《WEB开发实践》实验报告

姓名：汤清云 学号：2013536 班级： 1075

**实验名称：**

**WEB开发实践**

**实验要求：**

**利用php编写简单的数据库插入、删除、查询操作的示例。**

**进一步了解WEB开发实践的细节。**

**实验知识点：**

**URL即标识以http或https协议请求的资源。**

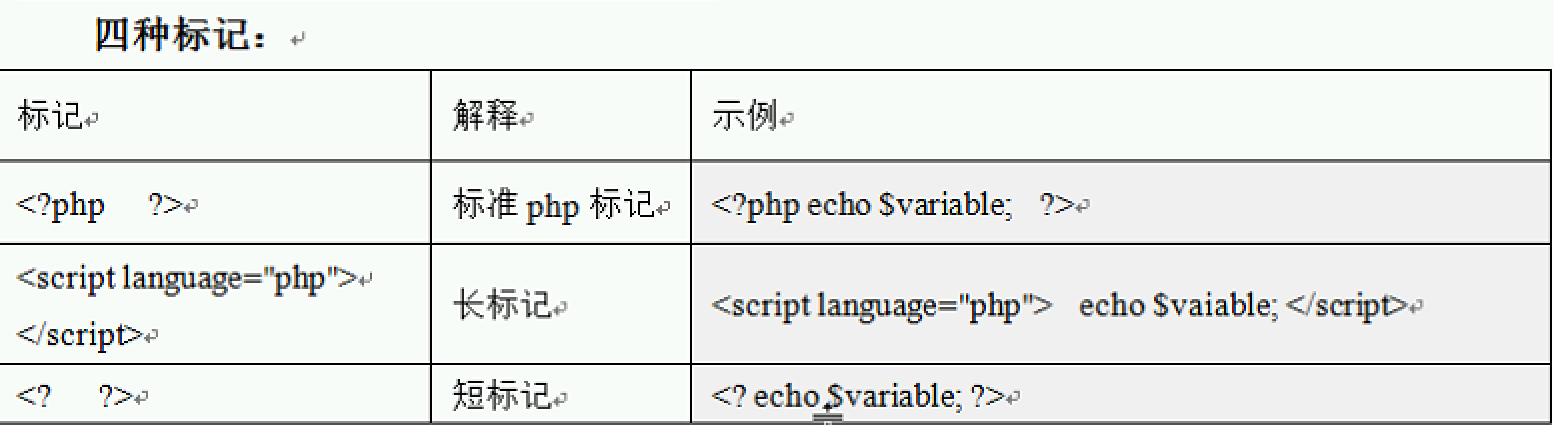
**HTML语言head应包括以下元素：**

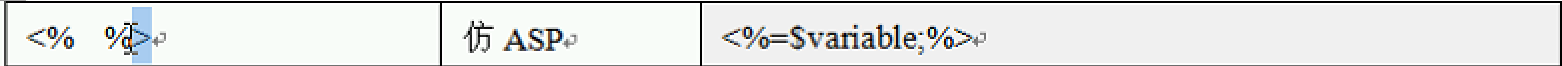


**其语言主体应被包括在<body> </body>中。**

**http属于无状态通信协议，服务器不会记录请求发自哪个浏览器。故而服务器引入session机制，使用类似散列表的方式来记录信息；而客户端引入cookie机制记录这些请求是否发送给同一个服务器。一般而言，服务器使用session维护客户端的会话状态，客户端使用cookie存储当前客户端的会话状态的session id，在使用时将它传递给服务器，让服务器找到对应状态并呈现给客户端。**

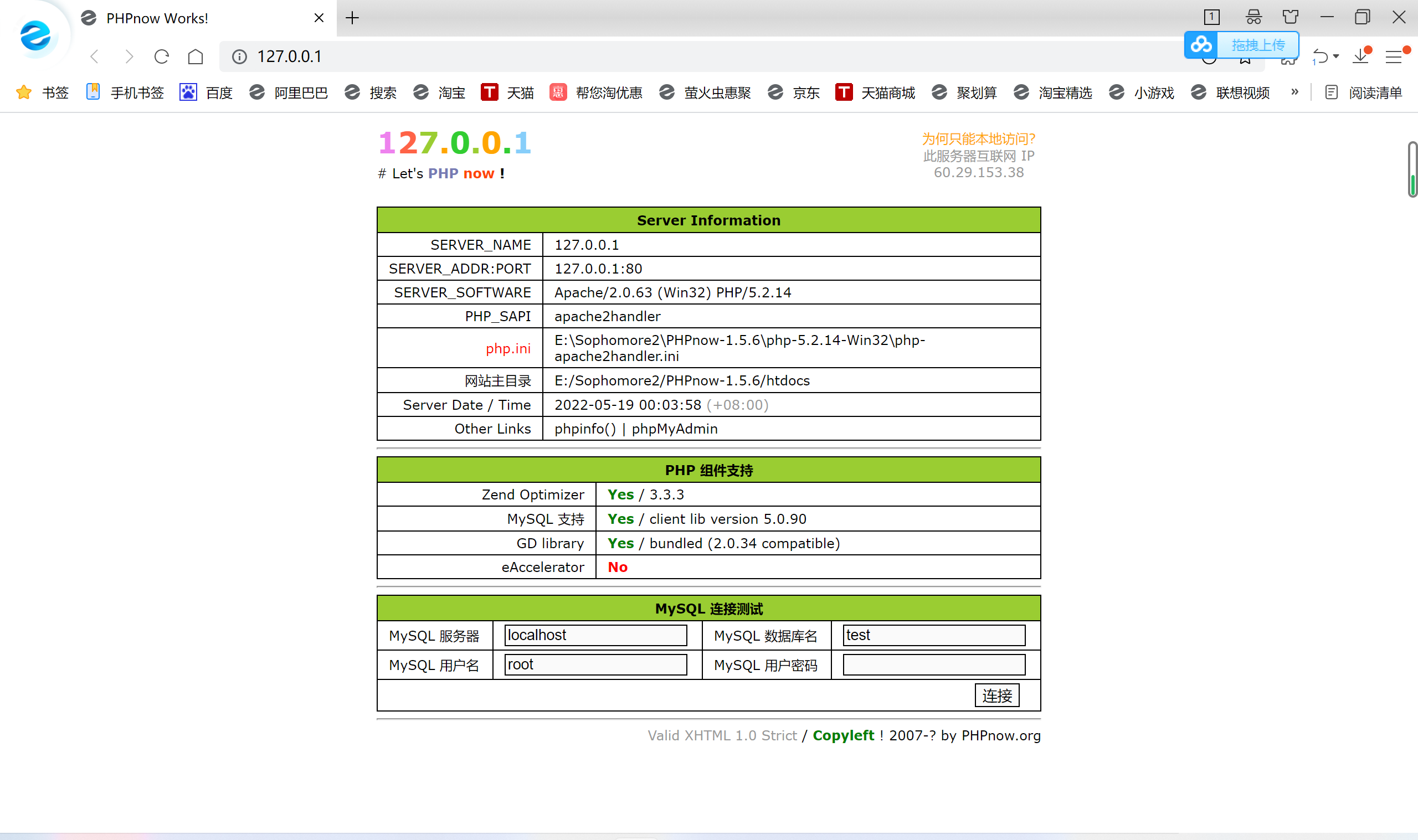
**PHP语言是一种免费的脚本语言，所有PHP语句都被包括在PHP标记里，PHP标记外的语句都会直接输出。PHP的四种标记如下：**





