

实验报告

一、实验目的：

- 1、掌握 PHP 开发过程与网页前后台工作原理
- 2、掌握 php 获取 html 表单数据的方法
- 3、掌握类的创建方法及应用
- 4、掌握正则表达式的使用方法

二、实验要求：

- 1、开发 PHP 架构的网页
- 2、用 PHP 语言实现学校实验系统。

三、实验各模块功能：

- 1、系统功能：实现实验课程的管理、学生课程查询、学生信息管理、教师学生个人信息修改。

2、实验环境：

PHP 版本：PHP7.3.6

MySQL 版本：5.7

Apache 版本：2.4.39（Unix）

3、电脑配置：

CPU：i7

内存：4G

虚拟机操作系统：Linux，ubuntu16.04 版本

操作系统：win10

4、数据库表包括：

学生签到表

Q_C_id 课程号 varchar(22) primary key

Q_S_id 学号 varchar(22) foreign key

Q_sign1 是否签到 tinyint(1)

Q_sign2 是否签到 tinyint(1)

Q_sign3 是否签到 tinyint(1)

Q_sign4 是否签到 tinyint(1)

Q_count 签到次数 tinyint(1)

Q_a 第几次实验课 int(11)

学生信息表

S_id 学号 varchar(22) primary key

S_name 姓名 varchar(22)

S_depart 部门 varchar(22)

S_sex 性别 varchar(22)

S_passwd 密码 varchar(22)

教师信息表、

T_id 教师号 varchar(22)

T_name 教师名 varchar(22)

T_depart 教师部门 varchar(22)

T_passwd 教师密码 varchar(22)

T_email 教师邮箱 varchar(22)
T_profession 职位 varchar(22)
课程信息表
C_id 课程号 varchar(22)
C_name 课程名 varchar(22)
C_start_time 课程开始时间 varchar(22)
C_over_time 课程结束时间 varchar(22)
C_room 课程占用教室 varchar(22)
C_ip 课程占用 ip 地址 varchar(22)
C_department 课程所属部门 varchar(22)

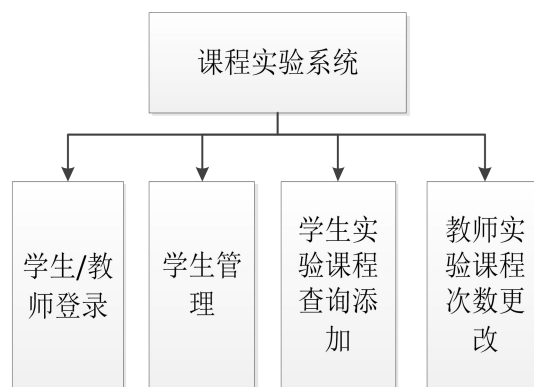


图 1 模块分布图

教师可以管理学生，学生登录可以查看课程上课时间位置以及

四、系统截图及关键代码

1、登录页面



图 2 登录页面

登录页面分为 2 个，一是教师登录页面，另一个是学生登录页面，实现该功能需要连接数据库文件，文件名为 'dbconnect.php'，数据库名为 'sign'。

1.1dbconnect.php 代码如下：

数据库连接文件，用于与 mysql 数据进行连接，并进行查询、更新、删除等数据库 sql 操作，然后选择网页需要连接的数据库名 'sign'。

```

<?php
ini_set("display_errors", "Off");//将错误隐藏
error_reporting(E_ALL | E_STRICT);
$conn=mysql_connect("localhost","root","")or die("不能连接数据库:".mysql_error());//连接数
  
```

数据库

```
mysql_query("SET NAMES 'utf-8'");//设置编码格式
```

```
mysql_select_db("sign",$conn)or die("不能选择数据库:".mysql_error());//选择数据库
```

```
?>
```

1.2 学生登录关键代码:

登录模块基本思想: 将查询到的结果行数返回, 若查询结果行数等于 1, 则说明当前用户可以登录, 否则用户名或者密码错误, 不予登录, 登录成功后要跳转至主页面, 登录失败后要跳转至登录页面。

```
<?php
```

```
//PHP 与 HEML 混编模式
```

```
require('dbconnect.php');
```

```
?>
```

```
<?php
```

```
//从数据库中取出每条留言, 按 ID 序号逆序排列,只取出一页的查询结果数
```

```
$pswd=$_POST["pswd"];
```

```
$sql="select * from student where S_id='$access' and S_passwd='$pswd'";
```

```
$result=mysql_query($sql,$conn);
```

```
$num=mysql_num_rows($result);//取得结果集中的行数
```

```
?>
```

```
<?php
```

```
if ($num>"0")//说明有查询结果, 则可以登录
```

```
{
```

```
?>
```

```
<table width="800" height="80" border="1" cellpadding="0" cellspacing="0" align="center" >
```

```
<tr>
```

```
<td align="center" height="40" colspan="4" bgcolor="#FFFFCC" ><font color="#FF0000"></font><?php echo "登入成功" ?></td>
```

```
</tr>
```

```
<tr>
```

```
<td bgcolor="#FFFFCC" align="center" bgcolor="#FF99CC"><font color="#FF0000"></font><?php echo "正在跳转到界面" ?></td>
```

```
</tr>
```

```
<tr>
```

```
</table>
```

```
<?php
```

```
echo "<script type='text/javascript' target='_parent'>";//嵌入 JavaScript
```

```
echo "function fun() {";
```

```
echo " window.location.href='S_index.html'";//要跳转到主页面
```

```
echo "};
```

```
echo "window.setTimeout('fun()',1000)";//设置延迟
```

```
echo "</script>";
```

```
}
```

```

else
{
?>
<table width="800" height="160" border="1" cellpadding="0" cellspacing="0"
align="center" >
<tr>
<td align="center" height="30" colspan="4" bgcolor="#FFFFCC" ><font
color="#FF0000">*</font><?php echo "登入失败" ?></td>
</tr>

<tr>
<td align="center" bgcolor="#FFFFCC" bgcolor="#FF99CC"><font
color="#FF0000">*</font><?php echo "请重新登入" ?></td>
</tr>
<tr>
<td>
<?php
echo "<script type='text/javascript'>";
echo "function fun() {";
echo "window.location.href='index.html'";//要重新跳转到登录页面
echo "}";
echo "window.setTimeout('fun()',1000)";//延迟 1s
echo "</script>";
}
?>

```



图 3 主页面

2、学生实验课程查询与添加

2.1、学生实验课程查询：

您当前的位置: [实验管理]-[学生签到]

课程号	课程名	开始时间	结束时间	教室	学院	操作	第几次实验
1001	操作系统	08:00:00	09:00:00	315	信息学院	签到	1
1003	计算机基础	10:00:00	11:00:00	201	信息学院	签到	1
10009	大学物理	16:00:00	19:00:00	321	信息学院	签到	2

[第1页/共1页] [共找到3个记录]

图 3 学生课程查询

查询数据库中的表 course 存储的课程信息, 将结果显示在网页中, 提示学生上课信息以及课程次数, 基本思想为将课程信息显示在表中, 当查询时, 输入三个查询条件中的任意一个都可以进行查询, 在程序中用 if 语句进行判断, 并用 sql 语句拼合的方法, 查询单一条件或组合条件的查询结果, 保证了查询条件的不唯一性。

```
<?php
//初始化 session
session_start();
//注册 session 变量
ini_set("display_errors", "Off"); //关闭警告
error_reporting(E_ALL^E_NOTICE);
require('dbconnect.php');//连接数据库
$access=$_SESSION['access'];
$sqlpage="select * from sign where Q_T_id = '$access'"; //获取对应课程信息
$re_page=mysql_query($sqlpage,$conn)or die ("查询数据失败".mysql_error());//执行查询语句
$row=mysql_fetch_array($re_page);//获得查询结果
$tnumber=$row[Q_T_id];//取出结果中 Q_T_id 的一列
?>
<html> //网页内容
<head>
<link href="css/ctop.css" rel="stylesheet" type="text/css" />
<meta http-equiv="Content-Type" content="text/html; charset='utf-8'">
<title>学生数据结果</title>
</head>
<body background="backgroud/6.jpg"> //背景图片
<div class="left_body">
<?php
//PHP 与 HTML 混编模式
$course_name=$_POST['course_name']; //保存表单中的内容
$S_name=$_POST['S_name'];
$S_num=$_POST['S_num'];
if( ($course_name=="")&&($S_name=="")&&($S_num==""))
{
    $gry=$_GET['course_name']; //如果获取不到表单中的内容, 将变量初始化为网页中的
    $name=$_GET['S_name'];
    $idnum=$_GET['S_num'];
}
```

```

$t=0;
$sql="select      C_name,      Q_S_id,S_name,Q_sign1,Q_sign2,Q_sign3,Q_sign4      from
sign,course,student  where  Q_T_id  like  '$access'and  sign.Q_C_id=course.C_id  and
sign.Q_S_id=student.S_id";//查询语句
if ($course_name!="")//若课程名不为空
{
    $sql=$sql." and Q_C_id like '$course_name'";//执行模糊查询，此句话保证了一定有查询
    结果。
    $t++;//保存查询课程的数量
}
if ($S_name!="") //若名字不为空
{
    if($t>0)//且课程查询有结果
        $sql=$sql." and S_name like '$S_name'";//继续查询他的名字，并与前面的 sql 语句合
        并，这样保证了即使两个字段为空，也能查询到结果。
    else
    {
        $sql=$sql." and S_name like '$S_name'";
        $t++;
    }
}

if ($S_num!="") //学号不为空，同理
{

    if($t>0)
        $sql=$sql." and Q_S_id like '$S_num'";
    else{
        $sql=$sql." and Q_S_id like '$S_num'";
        $t++;
    }

}

$result=mysql_query($sql,$conn);
$num=mysql_num_rows($result);//取得结果集中的行数

//定义每页显示数
$page_size=5;

//计算页数
//如果总数小于分页数，则页数为 1
if($num<=$page_size)

