块主题数

根据表 3 的内容创建 section 表。创建 section 表的 SQL 语句如下：

```
CREATE TABLE section (  
    sID INT PRIMARY KEY UNIQUE NOT NULL,  
    sName VARCHAR(20) NOT NULL,  
    sMasterID INT NOT NULL,  
    sStatement VARCHAR NOT NULL,  
    sClickCount INT NOT NULL,  
    sTopicCount INT NOT NULL  
);
```

4. topic 表

主贴信息表（topic）主要用来存放主贴信息，如图所示。

表 4 topic 主贴信息表的内容

列名	数据类型	允许 NULL 值	说明
tID	INT	否	主贴编号
sID	INT	否	主贴版块编号
uid	INT	否	主贴用户编号
tReplyCount	INT	否	主贴回复次数
tEmotion	VARCHAR	否	主贴表情
tTopic	VARCHAR	否	主贴标题
tContents	TEXT	否	主贴内容
tTime	TIMESTAMP	否	发帖时间
tClickCount	INT	否	主贴点击次数
tLastClickT	TIMESTAMP	否	主贴最后点击时间

根据表 4 的内容创建 topic 表。创建 topic 表的 SQL 语句如下：

```
CREATE TABLE topic (  
    tID INT PRIMARY KEY UNIQUE NOT NULL,  
    tSID INT NOT NULL,  
    tuid INT NOT NULL,  
    tReplyCount INT NOT NULL,  
    tEmotion VARCHAR NOT NULL,  
    tTopic VARCHAR NOT NULL,  
    tContents TEXT NOT NULL,  
    tTime TIMESTAMP NOT NULL,  
    tClickCount INT NOT NULL,  
    tLastClickT TIMESTAMP NOT NULL  
);
```

5. reply 表

回复贴信息表（reply）主要用来存放回复贴的信息。

表 5 reply 回复贴信息表的内容

列名	数据类型	允许 NULL 值	说明
rID	INT	否	回复编号
tID	INT	否	回复帖子编号
uID	INT	否	回复用户编号

rEmotion	CHAR	否	回帖表情
rTopic	VARCHAR (20)	否	回帖主题
rContents	TEXT	否	回帖内容
rTime	TIMESTAMP	否	回帖时间
rClickCount	INT	否	回帖点击次数

根据表 5 的内容创建 reply 表。创建 reply 表的 SQL 语句如下：

```
CREATE TABLE reply (
    rID INT PRIMARY KEY UNIQUE NOT NULL,
    rtID INT NOT NULL,
    ruID INT NOT NULL,
    rEmotion CHAR NOT NULL,
    rTopic VARCHAR (20) NOT NULL,
    rContents TEXT NOT NULL,
    rTime TIMESTAMP NOT NULL,
    rClickCount INT NOT NULL
);
```

创建完成后，可以使用 DESC 语句查看 reply 表的基本结构，也可以通过 SHOW CREATE TABLE 语句查看 reply 表的详细信息。

4. 设计索引

索引是创建在表上的，是对数据库中一列或者多列的值进行排序的一种结构。索引可以提高查询的速度。论坛系统需要查询论坛的信息，这就需要在某些特定字段上建立索引，以便提高查询速度。

1. 在 topic 表上建立索引

新闻发布系统中需要按照 tTopic 字段、tTime 字段和 tContents 字段查询新闻信息。在本书的前面的章节中介绍了几种创建索引的方法。本小结将使用 CREATE INDEX 语句和 ALTER TABLE 语句创建索引。

下面使用 CREATE INDEX 语句在 tTopic 字段上创建名为 index_topic_title 的索引。SQL 语句如下：

```
CREATE INDEX index_topic_title ON topic(tTopic);
```

然后，再使用 CREATE INDEX 语句在 tTime 字段上创建名为 index_topic_time 的索引。SQL 语句如下：

```
CREATE INDEX index_topic_date ON topic(tTime);
```

最后，再使用 ALTER TABLE 语句在 tContents 字段上创建名为 index_topic_contents 的索引。SQL 语句如下：

```
ALTER TABLE topic ADD INDEX index_new_contents (contents);
```

2. 在 section 表上建立索引

论坛系统中需要通过版块名称查询该版块下的帖子信息，因此需要在这个字段上创建索引。创建索引的语句如下：

```
CREATE INDEX index_section_name ON section (sName);
```

代码执行完成后，读者可以使用 SHOW CREATE TABLE 语句查看 section 表的详细信息。

3. 在 reply 表上建立索引

论坛系统需要通过 rTime 字段、rTopic 字段和 rID 字段查询回复帖子的内容。因此可以在这 3 个字段上创建索引。创建索引的语句如下：

```
CREATE INDEX index_reply_rtime ON comment(rTime);
```

```
CREATE INDEX index_reply_rtopic ON comment (rTopic);  
CREATE INDEX index_reply_rid ON comment (tID);
```