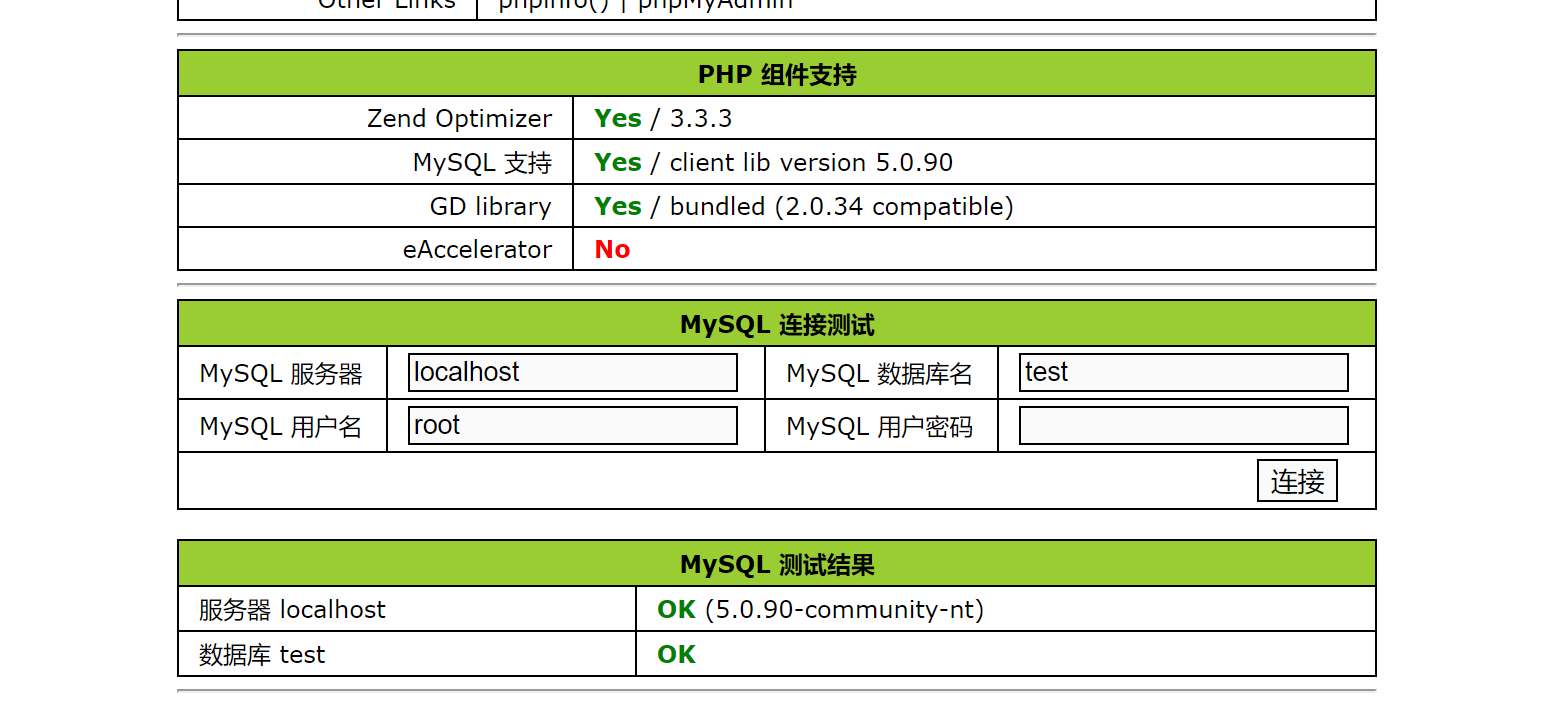
**实验过程：**

1. **安装PHPnow：使用cmd.exe以管理员身份运行Init.exe，完成各项任务安装。此时弹出以下界面，访问资源为hpdocs-index.php的默认资源**

