《软件安全》实验报告

姓名：王昱　　学号：2212046　　班级：信息安全班

一、实验名称：

Web 开发实践

二、实验要求：

复现课本第十章的实验三(10.3.5 节):利用 php，编写简单的数据库插入、查询和删除操作的示例。基于课本的完整的例子，进一步了解 WEB 开发的细节。

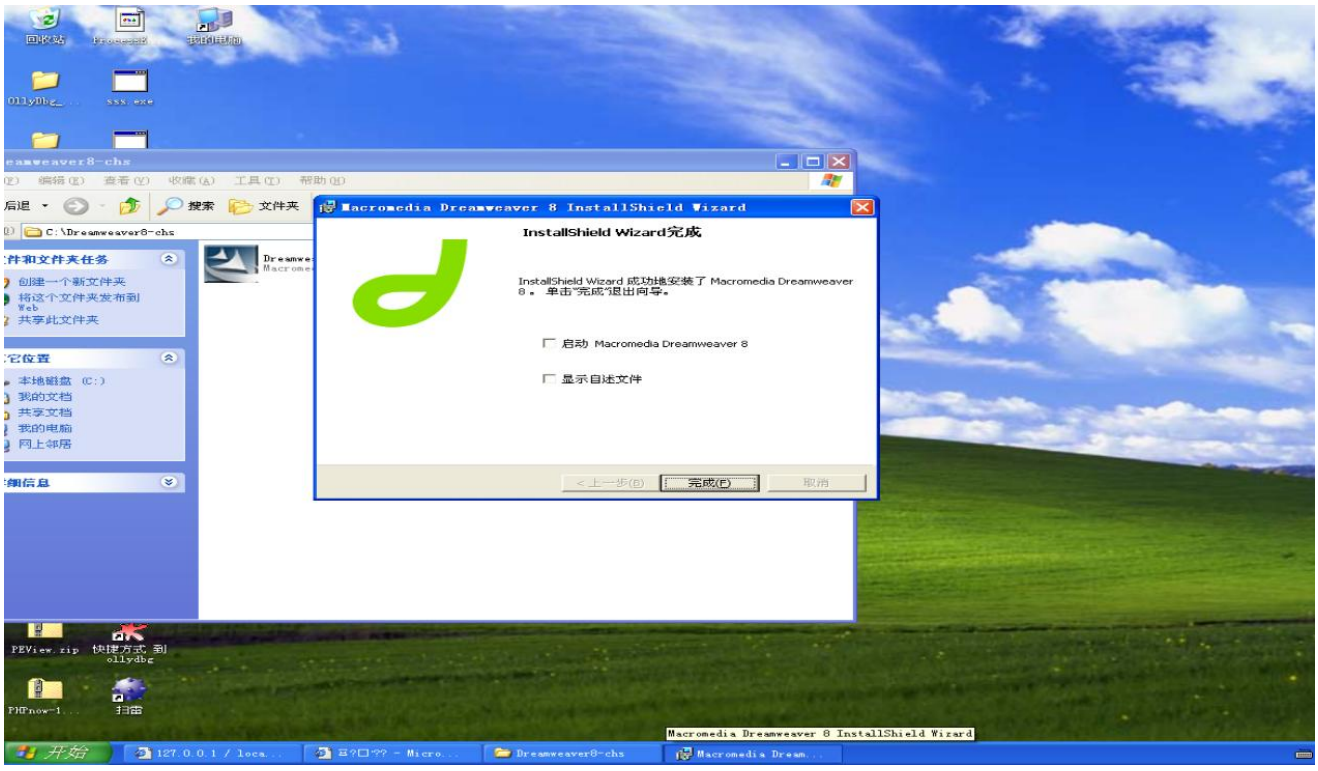
三、实验原理：

PHP Web 开发是使用 PHP 语言编写服务器端代码以创建动态网站和 Web 应用程序的过程。它通过嵌入 HTML 代码，处理客户端请求，生成动态网页内容，具有跨平台、与 HTML 无缝集成、丰富的内置功能库和强大的社区支持等特点。PHP Web 开发包括安装 PHP 和 Web 服务器、创建 PHP 文件、与数据库交互和处理用户输入等步骤，适用于构建从简单的个人网站到复杂的企业级应用的各种 Web 项目。