```

```

{
    //echo eee;
    $page_count=1;
}
//如果总留言数大于每页数，则计算总页数
if($num % $page_size)
{
    $page_count=(int)($num/$page_size)+1;
}
else
{
    $page_count=$num/$page_size;
}

//获取当前显示页数
if(isset($_GET['page']))
{
    $page=intval($_GET['page']);//intval 函数是将变量转换为 10 进制
}
else
{
    $page=1;
}
?>

```

2.2、插入学生实验课程

您当前的位置： [教师管理]-[课程选择]

请老师给每个院的学生分配对应的实验课，点击查询进行分类

课程院系：

课程ID	课程名称	开始时间	结束时间	实验学院	实验教室	实验室IP
1001	操作系统	08:00:00	09:00:00	信息学院	315	192.168.1.1
1002	数据库	09:00:00	10:00:00	数学学院	329	192.168.2.1
1003	计算机基础	10:00:00	11:00:00	信息学院	201	192.168.3.1

[第2页/共3页] [首页](#) [上一页](#) [下一页](#) [尾页](#) [共找到8个课程]

图 4 插入实验课程

将表单中输入的课程信息，通过表单提交按钮，进行插入操作。插入成功则提示插入成功，否则提示插入失败。

```

<?php $C_id=$_POST["C_id"];//获取表单中的变量内容，用作数据库的操作
    $C_name=$_POST["C_name"];
    $C_start_time=$_POST["C_start_time"];

```

```

$C_over_time=$_POST["C_over_time"];
$C_room=$_POST["C_room"];
$C_ip=$_POST["C_ip"];
$C_department=$_POST["department"];
$link=mysqli_connect("localhost","root","");
mysqli_select_db($link,"sign") or die("数据库连接失败!");
mysqli_query($link,"set names 'utf-8'");
$sql="insert
course(C_id,C_name,C_department,C_start_time,C_over_time,C_room,C_ip)
values('$C_id','$C_name','$C_department','$C_start_time','$C_over_time','$C_room','$C_ip')";
$result=mysqli_query($link,$sql);//执行 sql 语句
if($result){//返回 1， 插入成功
?>
    <form name="name0" id="name0" method="post" action="insert.php"></form>
    <script language="javascript">
        alert("插入成功");
        name0.submit();
    </script>
<?php }
    else {//插入失败
        echo "添加失败， 请重新添加";
    ?>
    <form
        name="name1"
        id="name1"
        method="post"
action="insert_course.php"></form>//表单， 提交该表单即可将课程插入。
    <script language="javascript">
        name1.submit();
    </script><br />
<?php
}
?>

```

3、学生登录管理

您当前的位置：[学生管理]-[账号管理]

【查询】

学号：	<input type="text"/>	姓名：	<input type="text"/>	学院：	<div>信息学院 ▾</div>
性别：	<input checked="" type="radio"/> 男 <input type="radio"/> 女			<input type="button" value="确定"/> <input type="button" value="重置"/>	

学号	姓名	学院	性别	操作
131401102	张三	信息学院	男	编辑 删除
131401104	赵四	信息学院	男	编辑 删除
131401105	周正	信息学院	男	编辑 删除
131401106	梅兰芳	信息学院	男	编辑 删除
131401107	小乔	信息学院	女	编辑 删除