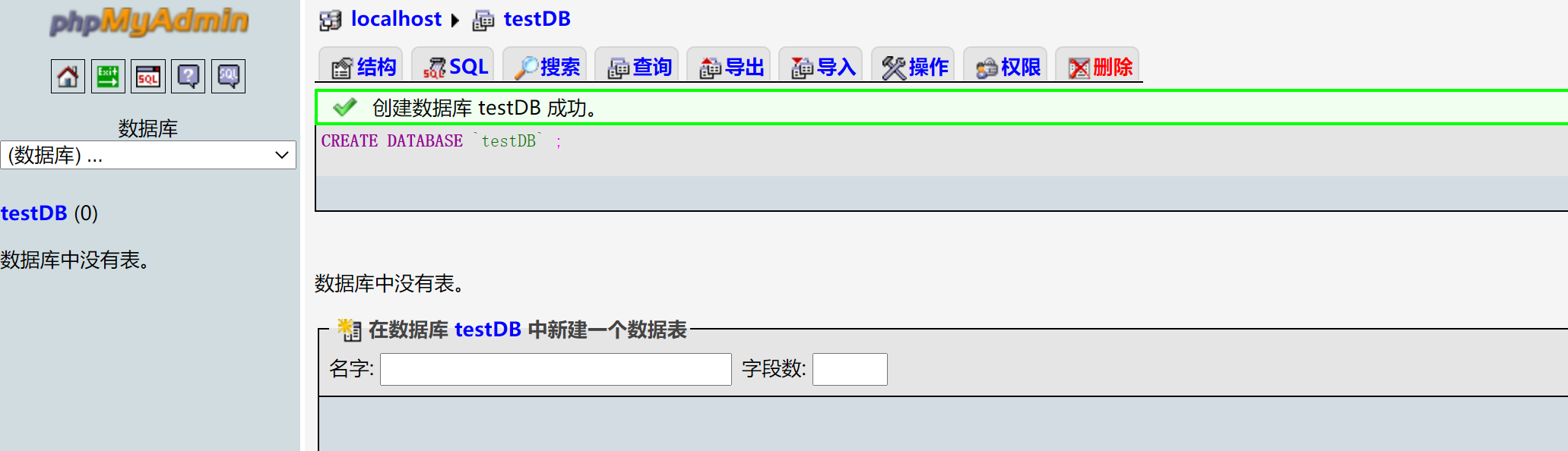




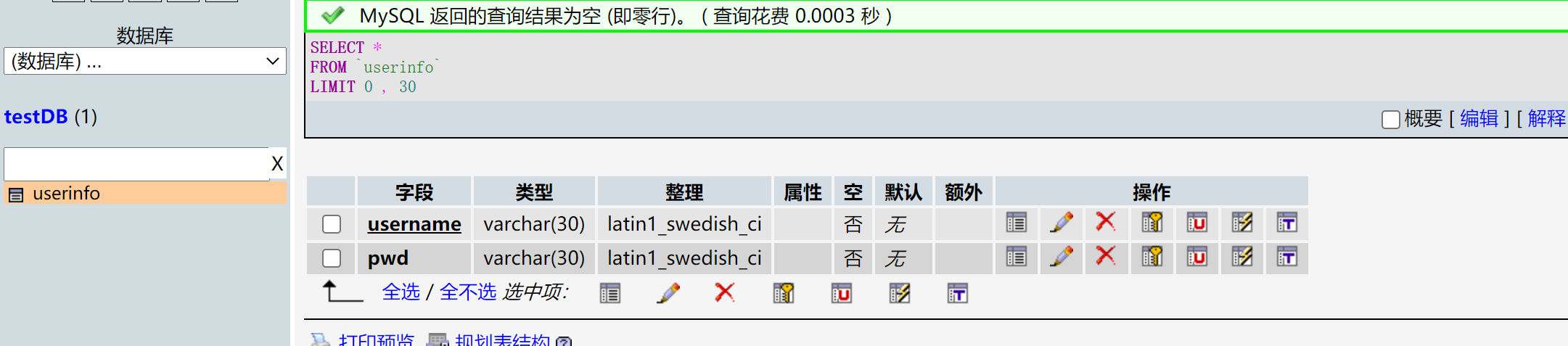
1. **输入密码123456查看数据库连接是否正常：**



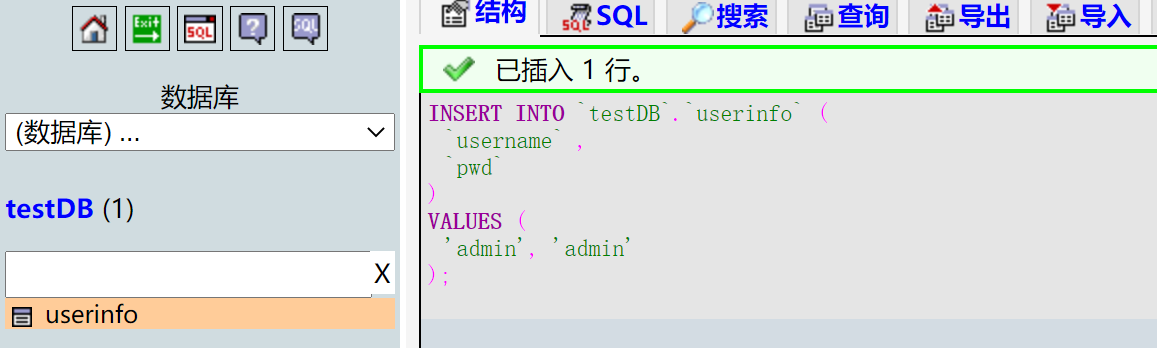
1. **点击phpMyAdmin，用户名密码为root/123456，登录进数据库管理系统。创建一个新的数据库：**



