****

Web系统与技术

实 验 报 告

|  |  |
| --- | --- |
| **学 院** | 网络空间安全学院 |
| **专 业** | 网络工程 |
| **班 级** | 19272401 |
| **学 号** | 19061440 |
| **学生姓名** | 张逸轩 |
| **教师姓名** | 胡伟通 |
| **完成日期** | 2021.04.27 |
| **成 绩** |  |

**作业2 数据库访问**

1. **实验目的**
2. 根据相应的Web开发语言来实践以代码方式连接访问数据库
3. **实验内容**

自己选择一个应用主题（比如：图书馆管理系统），并用PHP语言和mysqli库实现**以代码的方式**来连接数据库、建立新数据库表（至少3张数据库表）、对数据库表进行**增删改查**等操作。

提示：

1. 可以参考网络资料，但不能照搬照抄。

2. 作业报告的格式尽量整齐，字体要统一，内容要层次分明。

1. **程序设计思路**

在PHP脚本中访问MySQL数据库的八个步骤如下：

1. 连接MySQL数据库服务器并选择数据库

2. 判断是否连接正确

3. 设置字符集

4. 准备SQL语句

5. 发送SQL语句到MySQL服务器

6. 处理结果集

7. 释放结果集资源

8. 关闭数据库连接

——引自课堂教学PPT

其中，发送SQL语句到MySQL服务器是关键也是经典的部分，因此，对于SQL语句的掌握程度是目标能否实现的重要组成部分。

1. **程序源代码**

**注意源代码要有详细的注释。 同学们提交的每个程序都应该遵循Honor Code（诚实代码保证）的要求。**

**请大家特别注意一定要在每个程序首部的注释中加上以下保证：**

**# 我真诚地保证：**

**# 我自己独立地完成了整个程序从分析、设计到编码的所有工作。**

**# 如果在上述过程中，我遇到了什么困难而求教于人，那么，我将在程序实习报告中**

**# 详细地列举我所遇到的问题，以及别人给我的提示。**

**# 在此，我感谢 XXX, …, XXX对我的启发和帮助。下面的报告中，我还会具体地提到**

**# 他们在各个方法对我的帮助。**

**# 我的程序里中凡是引用到其他程序或文档之处，**

**# 例如教材、课堂笔记、网上的源代码以及其他参考书上的代码段,**

**# 我都已经在程序的注释里很清楚地注明了引用的出处。**

**# 我从未没抄袭过别人的程序，也没有盗用别人的程序，**

**# 不管是修改式的抄袭还是原封不动的抄袭。**

**# 我编写这个程序，从来没有想过要去破坏或妨碍其他计算机系统的正常运转。**

**# 张逸轩**

1. **网上参考代码**





1. **源代码**

<?php

//实例 (MySQLi - 面向对象)创建数据库连接

echo "实例 (MySQLi - 面向对象)<br/>";

$servername = "127.0.0.1";

$username = "root";

$password = "root";

date\_default\_timezone\_set("Asia/Shanghai");

echo date('Y-m-d H:i:s',strtotime('now'));

// 创建连接

$conn = new mysqli($servername, $username, $password);

// 检测连接

if ($conn->connect\_error) {

die("连接失败: " . $conn->connect\_error);

}

// 创建数据库myDB

$sql = "CREATE DATABASE myDB";

if ($conn->query($sql) === TRUE) {

echo "数据库创建成功";

} else {

echo "<mark><br/>Error creating database: " . $conn->error."</mark>";

echo "<br/>";

}/\* 注意： 当你创建一个新的数据库时，你必须为 mysqli 对象指定三个参数 (servername, username 和 password)。

Tip: 如果你使用其他端口（默认为3306），为数据库参数添加空字符串，指定端口号

如: new mysqli("localhost", "username", "password", "", port)\*/

echo "连接成功<br/>";

$conn->close();//关闭连接

echo "关闭连接<br/>";

