

学生信息管理系统

01 需求分析

能录入、修改、查询、输出学生的档案信息，这些信息包括学生的基本情况、简历情况、获得奖励情况、受到处分情况、学生的家庭信息、学生的体检情况等

能录入、修改、查询、输出学生的入校成绩、各学期各门课的成绩信息，并支持按年级、班级等条件的统计、查询、报表输出。

具体实现：

- (1) 进行需求分析，编写数据字典。
- (2) 设计E-R图。
- (3) 学生数据输入：输入学号、姓名、性别、年龄、出生年月等学生基本信息。
- (4) 成绩输入：输入各学期学生考试成绩。
- (5) 学生信息查询：查询学生基本信息。
- (6) 学生成绩查询：查询学生考试成绩。
- (7) 按年级输出报表：输出全年级学生各科目考试成绩。
- (8) 按班级输出报表：输出全班学生各科目考试成绩。

02 开发语言及环境

开发平台 OSX10.8

软件整合包 XMAPP

—包括了PHP解释环境（搭建本地网络服务器）

—以整合的MySQL数据库

03 简述

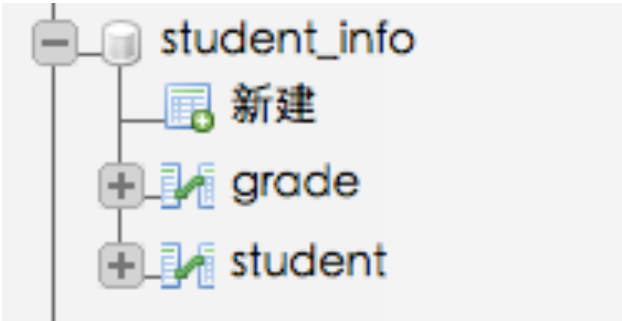
数据库开发前，准备使用PHP动态网页语言+开源数据库MySQL的开发模式。一方面，由于有XMAPP软件整合包，所以开发环境的搭建十分快捷简单。由于前期知识储备几乎为零，花了两天半的时间来学习PHP和数据库方面的基础知识。

《Learning.PHP.MySQL.&.JavaScript.4th.Edition》看相关部分大约250页左右，最后历时半天做出了本文中的学生信息管理系统。整个开发过程几乎没有遇到太大的BUG，还算顺利。基于网页的CSS样式就没有在进一步的学习，所以界面不是很好看。不过基本功能也算是都实现了。

04 数据库层面的准备

由于本问题处理的数据不多，关系也比较简单，经过简单分析后直接建表：

4.1 student_info数据库的结构



只用了两张表grade 和student 分别用来存储全校学生选课成绩与全校学生的基本信息。

4.2 表grade：

#	名字	类型	排序规则	属性	空	默认	额外	操作
<input type="checkbox"/>	1 course	varchar(128)	latin1_swedish_ci		是	NULL		修改 删除 主键 唯一 索引 更多
<input type="checkbox"/>	2 term	char(1)	latin1_swedish_ci		是	NULL		修改 删除 主键 唯一 索引 更多
<input type="checkbox"/>	3 year	char(4)	latin1_swedish_ci		是	NULL		修改 删除 主键 唯一 索引 更多
<input type="checkbox"/>	4 grades	int(11)			是	NULL		修改 删除 主键 唯一 索引 更多
<input type="checkbox"/>	5 id	int(10)		UNSIGNED	是	NULL		修改 删除 主键 唯一 索引 更多
<input type="checkbox"/>	6 num	int(10)		UNSIGNED	否	无	AUTO_INCREMENT	修改 删除 主键 唯一 索引 更多

该表中需要注意：

1：num值相当于选课流水号，唯一标识学生的选课信息。在数据库里设置了值自动增加的属性，也就是说在录入学生信息的时候可以不要考虑，由系统自动生成。这么做的原因是其他属性要多个才可以当做主键，这样处理可以简化代码方便处理。