1. **新建表userinfo，属性为username与pwd，点击保存：**



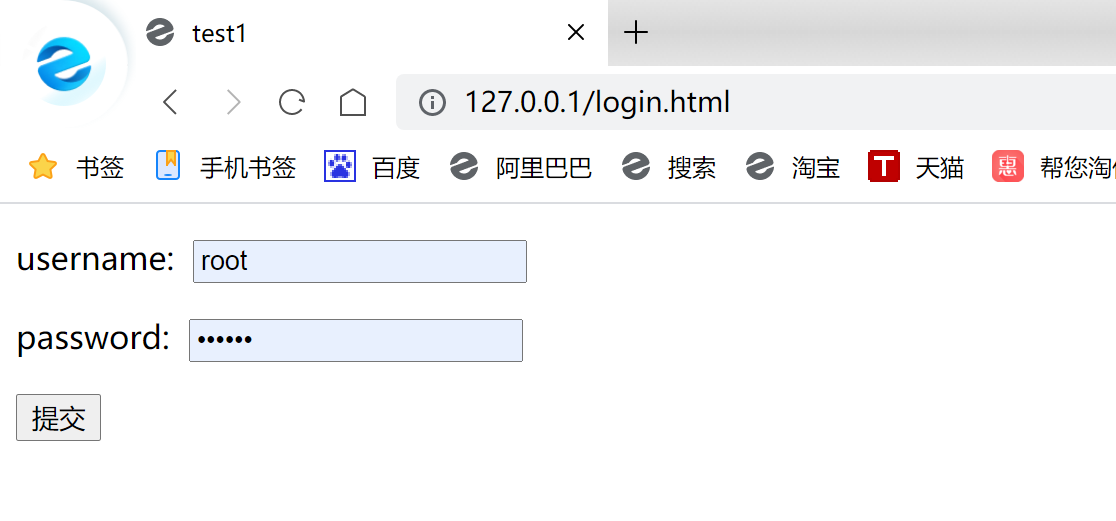
1. **插入数据，admin/admin**



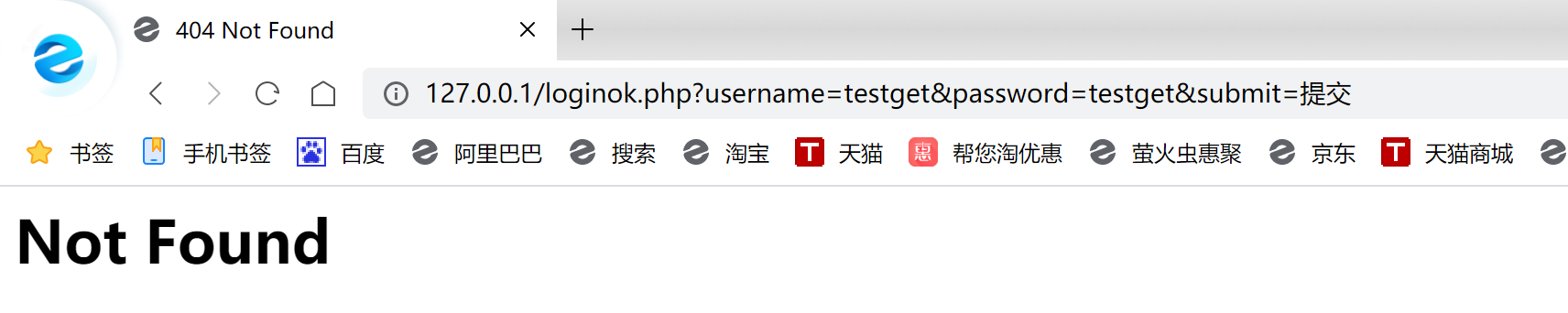
1. **使用DW软件创建html文件，保存在hpdocs文件夹下。修改html文件文档。添加表单form1，添加action，添加表单对象为文本。添加username，password文本以及提交按钮。具体代码如下：**

|  |
| --- |
| **<!doctype html>**  **<html>**  **<head>**  **<meta charset="utf-8">**  **<title>test1</title>**  **</head>**  **<body>**    **<form action="loginok.php" method="get" name="form1" id="form1">**  **<p>**  **<label for="password">username:&nbsp;</label>**  **<input type="text" name="username" id="username">**  **</p>**  **<p>**  **<label for="password">password:&nbsp;</label>**  **<input type="password" name="password" id="password">**  **</p>**  **<p>**  **<input type="submit" name="submit" id="submit" value="提交">**  **</p>**    **</form>**  **</body>**  **</html>** |

**实现效果为：**



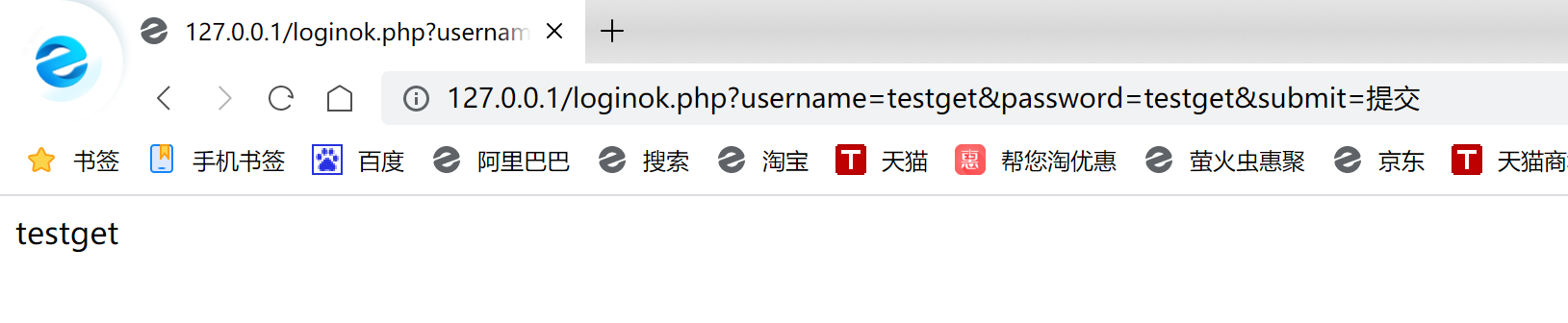
1. **设置username为testget，密码为testget得出如下效果：显然get模式会将具体的值显示在URL中。**