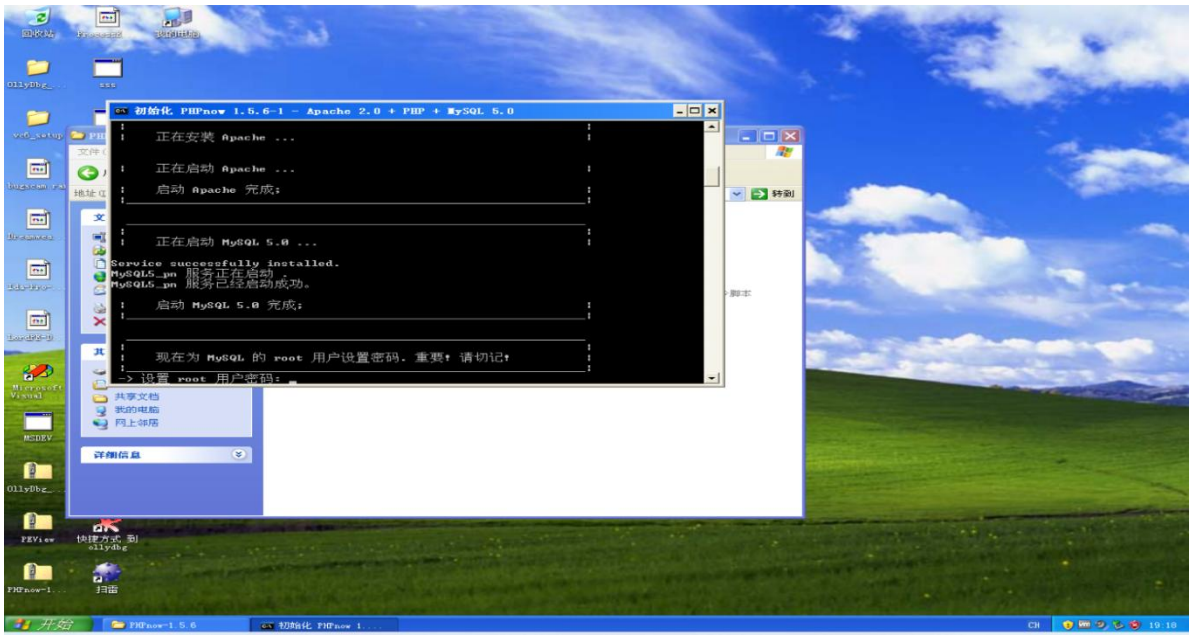
四、实验过程：

(一)在 Windows XP 环境下安装 Dreamweaver,Phpnow（注意路径不能含有中文）

安装 Dreamweaver：

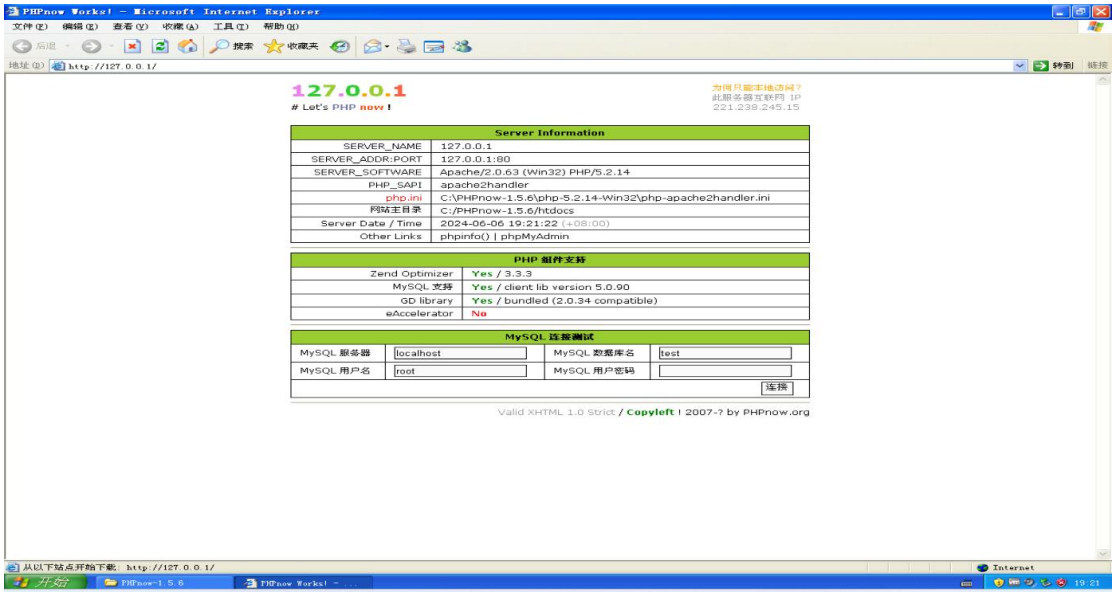


安装 Phpnow:



设置 mysql 密码：123456

按任意键进入默认界面：

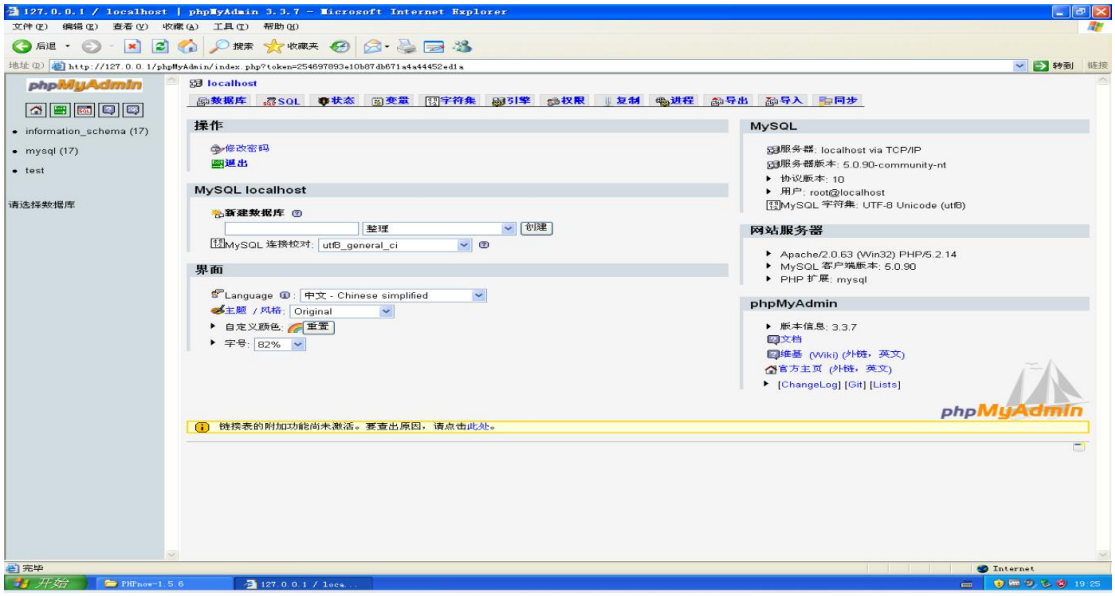


输入密码连接数据库：

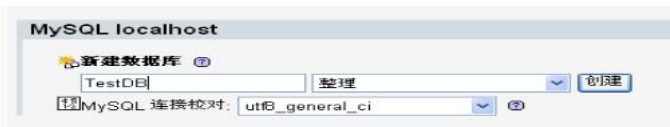
至此，环境配置完毕！

（二）创建数据库

1.点击 phpMyAdmin,进入数据库管理界面

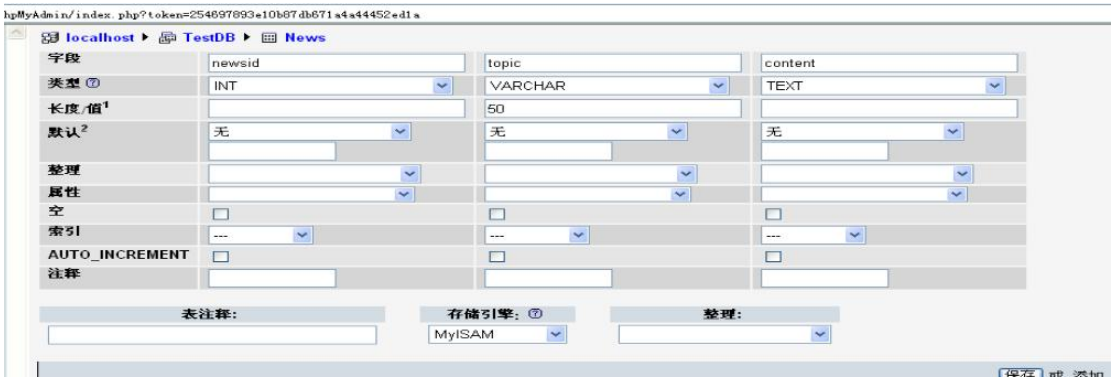


2.创建数据库 TestDB

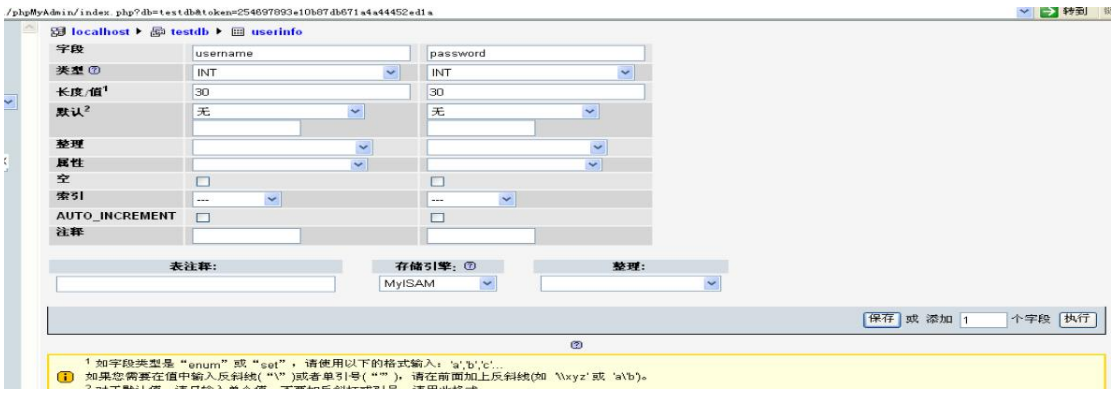