2：id是本表中的外键，与表student中的学生id相关联，查询时起连接作用。

4.3 表student

#	名字	类型	排序规则	属性	空	默认	额外	操作
<input type="checkbox"/>	1 id	int(10)		UNSIGNED	否	0		修改 删除 主键 唯一 索引 空间 更多
<input type="checkbox"/>	2 name	varchar(128)	latin1_swedish_ci		是	NULL		修改 删除 主键 唯一 索引 空间 更多
<input type="checkbox"/>	3 sex	varchar(8)	latin1_swedish_ci		是	NULL		修改 删除 主键 唯一 索引 空间 更多
<input type="checkbox"/>	4 age	int(10)		UNSIGNED	是	NULL		修改 删除 主键 唯一 索引 空间 更多
<input type="checkbox"/>	5 grade	int(11)			是	NULL		修改 删除 主键 唯一 索引 空间 更多
<input type="checkbox"/>	6 class	int(11)			是	NULL		修改 删除 主键 唯一 索引 空间 更多
<input type="checkbox"/>	7 punishment	varchar(128)	latin1_swedish_ci		是	NULL		修改 删除 主键 唯一 索引 空间 更多
<input type="checkbox"/>	8 awards	varchar(128)	latin1_swedish_ci		是	NULL		修改 删除 主键 唯一 索引 空间 更多
<input type="checkbox"/>	9 born	year(4)			是	NULL		修改 删除 主键 唯一 索引 空间 更多

该表中需要注意：

1: id是主键用来唯一标识数据库中不同的学生。

4.4 数据库存储限制

以上表中的所有属性的值都用 constrain 命令加以限制

比如：mysql> alter table grade add constraint c1 check(grades between 0 and 100);
说明学生的成绩只可以在0到100之间，其他数据是无法被存到数据库中的。

其他限制如下

```
mysql> alter table grade add constraint c2 check(term in ('1','2','3'));  
mysql> alter table grade add constraint c3 check(year between '1990' and '2050');  
mysql> alter table student add constraint c1 check(born between '1980' and '2020');  
mysql> alter table student add constraint c2 check(age between 1 and 100);  
mysql> alter table student add constraint c3 check(sex in ('male','female'));  
mysql> alter table student add constraint c4 check(class in (1,2,3,4,5,6));  
mysql> alter table student add constraint c5 check(grade in (1,2,3,4,5,6));
```

以上简述了开发数据库的前期准备，接下来就开始我们的网页版学生信息管理系统的开发。

05 数据库的网页开发层面

5.1 登陆界面：

学生信息管理系统

用户名：	<input type="text" value="ssd227"/>
用户密码：	<input type="password"/>
<input type="button" value="log in"/>	

用户名已经自动补全输入密码即可登陆主功能目录。

5.2 主菜单目录：

系统主要功能目录

1:录入或修改成绩

2:录入或修改学生信息

3:输出年级/班级报表

4:输出学生信息或成绩

5.3 具体功能实现子目录：

子目录1：录入和修改成绩

Course	<input type="text"/>	如: math, chinese, english
Term	<input type="text"/>	如: 只可以输入 1, 2
Year	<input type="text"/>	如: 2014 （表示2014学年）
Grades	<input type="text"/>	如: 99 （分）单位省略
ID	<input type="text"/>	如: 选课学生的ID
NUM(仅在修改时需要填写, 系统自动生成)		<input type="text" value="NULL"/>
ADD	添加 <input checked="" type="radio"/>	
MODIFY	修改 <input type="radio"/>	
提交	<input type="button" value="SUBMIT"/>	

结果如下：（可随意删改）

Course english
Term 2
Year 2012
Grade 99
ID 1
NUM 1

Course math
Term 1
Year 2010
Grade 88

如上图所示，本页实现了学生成绩的修改和录入功能。而录入和修改的切换有radio按钮实现，这些控件都是html 语言所自带的，添加和使用起来也是比较方便的。输入框下方是所有数据中信息的反馈，用户可以直接删除任何一条记录。由于时间仓促，本来是可以将下方信息做成表格输出的，那样用户的使用体验将会大大提升。

子目录2：录入和修改学生信息

ID	学生ID	<input type="text"/>	正整数即可
NAME	姓名	<input type="text"/>	
SEX	性别	<input type="text"/>	如: male, female
AGE	年龄	<input type="text"/>	正整数即可
GRADE	年级	<input type="text"/>	如: 2 (表示2年级)
CLASS	班级	<input type="text"/>	如: 1 (表示1班)
PUNISHMENT		<input type="text"/>	限制在128个字符
AWARDS		<input type="text"/>	限制在128个字符
BORN	出生年份	<input type="text"/>	如: 1993
ADD	添加	<input checked="" type="radio"/>	
MODIFY	修改	<input type="radio"/>	
提交		<input type="button" value="SUBMIT"/>	

结果如下：（可随意删改）

```
ID      0
NAME    haha
SEX     male
AGE     13
GRADE   2
CLASS   2
PUNISHMENT
AWARDS
BORN    1973
```

与上文中的修改录入操作不同，下面的子目录3和4则实现了固定需求的查询操作。比如全校学生成绩按班级或年级输出，和按学号id来查询学生个人的成绩以及详细的个人信息。总而言之子目录1，2代码实现类似，子目录3，4代码实现类似。更多信息请查阅附录中的程序源码。