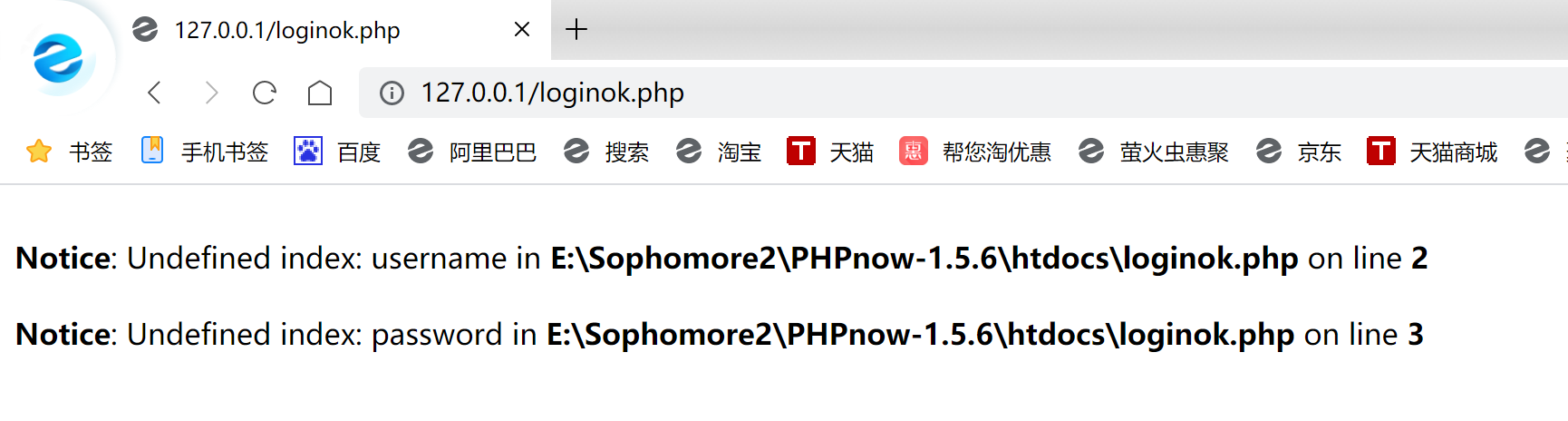
1. **在htdocs文件夹下新建login.php文件，打开并编辑代码如下：**

|  |
| --- |
| **<?php**  **$username=$\_GET['username'];变量username获取标签为username的内容，由于html文件中方式为get，故此处可以得到。**  **$password=$\_GET['password'];**  **echo $username;输出变量username的值**  **?>** |

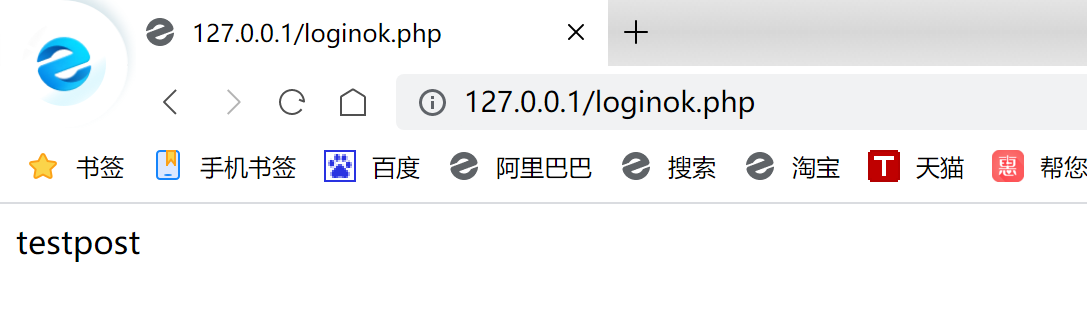
1. **再次刷新网页即可得到如下结果，输出了我们定义的username testget**



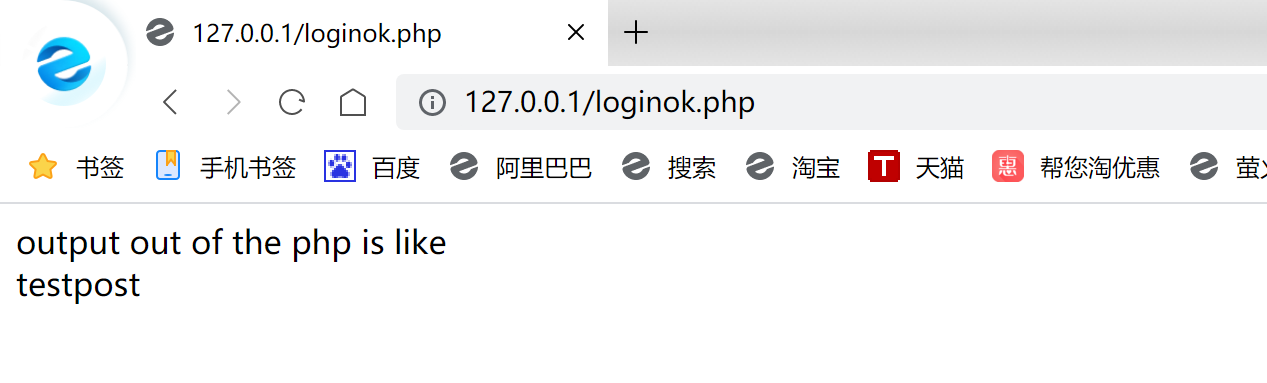
1. **修改html文件中方式为post，再次回到登陆页面并输入用户名密码为testpost/testpost，结果如下：**