代码执行完成后，读者可以通过 SHOW CREATE TABLE 语句查看 reply 表的结构。

5. 设计视图

在论坛系统中，如果直接查询 section 表，显示信息时会显示版块编号和版块名称等信息。这种显示不直观显示主贴的标题和发布时间，为了以后查询方面，可以建立一个视图 topic_view。这个视图显示版块的编号、版块的名称、同一版块下主贴的标题、主贴的内容和主贴的发布时间。创建视图 topic_view 的 SQL 代码如下：

```
CREATE VIEW topic_view  
AS SELECT s.ID,s.Name,t.tTopic,t.tContents,t.tTime  
FROM section s,topic t  
WHERE section.sID=topic.sID;
```

6. 设计触发器

1. 设计 INSERT 触发器

如果向 section 表插入记录，说明版块的主题数目要相应的增加。这可以通过触发器来完成。在 section 表上创建名为 section_count 的触发器，其 SQL 语句如下：

```
DELIMITER &&  
CREATE TRIGGER section_count AFTER UPDATE  
ON section FOR EACH ROW  
BEGIN  
    UPDATE section SET sTopicCount=sTopicCount+1  
    WHERE sID=NEW.sID;  
END  
&&  
DELIMITER ;
```

其中 NEW.sID 表示 section 表中增加的记录 sID 值。

2. 设计 UPDATE 触发器

在设计表时，user 表和 reply 表的 uID 字段的值是一样的。如果 user 表中的 uID 字段的值更新了，那么 reply 表中的 uID 字段的值也必须同时更新。这可以通过一个 UPDATE 触发器来实现。创建 UPDATE 触发器 update_userID 的 SQL 代码如下：

```
DELIMITER &&  
CREATE TRIGGER update_userID AFTER UPDATE  
ON user FOR EACH ROW  
BEGIN  
    UPDATE reply SET uID=NEW.uID  
END  
&&  
DELIMITER ;
```

其中 NEW.uID 表示 user 表中更新的记录的 uID 值。

3. 设计 DELETE 触发器

如果从 user 表中删除一个用户的信息，那么这个用户在 topic 表中的信息也必须同时删除。这也可以通过触发器来实现。在 user 表上创建 delete_user 触发器，只要执行 DELETE 操作，那么就删除 topic 表中相应的记录。创建 delete_user 触发器的 SQL 语句如下：

```

DELIMITER &&
CREATE TRIGGER delete_user AFTER DELETE
    ON user FOR EACH ROW
BEGIN
    DELETE FROM top WHERE uID=OLD.uID
END
&&
DELIMITER ;

```

其中，OLD.uID 表示新删除的记录的 uID 值。

小白提高项目——开发新闻发布系统

新闻发布系统是动态网站建设中最常见的系统，几乎每一个网站都有新闻发布系统，尤其是政府部门、教育系统或企业网站。新闻发布系统的作用就是在网上发布新闻，通过对新闻的不断更新，让用户及时了解行业信息、企业状况。本章就以一个新闻发布系统为例，来介绍 **MySQL** 在开发中的应用技能。

1. 系统功能描述

该系统功能主要包括用户登录及验证、新闻管理、发布新闻、删除新闻、修改新闻等功能。

整个项目以登录界面为起始，在用户输入用户名和密码后，系统通过查询数据库验证该用户是否存在。

新闻发布系统

标题	内容	发布时间	操作
"国考"今起开始报名	2018年国家公务员考试今起开始报名，据悉，本次招录共有120多个中央机关及其直属机构和参照公务员法管理的单位参加，计划招录2.8万余人，创下历年新高。	2017-10-30 16:59:00	增加 删除 修改
漂流5个月获救	据外媒报道称，美国两名女子在5个月前从夏威夷出发，准备航行到位于南太平洋的大溪地，但途中由于恶劣天气，她们失去了她们的引擎，被迫漂流在茫茫大海中。	2017-10-30 17:02:00	增加 删除 修改

2. 系统功能分析

一个简单的新闻发布系统，包括用户登录及验证、新闻管理、发布新闻、删除新闻、修改新闻等功能。。本节就来学习新闻发布系统的功能以及实现方法。

整个系统的功能结构如下图所示：

整个项目包含 **4** 个功能：

(1) 用户登录及验证：在登录界面用户输入用户名和密码后，系统验证通过查询数据库验证是否存在该用户，如果验证成功显示信新闻管理界面，否则提示“无效的用户名和密码”，并返回登录界面。

(2) 发布新闻：用户登录系统后，可以查看新闻列表，然后在对应新闻操作，选择“增加”，系统进入发布新闻界面，在输入相应信息后，点击“发布新闻”，系统会向数据库插入一条记录，并提示成功，返回新闻管理界面。

(3) 删除新闻：在对应新闻操作，选择“删除”后，系统会向数据库删除此条记录，并

提示成功，返回新闻管理界面。

(4) 修改新闻：在对应新闻操作，选择“修改”后，系统进入修改新闻界面，在修改相应信息后，点击“确定修改”，系统会向数据库更新一条记录，并提示成功，返回新闻管理界面。