[第1页/共2页]

图 5 查询更新删除插入学生

3.1 插入学生

利用表单将要插入学生信息填入，赋予各个变量，并将各个变量提交，分别赋给 `student` 中的各个数据项，利用 `sql` 的 `insert` 语句，插入新学生，对插入新生的学号进行唯一性判断，若存在，则提示该学生已经存在，不能进行插入操作。

关键代码：

```
<?php
    $admin_id=$_GET['admin_id'];
    $admin_name=$_GET['admin_name'];
    $admin_passwd=$_GET['admin_passwd'];
    $admin_passwd1=$_GET['admin_passwd1'];
    $admin_sex=$_GET['admin_sex'];
    $admin_bumen=$_GET['admin_bumen'];
    if ($admin_id=="" || $admin_passwd=="" || $admin_passwd1=="")
    {
        echo "内容不得为空";
        echo '<p align=center> <br><a href="student_insert.html">返回</a></p>';
        die();
    }

    if ($admin_passwd!=$admin_passwd1)
    {
        echo "两次输入密码不一致";
        echo '<p align=center> <br><a href="studenty_insert.html">返回</a></p>';
        die();
    }

    $link=mysqli_connect("localhost","root","");
    mysqli_select_db($link,"sign");
    mysqli_query("set names 'utf8'");
```

```

$sql="select admin_id from student where S_id='$admin_id'";
$result=mysqli_query($link,$sql);
$row=mysqli_fetch_array($result);
if ($row)//若查询结果不为 0，则说明当前学生已经在数据库表中。
{
    echo "此学号已经存在!\n";
    echo '<p align=center> <br><a href="admin_insert.php">返回</a></p>';
    die();//返回
}
$sql="insert into student (S_id,S_name,S_passwd,S_sex,S_depart)";
$sql=$sql."
values('$admin_id','$admin_name','$admin_passwd','$admin_sex','$admin_bumen')";
if (mysqli_query($link,$sql))
{
    echo "增加成功!";
    echo '<p align=center> <br><a href="student_main.php">返回</a></p>';

}
else
{
    echo '增加失败!';
    echo '<p align=center> <br><a href="student_main.php">返回</a></p>';
}
mysqli_close($link);
?>

```

3.2 修改学生信息

同样利用网页中的表单提交功能，将表单中输入的数据项，提交给变量，并将各个变量中的值进行更新，利用 sql 中的 update 语句更新修改后的学生信息，对应数据库中的表为：student。

```

<?php
    $admin_account=$_GET['admin_account'];//从表单中获取数据
    $admin_account0=$_GET['admin_account0'];
    $admin_name=$_GET['admin_name'];
    $admin_passwd=$_GET['admin_passwd'];
    $admin_passwd1=$_GET['admin_passwd1'];
    $admin_sex=$_GET['admin_sex'];
    $admin_bumen=$_GET['admin_bumen'];

    if ($admin_account=="" || $admin_passwd=="" || $admin_passwd1=="")
    {
        echo "内容不得为空";
        echo '<p align=center> <br><a href="student_main.php">返回</a></p>';
        die();
    }

```

```

if ($admin_passwd!=$admin_passwd1)
{
    echo "两次输入密码不一致";
    echo '<p align=center> <br><a href="student_main.php">返回</a></p>';
    die();
}

$link=mysqli_connect("localhost","root","");
mysqli_select_db($link,"sign");
mysql_query("set names 'utf8'");

if ($admin_account<>$admin_account0) {
    $sql="select S_id from student where S_id='$admin_account'";
    $result=mysqli_query($sql,$link);
    $row = mysqli_fetch_array($result);
    if ($row) {
        echo "此学号已经存在!";
        echo '<p align=center> <br><a href="student_main.php">返回</a></p>';
        die();
    }
}

$sql="update
                                student
                                set
S_id='$admin_account',S_name='$admin_name',S_passwd='$admin_passwd',S_depart='$admin
_bumen',S_sex='$admin_sex' where S_id='$admin_account0'";

if (mysqli_query($link,$sql))
{
    echo "修改成功!";
    echo '<a href="student_main.php">返回</a></p>';
}
else
{
    echo '修改失败!';
    echo '<p align=center> <br><a href="student_main.php">返回</a></p>';
}
mysqli_close($link);
?>

```

3.3 删除学生信息

根据学号对新生进行删除操作，对删除操作进行提示。

```

<?php
    $admin_account=$_GET['admin_id'];
    $link=mysqli_connect("localhost","root","");