1. **再次将php中获取方式改为POST方式得结果：**



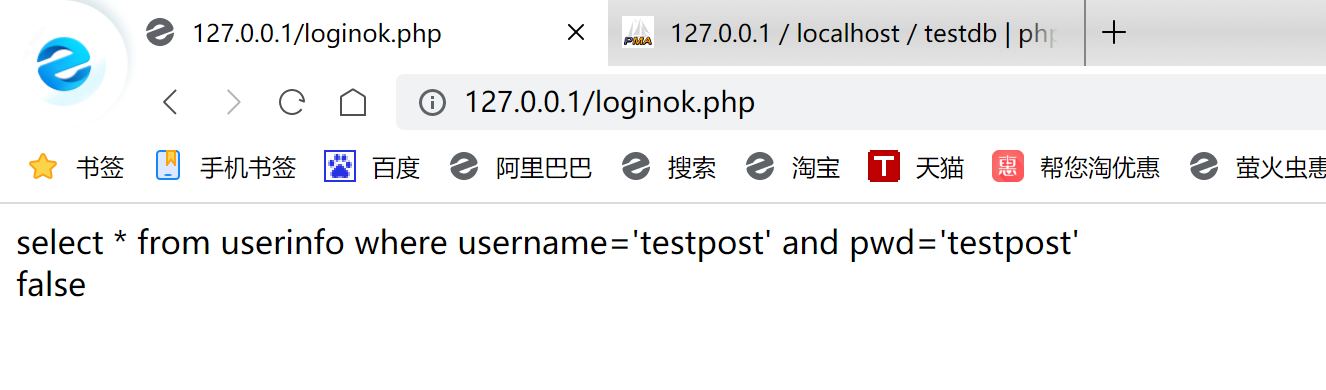
1. **在php？范围外的文本将被直接输出：**



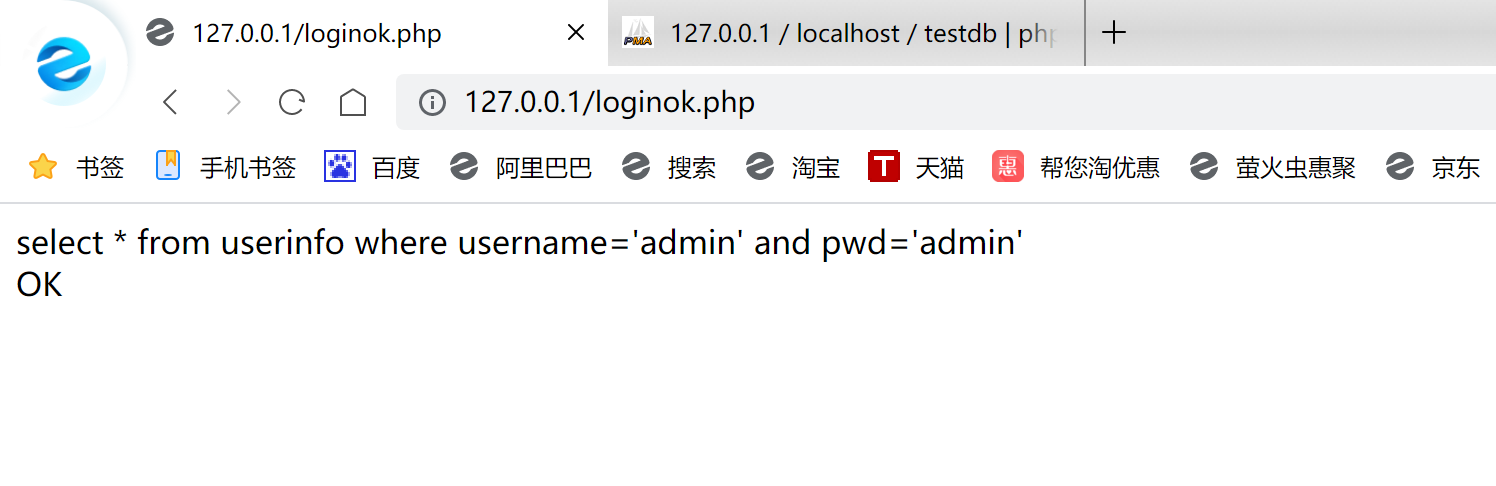
1. **添加构造SQL语句如下：**

|  |
| --- |
| **<?php**  **$conn=mysql\_connect("localhost","root","123456");//连接数据库**  **$username=$\_POST['username'];**  **$password=$\_POST['password'];**  **$SQLstr="select \* from userinfo where username='$username' and pwd='$password'";**  **echo $SQLstr;**  **$result=mysql\_db\_query("testDB",$SQLstr,$conn);//执行数据库语句**  **//获取查询结果**  **if($row=mysql\_fetch\_array($result))**  **echo "<br>OK<br>";**  **else**  **echo "<br>false<br>";**  **//释放资源**  **mysql\_free\_result($result);**  **//关闭连接**  **mysql\_close($conn);**  **?>** |

**结果如下：**



**若输入之前插入的数据admin/admin，结果为：**



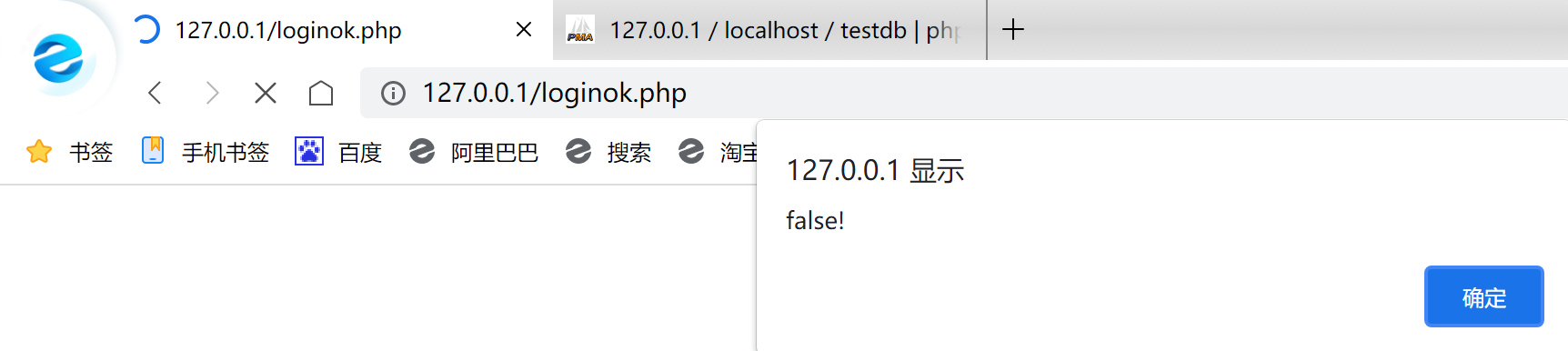
1. **修改代码实现以下功能：输入正确的用户名密码跳转至sys.html界面，否则回到历史界面**