?>

<?php

//实例 (MySQLi 面向对象)创建表格

//mysql\_query("set names 'utf8'");

date\_default\_timezone\_set("Asia/Shanghai");

echo date('Y-m-d H:i:s',strtotime('now'));

echo "<br/>";

$servername = "127.0.0.1";

$username = "root";

$password = "root";

$dbname = "myDB";

// 创建连接

$conn = new mysqli($servername, $username, $password, $dbname);

// 检测连接

if ($conn->connect\_error) {

die("连接失败: " . $conn->connect\_error);

}

else

{

echo '连接成功！';

}

// 使用 sql 创建数据表MyGuests

$sql = "CREATE TABLE MyGuests (

id INT(6) UNSIGNED AUTO\_INCREMENT PRIMARY KEY,

firstname VARCHAR(30) NOT NULL,

lastname VARCHAR(30) NOT NULL,

email VARCHAR(50)

)";

echo "<mark>";

if ($conn->query($sql) === TRUE) {

echo "Table MyGuests created successfully";

}

else {

echo "创建数据表错误: " . $conn->error;

}

echo "</mark>";

$sql = "INSERT INTO MyGuests( `firstname`, `lastname`, `email`) VALUES ('Hangzhou', 'Zhejiang', 'hz@zj.com')";

$conn->query($sql);

printf ("New Record has id %d.\n", $conn->insert\_id);

//连接

$conn->close();

?>

由于一开始使用了这段代码进行相应的调试，因此不断地重复导致id号不断地增加，并且增添了很多重复字段，下次是需要注意的。其原因是由于主键未设置成为不能为空，因此产生了很多的无效操作

因此，在明确了实验的目标后，重新修改相应的代码进行测试。

**建立数据库**

<?php

//实例 (MySQLi - 面向对象)创建数据库连接

echo "实例 (MySQLi - 面向对象)<br/>";

$servername = "127.0.0.1";

$username = "root";

$password = "root";

date\_default\_timezone\_set("Asia/Shanghai");

echo date('Y-m-d H:i:s',strtotime('now'));

// 创建连接

$conn = new mysqli($servername, $username, $password);

// 检测连接

if ($conn->connect\_error) {

die("连接失败: " . $conn->connect\_error);

}

// 创建数据库myDB

$sql = "CREATE DATABASE MyDataBase\_Perform";

if ($conn->query($sql) === TRUE) {

echo "数据库创建成功";

} else {

echo "<mark><br/>Error creating database: " . $conn->error."</mark>";

echo "<br/>";

}/\* 注意： 当你创建一个新的数据库时，你必须为 mysqli 对象指定三个参数 (servername, username 和 password)。

Tip: 如果你使用其他端口（默认为3306），为数据库参数添加空字符串，指定端口号

如: new mysqli("localhost", "username", "password", "", port)\*/

echo "连接成功<br/>";

$conn->close();//关闭连接

echo "关闭连接<br/>";

?>

**创建三个数据表，分别为演员、道具和顺序三个数据表**

<?php

//实例 (MySQLi 面向对象)创建表格

//mysql\_query("set names 'utf8'");

date\_default\_timezone\_set("Asia/Shanghai");

echo date('Y-m-d H:i:s',strtotime('now'));

echo "<br/>";

$servername = "127.0.0.1";

$username = "root";

$password = "root";

$dbname = "MyDataBase\_Perform";

// 创建连接

$conn = new mysqli($servername, $username, $password, $dbname);

// 检测连接

if ($conn->connect\_error) {

die("连接失败: " . $conn->connect\_error);

}

else

{

echo '连接成功！';

}

// 使用 sql 创建数据表Actors，存储演员信息，包括序号，名字，年龄

$sql = "CREATE TABLE Actors (

id INT(6) UNSIGNED AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(30) NOT NULL,

age VARCHAR(30) NOT NULL

)";

echo "<mark>";

if ($conn->query($sql) === TRUE) {

echo "Table Actors created successfully";