```

```

mysqli_select_db($link,"sign");
mysql_query("set names 'utf8'");
$sql="delete from student where S_id='$admin_account'";
$result=mysqli_query($link,$sql);
$rows=mysqli_num_rows($result);
$a=mysqli_query($link,$sql);

$sql="delete from sign where Q_S_id='$admin_account'";
$result=mysqli_query($link,$sql);
$rows=mysqli_num_rows($result);
$b=mysqli_query($link,$sql);

if ($a&&$b)
{
    echo "删除成功!";
    echo '<a href="student_main.php">返回</a></p>';
}
else
{
    echo '删除失败!';
    echo '<p align=center> <br><a href="student_main.php">返回</a></p>';
}
mysqli_close($link);
?>

```

4、教师管理

教师与学生功能相同，但增加了实验课的次数更新功能，即每次上课前，更新当前课程的次数，便于教师掌握实验课的进度。

您当前的位置-[签到控制]								
请老师在每次手动控制，实验课开始和即将开始的为第几次实验								
<input type="checkbox"/>	课程ID	课程名称	开始时间	结束时间	实验学院	实验教室	实验室IP	实验次数
<input type="checkbox"/>	10009	大学物理	16:00:00	19:00:00	信息学院	321	192.168.0.0	第1次实验 ▾
<input type="checkbox"/>	1001	操作系统	08:00:00	09:00:00	信息学院	315	192.168.1.1	第1次实验 ▾
<input type="checkbox"/>	1003	计算机基础	10:00:00	11:00:00	信息学院	201	192.168.3.1	第1次实验 ▾
[第1页/共1页] [共找到3个课程]								
开始实验								

图 6 教师课程控制

先将课程信息查询，显示查询结果，在网页的表单中可以修改课程次数，再将更新后的次数写回数据库中，完成更新课程次数数据项，实现课程次数更新的功能。

关键代码：

//先查询课程号，再根据课程号更新课程次数。

<?php

```
$link=mysqli_connect("localhost","root","")or die("数据库连接失败！");
```

```

mysqli_select_db($link,"sign")or die("数据链接失败！");
mysqli_query($link,"set names 'utf-8'");
$sql="select  * from course where C_id in(select Q_C_id from sign) order by C_id";
$result=mysqli_query($link,$sql);
$rows=mysqli_num_rows($result);
$page_size=3;
$page_count=ceil($rows/$page_size);
if(isset($_GET["pageno"]))//获取控件中当前页数，检查是否是空，如果不是
    $pageno=$_GET["pageno"];//获取控件中当前页数
if(!isset($pageno)||$pageno<1)//如果 pageno 为空，或 pageno 小于 1
    $pageno=1;//就另 pageno 为 1
if($pageno>$page_count)
    $pageno=$page_count;
$offset=($pageno-1)*$page_size;
mysqli_data_seek($result,$offset);//将指针指向那一页
?>
//更新课程次数
<?php
    $s=$_POST["s"];
    $link=mysqli_connect("localhost","root","")or die("数据库连接失败！");
    $count=0;
    mysqli_select_db($link,"sign")or die("数据链接失败！");
    mysqli_query($link,"set names 'utf-8'");
    for($i=0;$i<count($_POST["list"]);$i++){
        $sql="update sign set Q_count='1',Q_a='$s[$i]' where Q_C_id='{$_POST["list"][$i]}';//
更新对应课程的课程次数
        $result=mysqli_query($link,$sql);
        $count++;
    }
    if($count>0){
        ?>
        <form        action="student_main.php"                id="form1"        name="form1"
method="post"></form>
        <script language="javascript">
            alert("实验设置成功！");
            history.go(-1);
        </script>
        <?php
        }
    else{?>
        <script language="javascript">
            alert("实验设置失败！");
            history.go(-1);
        </script>

```

```
<?php  
}  
?>
```

五、实验总结：

通过对 PHP 代码的开发，不仅掌握了在 Linux 下的 PHP 环境的搭建，而且更进一步了解了 PHP 工作原理与工作过程，掌握了关于 PHP 表单的提交、数据库的连接、增删改查操作，HTML 与 PHP 的混合编程，并对 PHP 的兴趣进一步增强，同时感受到了 Linux 系统与 Windows 系统的区别，相同情况下 Linux 系统优化更好，但没有 Windows 系统方便快捷。今后有兴趣做一个个人博客网站，记录自己编程的点滴，总结自己的工作过程，争取更大进步！