3.创建 News（newsid,topic,content）、userinfo（username, password）两张表，如下图所示：

- News（newsid,topic,content）：

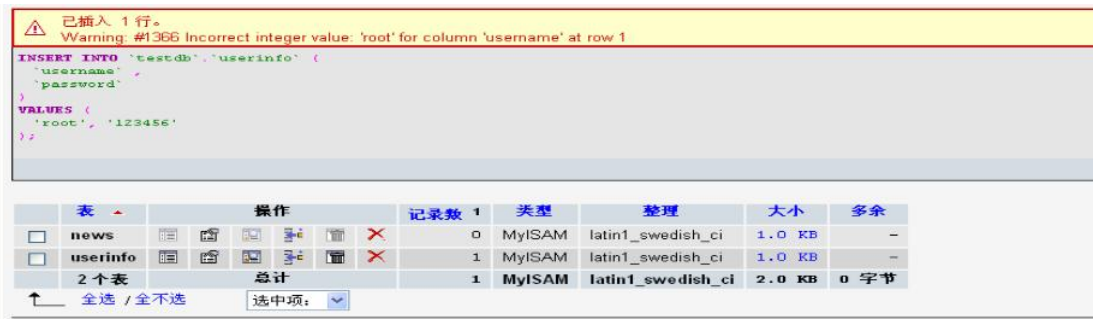


- userinfo（username, password）：



4.在表中初始化数据

在 userinfo 插入一条（root，123456）的一行数据，为了之后用户能够使用该用户名和密码正确登录.