}

else {

echo "创建数据表错误: " . $conn->error;

}

echo "</mark>";

printf ("创建Actors成功！New Record has id %d.\n", $conn->insert\_id);

// 使用 sql 创建数据表Things，存储道具，包括价值及名称

$sql = "CREATE TABLE Things (

id INT(6) UNSIGNED AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(30) NOT NULL,

value VARCHAR(30) NOT NULL

)";

echo "<mark>";

if ($conn->query($sql) === TRUE) {

echo "Table Actors created successfully";

}

else {

echo "创建数据表错误: " . $conn->error;

}

echo "</mark>";

printf ("创建Things成功！New Record has id %d. \n", $conn->insert\_id);

// 使用 sql 创建数据表Orders，节目名称及序号

$sql = "CREATE TABLE Orders (

id INT(6) UNSIGNED AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(30) NOT NULL

)";

echo "<mark>";

if ($conn->query($sql) === TRUE) {

echo "Table Actors created successfully";

}

else {

echo "创建数据表错误: " . $conn->error;

}

echo "</mark>";

printf ("创建Orders数据表成功！New Record has id %d. \n", $conn->insert\_id);

//连接关闭

$conn->close();

?>

**新增相应的数据**

<?php

$host = 'localhost';

$user = 'root'; // MySQL用户账号

$pass = 'root';

$dbname = 'MyDataBase\_Perform';

$conn = mysqli\_connect($host, $user, $pass,$dbname);

if(!$conn){

die('Could not connect: '.mysqli\_connect\_error());

}

echo 'Connected successfully<br/>';

$sql1 = "INSERT INTO Actors(id, name,age) VALUES ('001', '王波','20')";

$sql2 = "INSERT INTO Things(id, name,value) VALUES ('001', '光学仪器','100')";

$sql3 = "INSERT INTO Order(id, name) VALUES ('001', '泊松亮斑')";

if ($conn->query($sql1) == TRUE)

{

echo "Actors entry saved successfully.";

}

else

{

echo "INSERT attempt failed" ;

}

if ($conn->query($sql2) == TRUE)

{

echo "Things entry saved successfully.";

}

else

{

echo "INSERT attempt failed" ;

}

if ($conn->query($sql3) == TRUE)

{

echo "Order entry saved successfully.";

}

else

{

echo "INSERT attempt failed" ;

}

mysqli\_close($conn);

?>

**修改数据**

<?php

//数据库相关初始化操作

$host = 'localhost';

$user = 'root'; // MySQL用户账号

$pass = 'root';

$dbname = 'MyDataBase\_Perform';

$conn = mysqli\_connect($host, $user, $pass,$dbname);

//连接成功与否检测

if(!$conn)

{

die('Could not connect: '.mysqli\_connect\_error());

}

else echo 'Connected successfully<br/>';

$sql3="update Actors set name='周松' where id='001'";

if ($conn->query($sql3))

{

echo "修改成功\n";

}

else

{

echo "修改失败\n";

}

$sql3="update Things set name='锅铲' where id='001'";

if ($conn->query($sql3))

{

echo "修改成功\n";

}

else

{

echo "修改失败\n";

}

$sql3="update Orders set id='002' where id='001'";

if ($conn->query($sql3))

{

echo "修改成功\n";

}

else

{

echo "修改失败\n";

}

mysqli\_close($conn);

?>

**删除数据**

<?php

//初始化

$server = 'localhost';

$user = 'root';

$pass = 'root';

$dbname = 'MyDataBase\_Perform';

$conn = mysqli\_connect($server, $user, $pass,$dbname);

if(!$conn)

{

die('Could not connect: '.mysqli\_connect\_error());

}

else echo 'Connected successfully<br/>';

//判断连接状态

$sql="delete from Actors where name='周松'";

if ($conn->query($sql))

{

echo "删除成功";