子目录3: 年级和班级成绩报表

GRADE

年级成绩报表

GRADE RECORD

GRADE

CLASS

班级成绩报表

CLASS RECORD

```
GRADE 1
ID      3
NAME    sala
COURSE  chinese
GRADES  90
```

子目录4: 学生成绩和信息查询

ID(学号)

学生成绩 ☒

学生信息 ☐

OUTPUT

```
STUDENT_ID      1
NAME            Jake1
BORN            1994
SEX             female
AGE            17
GRADE           2
CLASS           1
PUNISHMENT      NULL
AWARDS          NULL
```

5.3 安全与异常诊断:

作为网页程序，我们就不能不考虑学生管理系统的安全性问题，不管是用户有意还是无意的输出。最恶劣的比如XSS注入和SQL注入等。以下是书中为我们提供的解决办法，附录中代码也有所体现

Functions for preventing both SQL and XSS injection attacks

```
<?php
function mysql_entities_fix_string($conn, $string)
{
    return htmlentities(mysql_fix_string($conn, $string));
}
function mysql_fix_string($conn, $string)
{
    if (get_magic_quotes_gpc()) $string = stripslashes($string);
    return $conn->real_escape_string($string);
}
?>
```

对于连接访问数据库可能发生的各种异常情况，本系统也做了相应的处理和文本提示，在此就不赘述

附录

代码关系

login.php被其他文件引用（内存用户名与密码等信息）

```
index.php |—> main.php |—> sub01.php
              |—> sub02.php
              |—> sub03.php
              |—> sub04.php
```

部分源码

index.php

```
<?php // formtest.php
echo <<<_END
<html>
<head>
<title>Form Test</title>
</head>
<body>
<h1>学生信息管理系统</h1>
<form method="post" action="main.php"><pre>
  用户名: <input type="text" name="name" value='ssd227'>
  用户密码: <input type="text" name="password">
  <input type="submit" value="log in">
</pre></form>
</body>
</html>
_END;
?>
```

main.php

```
<?php // sqltest.php
require_once 'login.php';

//check login information
if ($_POST['name']!= 'ssd227' || $_POST['password']!= 'ssd227')
    die("your name or password is wrong");

//$un=$_POST['name'];
//$pw=$_POST['password'];

echo <<<_END
<h1>系统主要功能目录</h1>
<form action="sub01.php" method="post">
  1:录入或修改成绩      <input type="submit" value="成绩">
</form>
<form action="sub02.php" method="post">
  2:录入或修改学生信息 <input type="submit" value="学生信息">
</form>
<form action="sub03.php" method="post">
  3:输出年级/班级报表  <input type="submit" value="年级/班级报表">
</form>
<form action="sub04.php" method="post">
  4:输出学生信息或成绩 <input type="submit" value="学生成绩/信息">
</form>
_END;
?>
```


sub01.php

```
<?php // sub01.php
require_once 'login.php';

$conn = new mysqli($hn, $un, $pw, $db);
if ($conn->connect_error) die($conn->connect_error);

//delete operation
if (isset($_POST['delete']) &&
    isset($_POST['g_num']))
{
    $num = get_post($conn, 'g_num');

    $query = "DELETE FROM grade WHERE
        num='$num'";

    $result = $conn->query($query);
    if (!$result) echo "DELETE failed: $query<br>" .
        $conn->error . "<br><br>";
}

if (isset($_POST['course']) &&
    isset($_POST['year']) &&
    isset($_POST['term']) &&
    isset($_POST['id']) &&
    isset($_POST['grades']))
{
    $course = get_post($conn, 'course');
    $year = get_post($conn, 'year');
    $term = get_post($conn, 'term');
    $id = get_post($conn, 'id');
    $grades = get_post($conn, 'grades');
    $select = get_post($conn, 'select');
    $num = $_POST['num'];

    if($select==1)
    {
        $query = "INSERT INTO grade VALUES" .
            "('$course', '$term', '$year', '$grades', '$id', NULL)";
        $result = $conn->query($query);
        if (!$result) echo "INSERT failed: $query<br>" .
            $conn->error . "<br><br>";
    }
    elseif($select==2 && $num>0)
    {
        $query = "UPDATE grade SET course='$course',term='$term',".
            "year='$year',grades='$grades',id='$id'".
            "where num='$num'";
        $result = $conn->query($query);
        if (!$result) echo "MODIFY failed: $query<br>" .
            $conn->error . "<br><br>";
    }
}

echo <<<_END
<h1>子目录1：录入和修改成绩</h1>
<form action="sub01.php" method="post"><pre>
    Course <input type="text" name="course"> 如： math, chinese, english
    Term <input type="text" name="term"> 如： 只可以输入 1, 2
```

```

Year <input type="text" name="year"> 如：2014 （表示2014学年）
Grades <input type="text" name="grades"> 如：99 （分）单位省略
ID <input type="text" name="id"> 如：选课学生的ID
NUM(仅在修改时需要填写，系统自动生成) <input type="text" name="num" value="NULL">
ADD 添加 <input type="radio" name="select" value="1" checked="checked">
MODIFY 修改 <input type="radio" name="select" value="2">
提交 <input type="submit" name="add" value="SUBMIT">