(三)编写 php 脚本文件

1.编写 login.html

首先编写登录代码，写入如下程序：

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>无标题文档</title>
</head>
<body>
<form id="form1" name="form1" method="post" action="loginok.php">
  <p>
    <label for="textfield">username:</label>
    <input type="text"    name="username" id="username">
  </p>
  <p>
    <label for="password">password:</label>
    <input type="password"    name="password" id="password">
  </p>
  <p>
    <label>
    <input type="submit" name="Submit" value="提交" />
    </label>
  </p>
</form>
</body>
</html>
```

将文件放到 phpnow 的 htdoc 目录下，进行测试，打开 html 文件后出现如下界面：

这段代码定义了一个简单的用户登录表单，用户可以输入用户名和密码，然后点击提交按钮将数据发送到 loginok.php 文件进行处理。

2.编写 Loginok.php 文件

本文件的目的是验证在 login.html 文件打开的网页上所输入的帐号和密码是否在数据库的 userinfo 表中，从而来处理用户的登录，即成功的话载入下一个界面，不成功的话返回登录界面重新登陆。由于我们已经有一条用户名为 root，密码为 123456 的记录，所以进行登录。

```
<?php
  $loginok=0;
  $conn=mysql_connect("localhost", "root", "123456");
  $username = $_POST['username'];
  $pwd = $_POST['password'];
  $SQLStr = "SELECT * FROM userinfo where username='$username' and pwd='$pwd'";
  echo $SQLStr;
$result=mysql_db_query("testDB", $SQLStr, $conn);
  if ($row=mysql_fetch_array($result))//通过循环读取数据内容
  {
  $loginok=1;
  }
  // 释放资源
  mysql_free_result($result);
  // 关闭连接
  mysql_close($conn);
  if ($loginok==1)
  {
```

```
?>
<script>
alert("login succes");
window.location.href="sys.php";
</script>
<?php
}
else{
?>
<script>
alert("login failed");
history.back();
</script>
<?php
}

?>
```

这段 PHP 代码的主要作用是处理用户登录。 它首先尝试连接到本地的 MySQL 数据库，然后获取用户在 HTML 表单中提交的用户名和密 码。然后，它在数据库的 userinfo 表中查找与这个用户名和密码匹配的记录。如果找到了匹配的记录，那么它会设置一个标志变量 $loginok 为 1，表示登录成功。

然后释放数据库查询的结果，关闭数据库连接。 接着，它会检查 $loginok 的值。如果为 1，就在网页上弹出一个提示框显示 "login success"，然后跳转到 "sys.php" 页

面。如果 $loginok 不为 1，表示登录失败，就弹出 一个提示框显示 "login failed"，然后返回到上一个页面。

以上，说明代码执行正确！

3.编写 sys.php 文件

```
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />
<title>主页</title>
</head>
<?php
$conn=mysql_connect("localhost", "root", "123456");
?>
<body>
<div align="center">
 <table width="900" border="0" cellspacing="0" cellpadding="0">
 <tr>
 <td height="40"><form id="form1" name="form1" method="post" action="add.php">
 <div align="right">新闻标题：
 <input name="topic" type="text" id="topic" size="50" />
 <BR>
 新闻内容：
 <textarea name="content" cols="60" rows="8" id="content"></textarea><BR>
 <input type="submit" name="Submit" value="添加" />
 </div>
 </form>
 </td>
```