3. 数据流程和数据库

根据系统功能和数据库设计原则，设计数据库 **newsreport**。SQL 语法如下：

```
CREATE DATABASE IF NOT EXISTS `newsreport`;
```

根据系统功能和数据库设计原则，共设计 2 张表，分别是：管理员表 **admin** 和新闻表 **news**。

表 1 管理员表 **admin**

字段名	数据类型	字段说明
userid	char(15)	管理员编码
pssw	char(10)	密码

表 2 新闻表 **news**

字段名	数据类型	字段说明
id	int(11)	新闻编码,自增
title	varchar(50)	标题
content	varchar(500)	内容
reporttime	datetime	发布时间

创建管理员表 **admin**，SQL 语句如下：

```
CREATE TABLE IF NOT EXISTS `admin` (  
  `userid` char(15) DEFAULT NULL,  
  `pssw` char(10) NOT NULL  
);
```

插入演示数据，SQL 语句如下：

```
INSERT INTO `admin` (`userid`, `pssw`) VALUES  
('test', '123');
```

创建会员表 **news**，SQL 语句如下：

```
CREATE TABLE IF NOT EXISTS `news` (  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  `title` varchar(50) DEFAULT NULL,  
  `content` varchar(500) DEFAULT NULL,  
  `reporttime` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',  
  PRIMARY KEY (`id`)  
);
```

插入演示数据，SQL 语句如下：

```
INSERT INTO `news` (`id`, `title`, `content`, `reporttime`) VALUES
```

(1, '“国考”今起开始报名', '2018 年国家公务员考试今起开始报名，据悉，本次招录共有 120 多个中央机关及其直属机构和参照公务员法管理的单位参加，计划招录 2.8 万余人，创下历年新高。', '2017-10-30 16:59:00'),

(2, '漂流 5 个月获救', '据外媒报道称，美国两名女子在 5 个月前从夏威夷出发，准备航行到位于

南太平洋的大溪地，但途中由于恶劣天气，她们失去了她们的引擎，被迫漂流在茫茫大海中。', '2017-10-30 17:02:00');

4. 代码的具体实现

该案例的代码清单包含 9 个 **php** 文件，实现了新闻发布系统的用户登录及验证、查看新闻列表、发布新闻、删除新闻、修改新闻等主要功能。

网站中文件的含义和代码如下：

1. index.php

文件是该案例的 **web** 访问入口，是用户的登录界面。具体代码如下：

```
<html>
<head>
<title>登录
</title>
</head>

<body>
<h1 align="center">新闻发布系统</h1>
<table width="100%" style="text-align:center">
<tr>
<form action="login.php" method="post">
<td width="60%" class="sub1">
<p class="sub">账号： <input type="text" name="userid" align="center" class="txttop"></p>
<p class="sub">密 码： <input type="password" name="pssw" align="center" maxlength="4"
class="txtbot"></p>
<button name="button" class="button" type="submit">登录</button>
</form>
</td>
</tr>
</table>
</body>
</html>
```

2. conn.php

该文件为数据库连接页面，代码如下：

```
<?php
// 创建数据库连接
$con = mysqli_connect("localhost", "root", "") or die("无法连接到数据库");
mysqli_select_db($con, "newsreport") or die(mysqli_error($con));
mysqli_query($con, 'set NAMES utf8');
?>
```

3. log.php

该文件是对用户登录进行验证。代码如下：

```
<html>
<head>
```



```

<title>
</title>
<link rel="stylesheet" type="text/css" href="css/main.css">
</head>
<body><h1 align="center">新闻发布系统</h1></body>
<p align="center">
<?php
//连接数据库
require_once("conn.php");
//账号
$userid=$_POST['userid'];
//密码
$pssw=$_POST['pssw'];
//查询数据库
$qry=mysqli_query($con,"SELECT * FROM admin WHERE userid='$userid'");
$row=mysqli_fetch_array($qry,MYSQLI_ASSOC);
//验证用户
if($userid==$row['userid'] && $pssw==$row['pssw']&&$userid!=null&&$pssw!=null)
{
    session_start();
    $_SESSION["login"] = $userid;
    header("Location: newslst.php");
}
else{
    echo "无效的账号或密码!";
    header('refresh:1; url= index.php');
}
//}
?>
</p>
</body>
</html>

```

4. list.php

该文件为系统的主界面。具体代码如下：

```

<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<style type="text/css">
table.gridtable {
    font-family: verdana,arial,sans-serif;
    font-size:11px;
    color:#333333;
    border-width: 1px;
    border-color: #666666;