**代码为：**

|  |
| --- |
| **<?php**  **$isok=0;**  **$conn=mysql\_connect("localhost","root","123456");//连接数据库**  **$username=$\_POST['username'];**  **$password=$\_POST['password'];**  **$SQLstr="select \* from userinfo where username='$username' and pwd='$password'";**  **echo $SQLstr;**  **$result=mysql\_db\_query("testDB",$SQLstr,$conn);//执行数据库语句**  **//获取查询结果**  **if($row=mysql\_fetch\_array($result))**  **$isok=1;**  **//释放资源**  **mysql\_free\_result($result);**  **//关闭连接**  **mysql\_close($conn);**  **if($isok==1)**  **{**  **?>**  **<script language="javascript">**  **alert("ok!");//弹出对话框**  **window.location.href="sys.php";//跳转至该页面**  **</script>**  **<?//使用问号打断和重新开始，使得两端之间的代码不解释运行**  **}**  **else**  **{**  **?>**  **<script language="javascript">**  **alert("false!");**  **history.back();//回去历史界面**  **</script>**  **<?//使用问号打断和重新开始**  **}**  **?>** |

**结果如下：**





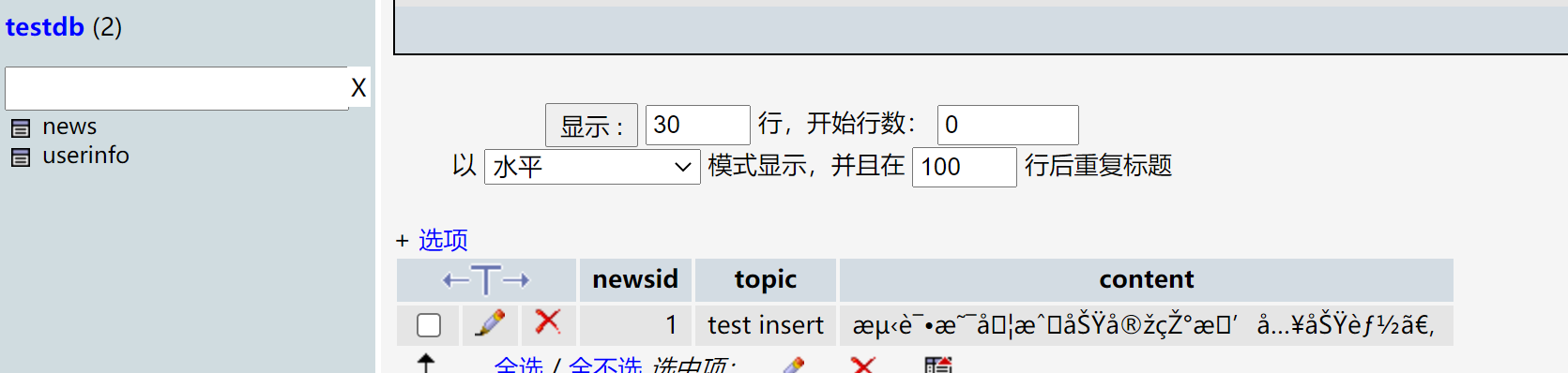
1. **在testdb数据库中新建news表，属性为newsid，topic，content。新建sys.php页面，布局为topic为标签，一个文本输入框，content为标签，一个文本区域用于输入。具体代码如下：**

|  |
| --- |
| **<!doctype html>**  **<html>**  **<head>**  **<meta charset="utf-8">**  **<title>test1</title>**  **</head>**  **<body>**    **<form action="addok.php" method="post" name="form1" id="form1">**  **<p>**  **<label for="password">topic:&nbsp;</label>**  **<input type="text" name="topic" id="topic">**  **</p>**  **<p>**  **<label for="password">content:&nbsp;</label>**  **<br>**  **<label for="content">&nbsp;</label>**  **<textarea name="content" cols="60" rows="8" id="content"></textarea>**  **</p>**  **<p>**  **<input type="submit" name="submit" id="submit" value="提交">**  **</p>**    **</form>**  **</body>**  **</html>** |

**新建addok.php文件，实现功能为对news表实现插入功能，其具体功能为读取sys.php页面中获取的topic和content值，并且插入到news表中。实现代码如下：**

|  |
| --- |
| **<?php**  **$conn=mysql\_connect("localhost","root","123456");//连接数据库**  **mysql\_select\_db("testDB");**  **$topic=$\_POST['topic'];**  **$content=$\_POST['content'];**  **$SQLstr="insert into news(topic,content) values('$topic','$content')";**  **echo $SQLstr;**  **$result=mysql\_query($SQLstr);//执行数据库语句**  **//关闭连接**  **mysql\_close($conn);**  **if($result)//执行操作是否成功**  **{**  **?>**  **<script language="javascript">**  **alert("add ok!");//弹出对话框**  **window.location.href="sys.php";//跳转至该页面**  **</script>**  **<?//使用问号打断和重新开始，使得两端之间的代码不解释运行**  **}**  **else**  **{**  **?>**  **<script language="javascript">**  **alert("add false!");**  **history.back();//回去历史界面**  **</script>**  **<?//使用问号打断和重新开始**  **}**  **?>** |