```
  </tr>
  <tr>
  <td><hr /></td>
  </tr>
  <tr>
  <td height="300" align="center" valign="top"><table width="600" border="0"
cellspacing="0"
cellpadding="0">
  <tr>
  <td width="100" height="30"><div align="center">新闻序号</div></td>
  <td><div align="center">新闻标题</div></td>
  <td><div align="center">删除</div></td>
  </tr>
<?php
  $SQLStr = "select * from news";
  $result=mysql_db_query("testDB", $SQLStr, $conn);
  if ($row=mysql_fetch_array($result))//通过循环读取数据内容
  {
  // 定位到第一条记录
  mysql_data_seek($result, 0);
  // 循环取出记录
  while ($row=mysql_fetch_row($result))
  {
?>
<tr>
  <td height="30"><div align="center"> <?php echo $row[0] ?> </div></td>
  <td width="400"> <div align="center"> <?php echo $row[1] ?> </div></td>
  <td><div align="center"><a href="del.php?newsid=<?php echo $row[0] ?> " > 删 除 </a>
</div></td>
```

```
  </tr>
<?php
  }
  }
?>
  </table></td>
  </tr>
  </table>
</div>
</body>
</html>
<?php
  // 释放资源
  mysql_free_result($result);
  // 关闭连接
  mysql_close($conn);
?>
```

这段代码展示了一个简单的新闻管理页面，允许用户添加新新闻并显示现有新闻列表。HTML 部分构建了一个表单，用于输入新闻标题和内容，通过 POST 方法提交到 add.php 处理。新闻列表显示在一个表格中，每条新闻有其序号、标题和删除链接，点击删除链接会触发删除操作，指向 del.php 文件，并传递新闻 ID 进行删除。PHP 部分连接到名为 testDB 的 MySQL 数据库，从 news 表中检索新闻记录，并在页面上显示这些记录。代码最后释放数据库查询结果并关闭数据库连接。

主体界面如下：

4.编写 add.php 文件

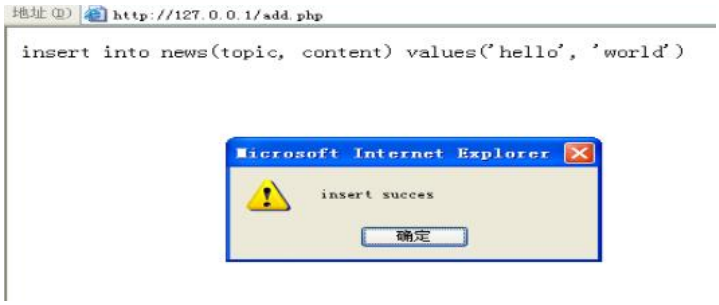
```
<?php
 $conn=mysql_connect("localhost", "root", "123456");
 mysql_select_db("testDB");
 $topic = $_POST['topic'];
 $content = $_POST['content'];
 $SQLStr = "insert into news(topic, content) values('$topic', '$content')";
 echo $SQLStr;
 $result=mysql_query($SQLStr);

 // ¹Ø±ÕÁ¬½Ó
 mysql_close($conn);
 if ($result)
 {
 ?>
 <script>
 alert("insert succes");
 window.location.href="sys.php";
 </script>
 <?php
 }
 else{
 ?>
```

```
 <script>
 alert("insert failed");
 history.back();
 </script>
 <?php
 }
?>
```

这段代码实现了从表单获取新闻标题和内容并插入到 MySQL 数据库中。首先，通过 mysql_connect 连接到数据库并选择 testDB 数据库，然后使用$_POST 获取表单提交的 topic 和 content。构建 SQL 插入语句，并通过 mysql_query 执行该语句。执行完后关闭数据库连接，并根据插入结果判断是显示插入成功还是失败的信息。如果插入成功，页面会跳转到 sys.php；如果失败，则返回到上一页面。

如图，插入成功：



可以看到，已经插入了这条记录：

5.编写 news.php：

该文件的目的就是在 index.php 中点击新闻标题后能够显示出来新闻的内容,代码如下所示:

```
<html>
```

```
<head>
<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />
<title>主页</title>
</head>
<body>
<div align="center">
 <table width="900" border="0" cellspacing="0" cellpadding="0">
 <tr>
 <td height="40"><form id="form1" name="form1" method="post" action="loginok.php">
 <div align="right">用户名：
 <input name="username" type="text" id="username" size="12" />
 密码：
 <input name="password" type="password" id="password" size="12" />
 <input type="submit" name="Submit" value="提交" />
 </div>
 </form>
 </td>
 </tr>
 <tr>
 <td><hr /></td>
 </tr>
 <tr>
 <td height="300" align="center" valign="top"><p> </p>
<?php
 $conn=mysql_connect("localhost", "root", "123456");
 $newsid = $_GET['newsid'];
 $SQLStr = "select * from news where newsid=$newsid";
 $result=mysql_db_query("testDB", $SQLStr, $conn);
 if ($row=mysql_fetch_array($result))//通过循环读取数据内容
```

```
 {
 // 定位到第一条记录
 mysql_data_seek($result, 0);
 // 循环取出记录
 while ($row=mysql_fetch_row($result))
 {
 echo "$row[1]<br>";
 echo "$row[2]<br>";
 }
 }
 // 释放资源
 mysql_free_result($result);
 // 关闭连接
 mysql_close($conn);

?>
</td>
 </tr>
 </table>
</div>
</body>
</html>
```

这段代码实现了一个简单的用户登录表单和新闻内容显示页面。HTML 部分包含了一个登录表单，用户可以输入用户名和密码并提交到 loginok.php 进行处理。PHP 部分连接到 MySQL 数据库，从 URL 参数 newsid 中获取新闻 ID，并从数据库中检索相应的新闻记录，显示新闻标题和内容。最后，释放数据库资源并关闭连接。代码使用了过时的 mysql 扩展，建议改用 mysqli 或 PDO 以提高安全性和兼容性。

界面如图所示，测试成功！

6.编写 del.php

该文件用于删除新闻：

```
<?php
 $conn=mysql_connect("localhost", "root", "123456");
 mysql_select_db("testDB");
 $newsid = $_GET['newsid'];
 $SQLStr = "delete from news where newsid=$newsid";
 echo $SQLStr;
 $result=mysql_query($SQLStr);
 // ¹Ø±ÕÁ¬½Ó
 mysql_close($conn);
 if ($result)
 {
 ?>
 <script>
 alert("delete succes");
 window.location.href="sys.php";
 </script>
 <?php
 }
 else{
 ?>
```

```
 <script>
 alert("delete failed");
 history.back();
 </script>
 <?php
 }
?>
```

这段代码实现了从数据库中删除新闻记录的功能。首先，通过 mysql_connect 连接到数据库，并选择 testDB 数据库。然后，从 URL 参数中获取要删除的新闻 ID (newsid)，并构建 SQL 删除语句。执行该删除操作后，关闭数据库连接。如果删除操作成功，页面会显示成功提示并跳转到 sys.php 页面；如果失败，则显示失败提示并返回到上一页面。

与添加原理基本一致，测试后成功删除！以上我们就完成了本次实验。

六、心得体会：

在学习和实践 PHP Web 开发的过程中，我深刻体会到了其强大与灵活性。PHP 作为一种开源的服务器端脚本语言，具有易于学习、与 HTML 无缝集成、跨平台支持和丰富的内置函数库等优势，使其成为 Web 开发的首选工具之一。通过 PHP，可以轻松地实现从简单的动态网页到复杂的 Web 应用程序的开发。与数据库的无缝连接，特别是 MySQL，极大地提高了数据处理的效率和便利性。同时，庞大的开发者社区和丰富的在线资源为问题的解决和技能的提升提供了强有力的支持。

总的来说，掌握 PHP Web 开发不仅增强了我的编程能力，还为我打开了一个更广阔的 Web 开发世界。