```

```

        border-collapse: collapse;
    }
    table.gridtable th {
        border-width: 1px;
        padding: 8px;
        border-style: solid;
        border-color: #666666;
        background-color: #dedede;
    }
    table.gridtable td {
        border-width: 1px;
        padding: 8px;
        border-style: solid;
        border-color: #666666;
        background-color: #ffffff;
    }
</style>
<title>新闻列表</title>
</head>
<body>
<h1 align="center">新闻发布系统</h1>
    <table class="gridtable" border="1" cellspacing="0" cellpadding="0" id="newsList" align="center">
    <tr align="center">
        <td>标题</td>
        <td>内容</td>
        <td>发布时间</td>
        <td>操作</td>
    </tr>
<?php
    //连接数据库
    require_once 'conn.php';
    //设置中国时区
    date_default_timezone_set("PRC");
    //查询数据库
    $sql = "select * from news";
    $result = mysqli_query($con,$sql);
    $newsList = "";
    while($rs = mysqli_fetch_array($result,MYSQLI_ASSOC)){
        $newsList[] = $rs;
    }
    // 循环用户列表
    foreach ($newsList as $news){
        echo "
            <tr>

```

```

        <td> ".$news['title']. "</td>
        <td> ".$news['content']. "</td>
        <td> ".$news['reporttime']. "</td>
        <td> <a href='addNews.php'>增加</a>
        <a href='deleteNews.php?id=".$news['id']. "';\"> 删除</a>
        <a href='editNews.php?id=".$news['id']. "';\"> 修改</a>
        </td>
    </tr>
    ";
}
?>
</table>
</body>
</html>

```

5. addNews.php 和 add.php

这两个文件为添加新闻页面。

addNews.php 的代码如下：

```

<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>新闻发布</title>
</head>
<body>
<h1 align="center">新闻发布系统</h1>
<form action="add.php" method="post" style="margin-left:41%">
    <input type="hidden" name="id" value=" " />
    标题: <input type="text" name="title"/><br/>
    <span style="height:148px;display:block;vetical-align:top;float:left;">内容: </span>
    <textarea cols=30 rows=5 name="content"></textarea><br/><br/>
    <input type="submit" value="发布新闻" style="margin-left:10%" />
</form>
</body>
</html>

```

add.php 的代码如下：

```

<?php
require_once 'conn.php';
//设置中国时区
date_default_timezone_set("PRC");
$title=trim($_POST['title']);
$content=trim($_POST['content']);
$reporttime= date('Y-m-d H:i:s');
$sql = "insert into news (title,content,reporttime)".
"values('$title', '$content', '$reporttime)";

```

```

echo $sql;
// 执行 sql 语句
mysqli_query($con,$sql);
// 获取影响的行数
$rows = mysqli_affected_rows($con);
// 返回影响行数
// 如果影响行数>=1,则判断添加成功,否则失败
if($rows >= 1){
    alert("添加成功");
    href("newsList.php");
}else{
    alert("添加失败");
}
function alert($title){
    echo "<script type='text/javascript'>alert('$title');</script>";
}
function href($url){
    echo "<script type='text/javascript'>window.location.href='$url'</script>";
}
?>

```

6. editNews.php 和 edit.php

这两个文件为修改新闻页面。

editNews.php 的代码如下：

```

<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>编辑用户</title>
</head>
<body>
<h1 align="center">新闻发布系统</h1>
<?php
//连接数据库
require_once 'conn.php';
$id=$_GET['id'];
//设置中国时区
date_default_timezone_set("PRC");
//查询数据库
$sql = "select * from news where id=".$id;
$result = mysqli_query($con,$sql);
$news = mysqli_fetch_array($result,MYSQLI_ASSOC);
?>
<form action="edit.php" method="post" style="margin-left:41%">
    <input type="hidden" name="id" value="<?php echo $news['id']?>" /><br/>

```

```

        标题:<input type="text" name="title" value="<?php echo $news['title']?>" /><br/>
        <span style="height:148px;display:block;vetical-align:top;float:left;">内容:</span>
        <textarea cols=30 rows=5 name="content"><?php echo $news['content']?></textarea><br/><br/>
        <input type="submit" value="修改新闻" style="margin-left:10%" />
    </form>
</body>
</html>

```

edit.php 的代码如下:

```

<?php
require_once 'conn.php';
//设置中国时区
date_default_timezone_set("PRC");
//获取用户信息
$id = $_POST['id'];
$title=trim($_POST['title']);
$content=trim($_POST['content']);
$reporttime= date('Y-m-d H:i:s');
$sql = "update news set title='$title',content='$content',reporttime='$reporttime' where id='$id'";
echo $sql;
// 执行 sql 语句
mysqli_query($con,$sql);
// 获取影响的行数
$rows = mysqli_affected_rows($con);
// 返回影响行数
// 如果影响行数>=1,则判断添加成功,否则失败
if($rows >= 1)
{
    alert("编辑成功");
    href("newsList.php");
}else{
    alert("编辑失败");
}
function alert($title){
    echo "<script type='text/javascript'>alert('$title');</script>";
}
function href($url){
    echo "<script type='text/javascript'>window.location.href='$url'</script>";
}
?>

