

2015-2016 中山大学本科生实验报告

科目：数据库系统实验

教师：郑贵锋

年级	13 级	专业（方向）	移动信息工程
学号	13354485	姓名	朱琳
电话	13726231932	Email	<u>280273861@qq.com</u>

一 实验目的

1. 了解 Mysql。
2. 学会使用 PHP 来操作 MySQL。

二 实验内容

I.实验题目

使用 mysql，实现一个 HTML 页面输入 Mysql 指令，php 对数据库进行操作。利用该页面完成以下问题：

- 1.创建一个数据库，week13。（操作结果显示在命令行中，截图）
- 2.创建如下表格：lab9，

Id	Name	Phone
1	Peter	2333
2	John	3222
3	Pan	2323

其中数据在 html 页面中使用 insert 插入，创建得到的结果在一个新页面中输出，如图所示。

```
select * from stucent;
week13
```

SUBMIT

id	name	phone
1	Peter	3323

[BACK](#)

- 3.任意修改某条数据，将修改后的表格截图。
- 4.任意删除某条数据，将删除后的表格截图。

II.课后作业

- 1.什么叫做 SQL Injection(SQL 注入)。
- 2.这次作业存在 SQL Injection 的可能吗？
- 3.POST 和 GET 两种方式有什么区别？
- 4.在 PHP 中什么是 PDO？

三 实验结果

1.创建一个数据库，**week13**。（操作结果显示在命令行中，截图）

(1)html 中操作结果

create database week13;
DATABASE

SUBMIT

Query OK
[Back](#)

(2)命令行结果:

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| artbase      |
| lab7         |
| labfour     |
| mysql       |
| performance_schema |
| school      |
| sysu        |
| test        |
| test2       |
| week12      |
| week13      |
| wwwwww      |
+-----+
13 rows in set (0.00 sec)
```

2.创建如下表格：**lab9**

(1)操作:

create table lab9(id int,name varchar(20),phone varchar(20));
week13

SUBMIT

insert into lab9 values(1,"Peter","2333");
week13

SUBMIT

insert into lab9 values(2,"John","3222");
week13

SUBMIT

insert into lab9 values(3,"Pan","2323");
week13

SUBMIT

结果均为:

Query OK

[Back](#)

(2)现在查看插入情况:

```
select * from lab9;
```

week13

SUBMIT

(3)点击 submit 之后:

id	name	phone
1	Peter	2333
2	John	3222
3	Pan	2323

[Back](#)

3.任意修改某条数据，将修改后的表格截图。

(1)html 运行:

```
update lab9 set name="zhulin" where id=2;
```

week13

SUBMIT

(2)执行结果:

Query OK

[Back](#)

(3)查看执行结果

```
select * from lab9;
```

week13

SUBMIT

id	name	phone
1	Peter	2333
2	zhulin	3222
3	Pan	2323

[Back](#)

4.任意删除某条数据，将删除后的表格截图。

(1)html 执行：

delete from lab9 where id=3;
week13

SUBMIT

(2)执行结果：

Query OK
[Back](#)

(3)查看执行结果：

select * from lab9;
week13

SUBMIT

id	name	phone
1	Peter	2333
2	zhulin	3222

[Back](#)

四 课后作业解答

1. 什么叫做 SQL Injection(SQL 注入)。

SQL 注入攻击是黑客对数据库进行攻击的常用手段之一。用户可以提交一段数据库查询代码，根据程序返回的结果，获得某些他想得知的数据，这就是所谓的 SQL Injection。SQL 注入是从正常的 WWW 端口访问，而且表面看起来跟一般的 Web 页面访问没什么区别，所以目前市面的防火墙都不会对 SQL 注入发出警报，如果管理员没查看 IIS 日志的习惯，可能被入侵很长时间都不会发觉。但是，SQL 注入的手法相当灵活，在注入的时候会碰到很多意外的情况，需要构造巧妙的 SQL 语句，从而成功获取想要的数据库。

2. 这次作业存在 SQL Injection 的可能吗？

存在。很明显，我们并没有对 html 中输入的字符串做任何的处理，并且我们使用安全性较低的 GET 方法，安全性大大下降。

3. POST 和 GET 两种方式有什么区别？

	GET	POST
方式	从服务器上获取数据	向服务器传送数据。
过程	get 是把参数数据队列加到提交表单的 ACTION 属性所指的 URL 中，值和表单内各个字段一一对应，在 URL 中可以看到	post 是通过 HTTP post 机制，将表单内各个字段与其内容放置在 HTML HEADER 内一起传送到 ACTION 属性所指的 URL 地址。用户看不到这个过程。
服务器处理方式	服务器端用 Request.QueryString 获取变量的值	服务器端用 Request.Form 获取提交的数据。
数据传送量	get 传送的数据量较小，不能大于 2KB	post 传送的数据量较大，一般被默认为不受限制。
安全性	get 安全性非常低，但是执行效率却比 Post 方法好	post 安全性较高
数据集	限制 Form 表单的数据集的值必须为 ASCII 字符	支持整个 ISO10646 字符集。
备注	1、get 方式的安全性较 Post 方式要差些，包含机密信息的话，建议用 Post 数据提交方式； 2、在做数据查询时，建议用 Get 方式；而在做数据添加、修改或删除时，建议用 Post 方式； 3. GET 是 Form 默认方式	

4.在 PHP 中什么是 PDO？

PHP Data Object,PHP 数据对象，PDO 扩展为 PHP 访问数据库定义了一个轻量级的、一致性的接口，它提供了一个数据访问抽象层，这样无论使用什么数据库，都可以通过一致的函数执行查询和获取数据，提高了数据库的安全性。

相比于 Mysql 和 mysqli，其更具有安全性，简便性。

①PDO 连接数据库时通过 Connection String 来决定连接何种数据库。

②PDO 可以通过 PDO::setAttribute 来决定连接时的设定，比如 Persistent Connection，回传错误的方式（Exception，E_WARNING，NULL）。甚至是回传参数名称的大小写等等。

③PDO 支持 Bind Column 的功能，除了基本的 Prepare，Execute 以外，也可以 Bind 单一参数，并且指定参数类型。

④PDO 是 Abstraction Layer，所以就算更换储存媒介，需要花的功夫比起来是最少的。

五 实验感想

1.本次实验，主要是使用 mysqli，在网页中输出想要的结果，其中操作数据库的操作需要表单来完成。不过之前对于表单接触比较少，因此用起来也没有那么得心应手，不过好在 TA 已经把 html 文件给出，我也就没有这个烦恼了。

2.不过最开始我想通过 JavaScript 来控制数据的运输什么的（上周我是这样做的），后来发现麻烦，才知道 TA 已经给了一个 html。

3.一说到正则表达式，我就想起了 java 的正则表达式，然而运行之后发现并不行，我才知道 php 的正则跟 java 很不一样呀。需要用 preg_match("/select.+/", \$sql) 这种形式。而 java 中值

需要`$sql="select%"`什么的就行了。

4.然后在网页上输出一个 `table` 什么的，因为 `jsp` 实验刚做了，也就没什么难度了。