}

else

{

echo "删除失败";

}

//删除检测并反馈

mysqli\_close($conn);

?>

**查看数据**

<?php

//初始化

$server = '127.0.0.1';

$user = 'root';

$pass = 'root';

$dbname = 'MyDataBase\_Perform';

$conn = mysqli\_connect($host, $user, $pass,$dbname);

//连接状态检测

if(!$conn)

{

die('Could not connect: '.mysqli\_connect\_error());

}

else echo 'Connected successfully<br/>';

$sql = 'select \* from Things';

$query = $conn->query($sql);

while($row = $query->fetch\_array())

{

echo $row['name'],'<br>';

}

$sql = 'select \* from Actors';

$query = $conn->query($sql);

while($row = $query->fetch\_array())

{

echo $row['name'],'<br>';

}

$sql = 'select \* from Orders';

$query = $conn->query($sql);

while($row = $query->fetch\_array())

{

echo $row['name'],'<br>';

}

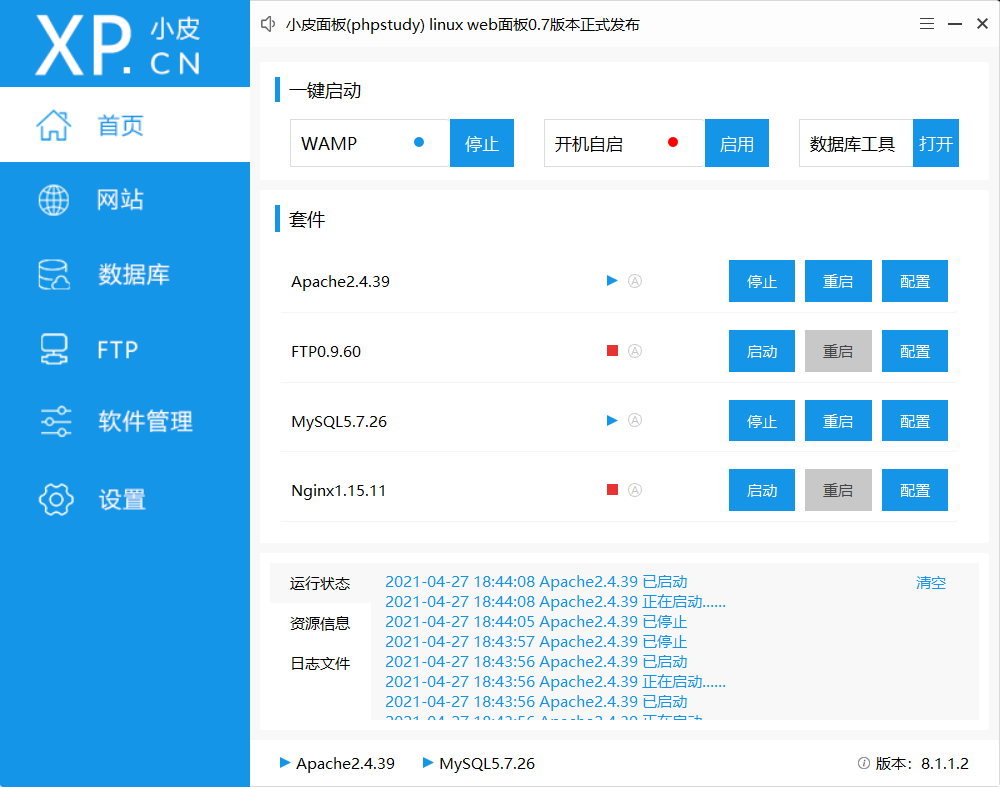
$query->close();

mysqli\_close($conn);