```

7. deleteNews.php

该文件为新闻删除页面，代码如下：

```

<?php
// 包含数据库文件
require_once 'conn.php';

```

```

// 获取删除的 id
$id = $_GET['id'];
$row = delete($id,$con);
if($row >=1){
    alert("删除成功");
}else{
    alert("删除失败");
}
// 跳转到用户列表页面
href("newsList.php");
function delete($id,$con){
    $sql = "delete from news where id='$id'";
    // 执行删除
    mysqli_query($con,$sql);
    // 获取影响的行数
    $rows = mysqli_affected_rows($con);
    // 返回影响行数
    return $rows;
}
function alert($title){
    echo "<script type='text/javascript'>alert('$title');</script>";
}
function href($url){
    echo "<script type='text/javascript'>window.location.href='$url'</script>";
}
?>

```

小白高级项目——开发企业会员管理系统

在动态网站中，用户管理系统是非常必要的，因为网站会员的收集与数据使用，不仅可以让网站累积会员人脉，利用这些会员的数据，也可能为网站带来无穷的商机。就以一个企业会员管理系统为例，来介绍 **MySQL** 在开发中的应用技能。

1. 系统功能描述

该案例介绍一个基于 **PHP+MySQL** 的企业会员管理系统。该系统功能主要包括管理员登录及验证、会员管理、增加会员、删除会员、修改会员信息、查看会员详情等功能。

整个项目以登录界面为起始，在管理员输入用户名和密码后，系统通过查询数据库验证该管理员是否存在。

验证成功则进入会员管理界面，可以查看会员列表、增加会员、删除会员、修改会员信息、查看会员详情，进行相应的功能操作。

2. 系统功能分析

一个典型的企业会员管理系统，包括管理员登录及验证、会员管理、增加会员、删除会员、修改会员信息、查看会员详情等功能。本节就来学习企业会员管理系统的功能以及实现方法。

整个项目包含 5 个功能：

(1) 管理员登录及验证：在登录界面管理员输入用户名和密码后，系统验证通过查询数据库验证是否存在该管理员，如果验证成功显示会员管理界面，否则提示“无效的用户名和密码”，并返回登录界面。

(2) 增加会员：管理员登录系统后，可以查看会员列表，然后在对应会员操作，选择“增加”，系统进入增加会员界面，在输入相应信息后，点击“增加用户”，系统会向数据库插入一条记录，并提示成功，返回会员管理界面。

(3) 删除会员：在对应会员操作，选择“删除”后，系统会向数据库删除此条记录，并提示成功，返回会员管理界面。

(4) 修改会员信息：在对应会员操作，选择“删除”后，系统进入修改会员界面，在修改相应信息后，点击“确定修改”，系统会向数据库更新一条记录，并提示成功，返回会员管理界面。

(5) 查看会员详情：在对应会员操作，选择“查询”后，系统向数据库查询并显示该会员的详细信息。

3. 数据流程和数据库

根据系统功能和数据库设计原则，设计数据库 **usermanagement**。SQL 语法如下：

```
CREATE DATABASE IF NOT EXISTS `usermanagement`;
```

设计数据表 **admin** 并插入数据，SQL 语法如下：

```
CREATE TABLE IF NOT EXISTS `admin` (  
  `userid` char(15) DEFAULT NULL,  
  `pssw` char(10) NOT NULL  
);
```

根据系统功能和数据库设计原则，共设计 2 张表，分别是：管理员表 **admin**、会员表 **user**。

表 1 管理员表 **admin**

字段名	数据类型	字段说明
userid	char(15)	管理员编码

pssw	char(10)	密码
表 2 会员表 user		
字段名	数据类型	字段说明
id	int(11)	用户编码,自增
name	varchar(32)	用户名
pswd	char(10)	密码
sex	char(4)	性别
age	char(50)	年龄
hobby	char(50)	爱好
addtime	datetime	注册时间
logintime	datetime	最近登录时间
detail	varchar(100)	个人简介

创建管理员表 **admin**，SQL 语句如下：

```
CREATE TABLE IF NOT EXISTS `admin` (
  `userid` char(15) DEFAULT NULL,
  `pssw` char(10) NOT NULL
);
```

插入演示数据，SQL 语句如下：

```
INSERT INTO `admin` (`userid`, `pssw`) VALUES
('test', '123');
```

创建会员表 **user**，SQL 语句如下：

```
CREATE TABLE IF NOT EXISTS `user` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `name` varchar(32) COLLATE utf8_unicode_ci NOT NULL,
  `pswd` char(10) COLLATE utf8_unicode_ci NOT NULL,
  `sex` char(4) COLLATE utf8_unicode_ci DEFAULT NULL,
  `age` char(50) COLLATE utf8_unicode_ci DEFAULT NULL,
  `hobby` char(50) COLLATE utf8_unicode_ci DEFAULT NULL,
  `addtime` datetime DEFAULT NULL,
  `logintime` datetime DEFAULT NULL,
  `detail` varchar(100) COLLATE utf8_unicode_ci DEFAULT NULL,
  PRIMARY KEY (`id`)
);
```