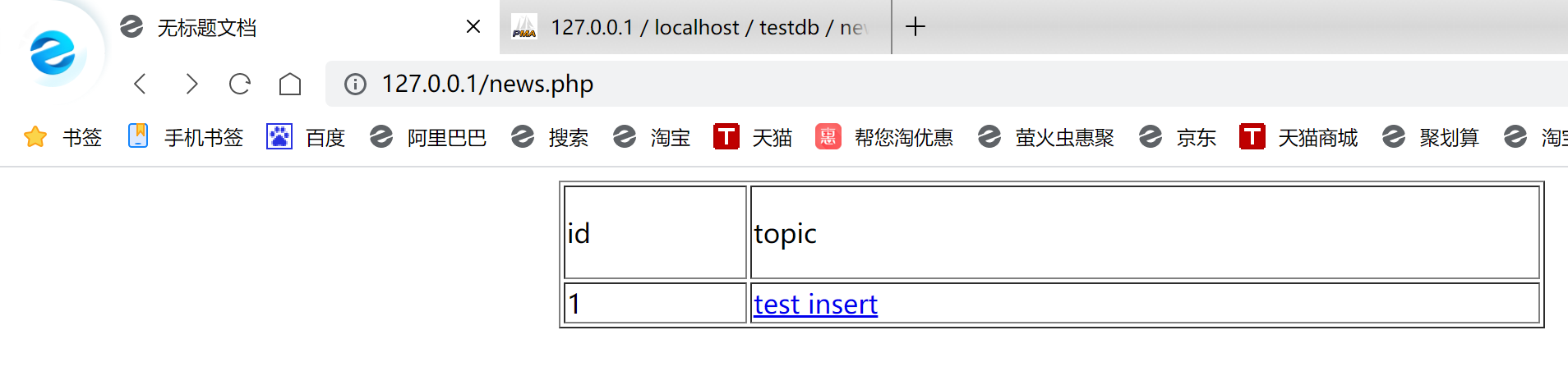
**在页面中成功登陆后即可实现插入操作：**



**（此处content乱码是因为输入为中文。）**

1. **新建html文件，另存为为news.php，实现功能为显示出news表中所存信息的id和topic，在topic处显示链接。具体实现代码如下：**

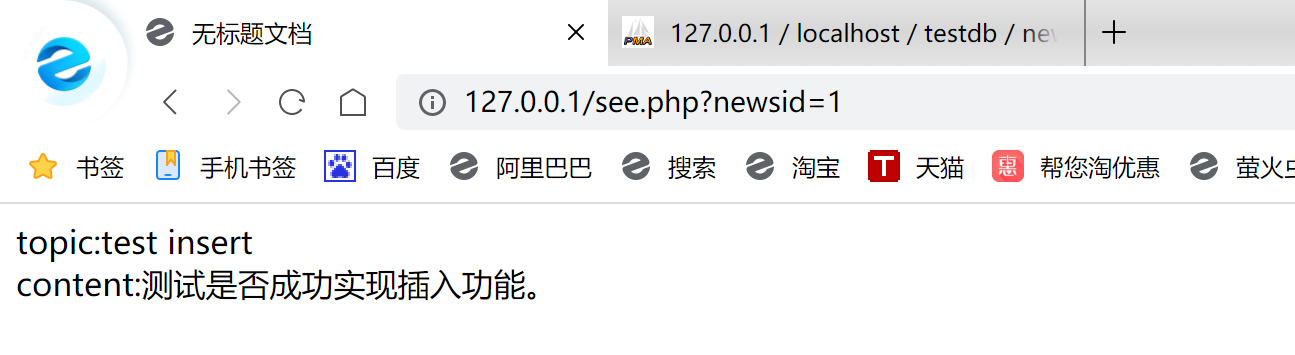
|  |
| --- |
| **<!doctype html>**  **<html>**  **<head>**  **<meta charset="utf-8">**  **<title>无标题文档</title>**  **</head>**  **<body>**    **<table width="600" border="1" align="center">**  **<tr>**  **<td><p>id</p></td>**  **<td>topic</td>**  **</tr>**  **<?php**  **$conn=mysql\_connect("localhost","root","123456");//连接数据库**  **$SQLStr="select newsid,topic 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**  **while($row=mysql\_fetch\_row($result))**  **{**      **?>**  **<tr>**  **<td><?php echo $row[0];?></td>**    **<td><a href="see.php?newsid=<?php echo $row[0];?>">**  **<?php echo $row[1];?>**  **</a></td>**  **</tr>**  **<?php**    **}**  **}**  **?>**  **</table>**  **</body>**  **</html>** |



1. **新建see.php，实现对该信息的查看。**

**代码如下：**

|  |
| --- |
| **<!doctype html>**  **<html>**  **<head>**  **<meta charset="utf-8">**  **<title>无标题文档</title>**  **</head>**  **<body>**    **<?php**  **$conn=mysql\_connect("localhost","root","123456");//连接数据库**  **$id=$\_GET['newsid'];**  **$SQLStr="select topic,content from news where newsid= '$id'";**  **$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($result,0);**  **//循环取出**  **while($row=mysql\_fetch\_row($result))**  **{**  **?>**  **topic:<?php echo $row[0];?><br>**  **content:<?php echo $row[1];?><br>**  **<?php**  **}**  **}**  **?>**  **</body>**  **</html>** |



1. **再次插入一条语句，topic为test delete，content为测试是否成功实现删除功能。**



**修改addok.php中sql语句为删除，代码变为：**

|  |
| --- |
| **<?php**  **$conn=mysql\_connect("localhost","root","123456");//连接数据库**  **mysql\_select\_db("testDB");**  **$topic=$\_POST['topic'];**  **$content=$\_POST['content'];**  **$SQLstr="delete from news where news.topic='$topic';";**  **echo $SQLstr;**  **$result=mysql\_query($SQLstr);//执行数据库语句**  **//关闭连接**  **mysql\_close($conn);**  **if($result)//执行操作是否成功**  **{**  **?>**  **<script language="javascript">**  **alert("delete ok!");//弹出对话框**  **window.location.href="sys.php";//跳转至该页面**  **</script>**  **<?//使用问号打断和重新开始，使得两端之间的代码不解释运行**  **}**  **else**  **{**  **?>**  **<script language="javascript">**  **alert("add false!");**  **history.back();//回去历史界面**  **</script>**  **<?//使用问号打断和重新开始**  **}**  **?>** |

**在addok界面的topic输入delete，结果如下：**



**可以看出语句被删除，删除语句测试成功。**

**心得体会：**

**学习了网站的初步构造，了解了php语言的编写和网页的布局。**