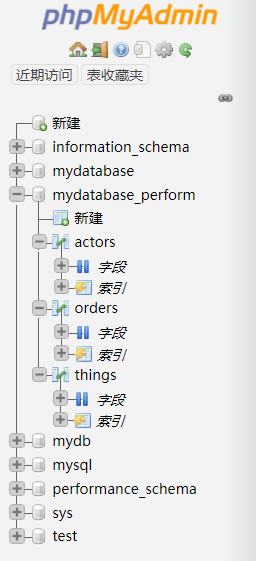
?>

1. **程序测试方法及测试结果记录（不能光截图，要有相应的文字说明）**
2. **测试方法**

**在PHPstudy中开启阿帕奇服务器和mysql，在数据库工具中打开phpmyadmin，可以查看相关的数据库、数据表及相关信息的存储情况。**



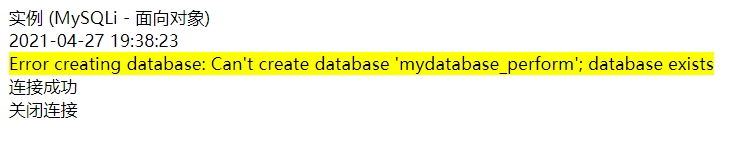
**查询后得到如下的情况**



**接下来的测试流程，将会逐步进行代码的运行，即上述的代码。**

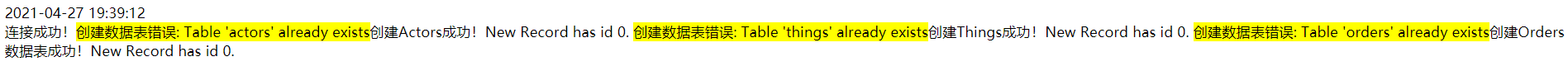
1. **测试流程**

创建数据库



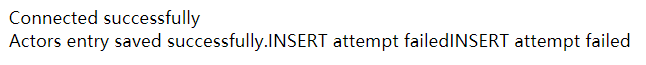
由于已经存在，报错，此时不管

创建三个数据表



同样的，已经存在因此报错，不需要管

按照上述的新增部分的代码，得到如下结果：



原因是因为Actors中的数据最后被删除，而后两个数据表中的数据仍然已经存在，因此插入失败







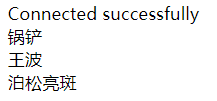
可以看出，三个数据表中均存在着相应的数据

接下来测试更改的代码



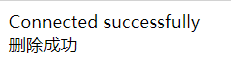
修改同样失败，也是因为存在相应的数据，在phpadmin中查看可知

接下来测试查看的代码



得到如上图所示的结果

接下来测试删除的代码





数据表中删除成功

1. **实验分析总结及心得（该部分也是评分的一个重点）**

（结合所学知识对实验过程中观察到的实验结果进行分析总结，以便加深对知识的理解，并总结通过实验学到的知识或技术）

特别推荐学生写出做实验遇到的问题，以及从原理分析得到解决方案的过程。

本次实验中遇到了许多的问题，但是由于发生错误后更多的反应是解决问题而不是记录问题，因此没能有相应的图片记录。遇到的问题更多是因为配置上的问题。例如常常出现80端口被占用的问题，网上查询了相关记录后，明白了可以通过cmd中的netstat命令去查找有什么进程占用了80端口，查找pid后调用任务管理器，将进程停用，最后能够完成80端口的调用。

除此之外，phpstorm软件也没有安装，使用txt进行相应的代码编写不论是在写代码还是编译相应代码的方面都有着不小的低效率问题。但是好在php的相关提示信息比较人性化，相较于C/C++等老牌编译器，报错信息直接明了例如xx数据表已经存在，找不到xxx数据表，无法连接xxxx，这对于我们查找调试相应的信息有着非常好的帮助。

建立在本次实验之上，以后将会养成及时记录问题的习惯，需要边做实验边做记录的习惯。这是在方法论上的重要知道。此外，还需要查看更多的书籍，由于上课更多依靠ppt，所以课后的教材也需要进行相应的了解。相信二者相结合，能够有着更多的解决方法。