插入演示数据，SQL 语句如下：

```
INSERT INTO `user` (`id`, `name`, `pswd`, `sex`, `age`, `hobby`, `addtime`, `logintime`, `detail`) VALUES
(1, 'sun', '111', '1', '16', '上网', '2017-10-24 20:51:15', '2017-10-24 20:51:17', '111 '),
(2, '小齐', '123', '1', '17', '锻炼', '2017-10-24 20:56:17', '2017-10-24 20:56:18', '运动快乐 '),
(6, '小花', '123456', '2', '15', '购物 shopping', '0000-00-00 00:00:00', '0000-00-00 00:00:00', '买买买 ');
```

4. 代码的具体实现

该案例的代码清单包含 10 个 php 文件，实现了企业会员管理系统的管理员登录及验证、查看会员列表、增加会员、删除会员、修改会员信息、查看会员详情等主要功能。

网站中文件的含义和代码如下：

1. index.php

文件是该案例的 **web** 访问入口，是用户的登录界面。具体代码如下：

```
<html>
<head>
<title>登录
</title>
</head>

<body>
<h1 align="center">会员管理系统</h1>
<table width="100%" style="text-align:center">
<tr>
<form action="login.php" method="post">
<td width="60%" class="sub1">
<p class="sub">账号： <input type="text" name="userid" align="center" class="txttop"></p>
<p class="sub">密 码： <input type="password" name="pssw" align="center" maxlength="4"
class="txtbot"></p>
<button name="button" class="button" type="submit">登录</button>
</form>
</td>
</tr>
</table>
</body>
</html>
```

2. conn.php

该文件为数据库连接页面，代码如下：

```
<?php
// 创建数据库连接
$con = mysqli_connect("localhost", "root", "") or die("无法连接到数据库");
mysqli_select_db($con, "usermanagement") or die(mysqli_error($con));
mysqli_query($con, 'set NAMES utf8');
?>
```

3. log.php

该文件是对用户登录进行验证。代码如下：

```
<html>
<head>
<title>
</title>
<link rel="stylesheet" type="text/css" href="css/main.css">
</head>
<body><h1 align="center">会员管理系统</h1></body>
<p align="center">
<?php
```

```

//连接数据库
require_once("conn.php");
//账号
$userid=$_POST['userid'];
//密码
$pssw=$_POST['pssw'];
//查询数据库
$qry=mysqli_query($con,"SELECT * FROM admin WHERE userid='$userid'");
$row=mysqli_fetch_array($qry,MYSQLI_ASSOC);
//验证用户
if($userid==$row['userid'] && $pssw==$row['pssw']&&$userid!=null&&$pssw!=null)
{
    session_start();
    $_SESSION["login"] =$userid;
    header("Location: userlist.php");
}
else{
    echo "无效的账号或密码!";
    header('refresh:1; url= index.php');
}
//}
?>
</p>
</body>
</html>

```

4. list.php

该文件为系统的主界面。具体代码如下：

```

<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<style type="text/css">
table.gridtable {
    font-family: verdana,arial,sans-serif;
    font-size:11px;
    color:#333333;
    border-width: 1px;
    border-color: #666666;
    border-collapse: collapse;
}
table.gridtable th {
    border-width: 1px;
    padding: 8px;
    border-style: solid;
    border-color: #666666;

```

```

        background-color: #dedede;
    }
    table.gridtable td {
        border-width: 1px;
        padding: 8px;
        border-style: solid;
        border-color: #666666;
        background-color: #ffffff;
    }
</style>
<title>会员信息列表</title>
</head>
<body>
<h1 align="center">会员管理系统</h1>
    <table class="gridtable" border="1" cellspacing="0" cellpadding="0" id="userList" align="center">
    <tr align="center">
        <td>用户名</td>
        <td>密码</td>
        <td>性别</td>
        <td>年龄</td>
        <td>爱好</td>
        <td>注册时间</td>
        <td>最后登录时间</td>
        <td>操作</td>
    </tr>
<?php
    //连接数据库
    require_once 'conn.php';
    //设置中国时区
    date_default_timezone_set("PRC");
    //查询数据库
    $sql = "select * from user";
    $result = mysqli_query($con,$sql);
    $userList = "";
    while($rs = mysqli_fetch_array($result,MYSQLI_ASSOC)){
        $userList[] = $rs;
    }

    // 循环用户列表
    foreach ($userList as $user){
        echo "
            <tr>
                <td> ".$user['name']. "</td>
                <td> ".$user['pswd']. "</td>
                <td> ".$user['sex']. "</td>