```

```
</pre></form>
```

```
<h2>结果如下：（可随意删改）</h2>
```

```
_END;
```

```

$query = "SELECT * FROM grade";
$result = $conn->query($query);
if (!$result) die ("Database access failed: " . $conn->error);

```

```
$rows = $result->num_rows;
```

```
for ($j = 0 ; $j < $rows ; ++$j)
```

```

{
    $result->data_seek($j);
    $row = $result->fetch_array(MYSQLI_NUM);

```

```
    echo <<<_END
```

```
<pre>
```

```
Course $row[0]
```

```
Term $row[1]
```

```
Year $row[2]
```

```
Grade $row[3]
```

```
ID $row[4]
```

```
NUM $row[5]
```

```
</pre>
```

```
<form action="sub01.php" method="post">
```

```
<input type="hidden" name="delete" value="yes">
```

```
<input type="hidden" name="g_num" value="$row[5]">
```

```
<input type="submit" value="DELETE"></form>
```

```
-----
_END;
```

```
}
```

```
$result->close();
```

```
$conn->close();
```

```
function get_post($conn, $var)
```

```

{
    return $conn->real_escape_string($_POST[$var]);
}

```

```
?>
```

sub03.php

```
<?php // sub03.php
```

```
require_once 'login.php';
```

```
$conn = new mysqli($hn, $un, $pw, $db);
```

```
if ($conn->connect_error) die($conn->connect_error);
```

```
echo <<<_END
```

```
<h1>子目录3: 年级和班级成绩报表</h1>
```

```
<form action="sub03.php" method="post"><pre>
```

```
GRADE <input type="text" name="grade">
```

```
年级成绩报表 <input type="submit" value="GRADE RECORD">
```

```
</pre></form>
```

```
<form action="sub03.php" method="post"><pre>
```

```
GRADE <input type="text" name="grade1">
```

```
CLASS <input type="text" name="class">
```

```
    班级成绩报表 <input type="submit" value="CLASS RECORD">
</pre></form>
```

```
-----
_END;
```

```
if (isset($_POST['grade1']) && isset($_POST['class']))
{
    $grade = get_post($conn, 'grade1');
    $class = get_post($conn, 'class');

    $query = "select grade,class,student.id,name,course,grades
              from
                  student join grade on student.id=grade.id
              where
                  grade='$grade' AND class='$class'
              ORDER BY id,course,grades DESC";
```

```
$result = $conn->query($query);
if (!$result) echo "SELECT failed: $query<br>" .
    $conn->error . "<br><br>";
```

```
//output ...
```

```
$rows = $result->num_rows;
```

```
for ($j = 0 ; $j < $rows ; ++$j)
{
    $result->data_seek($j);
    $row = $result->fetch_array(MYSQLI_NUM);
```

```
echo <<<_END
<pre>
    GRADE $row[0]
    CLASS $row[1]
    ID   $row[2]
    NAME $row[3]
    COURSE $row[4]
    GRADES $row[5]
</pre>
```

```
-----
_END;
```

```
}
    $result->close();
}
```

```
elseif (isset($_POST['grade']) )
{
    $grade = get_post($conn, 'grade');
```

```
$query = "select grade,student.id,name,course,grades
          from
              student join grade on student.id=grade.id
          where
              grade='$grade'
          ORDER BY id,course,grades DESC";
```

```
$result = $conn->query($query);
if (!$result) echo "SELECT failed: $query<br>" .
    $conn->error . "<br><br>";
```

```
//output ...
```

```
$rows = $result->num_rows;
```

```
for ($j = 0 ; $j < $rows ; ++$j)
{
    $result->data_seek($j);
```

```
$row = $result->fetch_array(MYSQLI_NUM);

echo <<<_END
  <pre>
    GRADE $row[0]
    ID   $row[1]
    NAME $row[2]
    COURSE $row[3]
    GRADES $row[4]
  </pre>
  -----
_END;
}
$result->close();
}

$conn->close();

function get_post($conn, $var)
{
  return $conn->real_escape_string($_POST[$var]);
}
?>
```