```

```

        <td> ".$user['age']."</td>
        <td> ".$user['hobby']."</td>
        <td> ".date("Y-m-d",$user['addtime'])."</td>
        <td> ".date("Y-m-d",$user['logintime'])."</td>
        <td> <a href='addUser.php'>增加</a>
        <a href='deleteUser.php?id=".$user['id'].");\"> 删除</a>
        <a href='editUser.php?id=".$user['id'].");\"> 修改</a>
        <a href='detailUser.php?id=".$user['id'].");\"> 查询</a>
    </td>
</tr>
";
    }
?>
</table>
</body>
</html>

```

5. addUser.php 和 add.php

这两个文件为添加会员页面。

addUser.php 的代码如下：

```

<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>新增用户</title>
</head>
<body>
<h1 align="center">会员管理系统</h1>
<form action="add.php" method="post">
    <input type="hidden" name="user_id" value=" " />
    <table width="444" align="center">
        <tr>
            <td>用户名: </td>
            <td> <input type="text" name="name" size="10" /></td>
        </tr>
        <tr>
            <td>密码: </td>
            <td> <input type="pswd" name="pswd" size="10" /></td>
        </tr>
        <tr>
            <td>性别: </td>
            <td> <input type="radio" name="sex" value="男" checked="checked" /> 男<input type="radio"
name="sex" value="女" /> 女 </td>
        </tr>
    </table>

```

```

        <td>年龄: </td>
        <td> <input type="text" name="age" size="3" /></td>
    </tr>
    <tr>
        <td>爱好: </td>
        <td> <input type="text" name="hobby" size="44" /></td>
    </tr>
    <tr>
        <td style="height:148px;display:block;vetical-align:top;float:left;">个人简介: </td>
        <td> <textarea name="detail" rows="10" cols="30"></textarea></td>
    </tr>
    <tr>
        <td colspan="2" align="center"><input type="submit" value="增加会员" /></td>
    </tr>
</table>
<p> </p>
<p> </p>
<p> </p>
</form>
</body>
</html>

```

add.php 的代码如下:

```

<?php
require_once 'conn.php';
//获取用户信息
$id = $_POST['id'];
$name = $_POST['name'];
echo $name;
$pswd= $_POST['pswd'];
if($_POST['sex']=='男')
{
    $sex=1;
}
else if($_POST['sex']=='女')
{
    $sex=2;
}
else $sex=3;
;
//设置中国时区
date_default_timezone_set("PRC");
$age = $_POST['age'];
$hobby = $_POST['hobby'];
$detail = $_POST['detail'];

```

```

$addtime=mktime(date("h"),date("m"),date("s"),date("m"),date("d"),date("Y"));
$logintime=$addTime;

$sql = "insert into user (name,pswd,sex,age,hobby,detail,addtime,logintime) ".
"values('$name','$pswd','$sex','$age','$hobby','$detail','$addtime','$logintime')";
echo $sql;
// 执行 sql 语句
mysqli_query($con,$sql);
// 获取影响的行数
$rows = mysqli_affected_rows($con);
// 返回影响行数
// 如果影响行数>=1,则判断添加成功,否则失败
if($rows >= 1){
    alert("添加成功");
    href("userList.php");
}else{
    alert("添加失败");
}
function alert($title){
    echo "<script type='text/javascript'>alert('$title');</script>";
}
function href($url){
    echo "<script type='text/javascript'>window.location.href='$url'</script>";
}
?>

```

6. editUser.php 和 edit.php

这两个文件为修改会员页面。

editUser.php 的代码如下：

```

<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<style type="text/css">
table.gridtable {
    font-family: verdana,arial,sans-serif;
    font-size:11px;
    color:#333333;
    border-width: 1px;
    border-color: #666666;
    border-collapse: collapse;
}
table.gridtable th {
    border-width: 1px;
    padding: 8px;
    border-style: solid;

```

```

        border-color: #666666;
        background-color: #dedede;
    }
    table.gridtable td {
        border-width: 1px;
        padding: 8px;
        border-style: solid;
        border-color: #666666;
        background-color: #ffffff;
    }
</style>
<title>查看会员详情</title>
</head>
<body>
<h1 align="center">会员管理系统</h1>
<?php
//连接数据库
require_once 'conn.php';
$id=$_GET['id'];
//设置中国时区
date_default_timezone_set("PRC");
//查询数据库
$sql = "select * from user where id=".$id;
$result = mysqli_query($con,$sql);
$user = mysqli_fetch_array($result,MYSQLI_ASSOC);
?>

<table class="gridtable" width="444" align="center">
    <tr>
        <td>用户 ID </td>
        <td><?php echo $id ?> </td>
    </tr>
    <tr>
        <td>用户名 </td>
        <td><?php echo $user['name'] ?> </td>
    </tr>
    <tr>
        <td>密码</td>
        <td><?php echo $user['pswd'] ?> </td>
    </tr>
    <tr>
        <td>性别</td>
        <td><?php if($user[sex]=='1') echo "男"; else if($user[sex]=='2') echo "女"; else "保密"; ?>
        </td>
    </tr>

```

```

        <tr>
            <td>年龄</td>
            <td><?php echo $user['age'] ?> </td>
        </tr>
        <tr>
            <td>爱好</td>
            <td><?php echo $user['hobby'] ?> </td>
        </tr>
        <tr>
            <td>个人简介</td>
            <td><?php echo $user['detail'] ?> </td>
        </tr>
        <tr>
            <td colspan="2" align="center"><a href="userList.php" >返回用户列表</a></td>
        </tr>
    </table>
    <p> </p>
    <p> </p>
    <p> </p>
</body>
</html>

```

edit.php 的代码如下：

```

<?php
require_once 'conn.php';
//设置中国时区
date_default_timezone_set("PRC");
//获取用户信息
$id = $_POST['id'];
$name = $_POST['name'];
$pswd = $_POST['pswd'];
if($_POST['sex']=='男')
{
    $sex=1;
}
else if($_POST['sex']=='女')
{
    $sex=2;
}
else $sex=3;
;
$age = $_POST['age'];
$hobby = $_POST['hobby'];
$detail = $_POST['detail'];
$addtime=mktime(date("h"),date("m"),date("s"),date("m"),date("d"),date("Y"));

```



```

    $loginTime=$addTime;

    $sql = "update user set
name='$name',pswd='$pswd',sex='$sex',age='$age',hobby='$hobby',detail='$detail' where id='$id'";

    echo $sql;
    // 执行 sql 语句
    mysqli_query($con,$sql);
    // 获取影响的行数
    $rows = mysqli_affected_rows($con);
    // 返回影响行数
    // 如果影响行数>=1,则判断添加成功,否则失败
    if($rows >= 1)
    {
        alert("编辑成功");
        href("userList.php");
    }else{
        alert("编辑失败");
    }
}

function alert($title){
    echo "<script type='text/javascript'>alert('$title');</script>";
}

function href($url){
    echo "<script type='text/javascript'>window.location.href='$url'</script>";
}

?>

```

7. deleteUser.php

该文件为用户删除页面，代码如下：

```

<?php
// 包含数据库文件
require_once 'conn.php';
// 获取删除的 id
$id = $_GET['id'];
$row = delete($id,$con);
if($row >=1){
    alert("删除成功");
}else{
    alert("删除失败");
}
// 跳转到用户列表页面
href("userList.php");
function delete($id,$con){
    $sql = "delete from user where id='$id'";
    // 执行删除
    mysqli_query($con,$sql);
    // 获取影响的行数

```

```

    $rows = mysqli_affected_rows($con);
    // 返回影响行数
    return $rows;
}
function alert($title){
    echo "<script type='text/javascript'>alert('$title');</script>";
}
function href($url){
    echo "<script type='text/javascript'>window.location.href='$url'</script>";
}
?>

```

8. detailUser.php

该页面为查看会员详情页面。代码如下：

```

<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<style type="text/css">
table.gridtable {
    font-family: verdana,arial,sans-serif;
    font-size:11px;
    color:#333333;
    border-width: 1px;
    border-color: #666666;
    border-collapse: collapse;
}
table.gridtable th {
    border-width: 1px;
    padding: 8px;
    border-style: solid;
    border-color: #666666;
    background-color: #dedede;
}
table.gridtable td {
    border-width: 1px;
    padding: 8px;
    border-style: solid;
    border-color: #666666;
    background-color: #ffffff;
}
</style>
<title>查看会员详情</title>
</head>
<body>

```

```
<h1 align="center">会员管理系统</h1>
<?php
//连接数据库
require_once 'conn.php';
$id=$_GET['id'];
//设置中国时区
date_default_timezone_set("PRC");
//查询数据库
$sql = "select * from user where id=".$id;
$result = mysqli_query($con,$sql);
$user = mysqli_fetch_array($result,MYSQLI_ASSOC);
?>
<table class="gridtable" width="444" align="center">
    <tr>
        <td>用户 ID </td>
        <td><?php echo $id ?> </td>
    </tr>
    <tr>
        <td>用户名 </td>
        <td><?php echo $user['name'] ?> </td>
    </tr>
    <tr>
        <td>密码</td>
        <td><?php echo $user['pswd'] ?> </td>
    </tr>
    <tr>
        <td>性别</td>
        <td><?php if($user[sex]=='1') echo "男"; else if($user[sex]=='2') echo "女"; else "保密"; ?>
        </td>
    </tr>
    <tr>
        <td>年龄</td>
        <td><?php echo $user['age'] ?> </td>
    </tr>
    <tr>
        <td>爱好</td>
        <td><?php echo $user['hobby'] ?> </td>
    </tr>
    <tr>
        <td>个人简介</td>
        <td><?php echo $user['detail'] ?> </td>
    </tr>
    <tr>
        <td colspan="2"><a href="userList.php" >返回用户列表</a></td>
```

```
        </tr>
    </table>

    <p> </p>
    <p> </p>
    <p> </p>

</body>
</html>
```