



西安电子科技大学

计算机学院

数据库概论实验报告

题 目： 设计学生管理系统

组 长： 阮思捷

组 员： 姜喆 苏奇 穆丹

班 级： 1303018

指导老师： 祁建军

2015 年 12 月

组员分工

- 阮思捷 13030188013 (组长) : 数据库编码实现和Web前端交互界面的编写。
- 姜喆 13030188006 (组员) : 将需求转换为E-R图, 并对其进行化简, 数据表设计。
- 苏奇 13030188018 (组员) : 前端界面设计, 完整性检查, 数据填充。
- 穆丹 13030188001 (组员) : 实验报告撰写, 数据填充。

需求分析

设计学生管理系统

今要建立关于系、学生、班级、学会等诸信息的一个关系数据库。一个系有若干专业, 每个专业每年只招一个班, 每个班有若干学生。一个系的学生住在同一宿舍区。每个学生可参加若干学会, 每个学会有若干学生。学生参加某学会会有一个入会年份。描述各个实体的属性 (加下划线者为主码) 如下:

- **学生**: 学号、姓名、年龄、系名、班号、宿舍区。
- **班级**: 班号、专业名、入校年份、系名、人数。
- **系**: 系号、系名、系办公室地点、人数。
- **学会**: 学会号、学会名、成立年份、地点。

任务要求

1. 画出E-R图。
2. 把E-R图转为关系模式。
3. 根据关系模式创建数据库。表名和属性名用英文, 属性的数据类型根据上面的描述自己定义。
4. 创建一个视图, 能显示每个学会的学会名, 学生数(实际不存在, 也不能增加)。
5. 创建一个触发器, 能根据每个班的学生变动情况自动增减班级表和系表的人数字段的值。
6. 创建一个存储过程, 实现如下功能: 给定一个班的旧班号和新班号, 把所有相关表中此班的旧班号改为新班号, 并返回此班的人数 (使用输出参数)。
7. 编写一段脚本, 使用游标完成如下功能: 确定系表中人数字段的值与实际学生数是否相符。如果不相符, 把人数字段的值改为实际数, 并在窗口打印此系的系号、系名、原人数、实际人数。

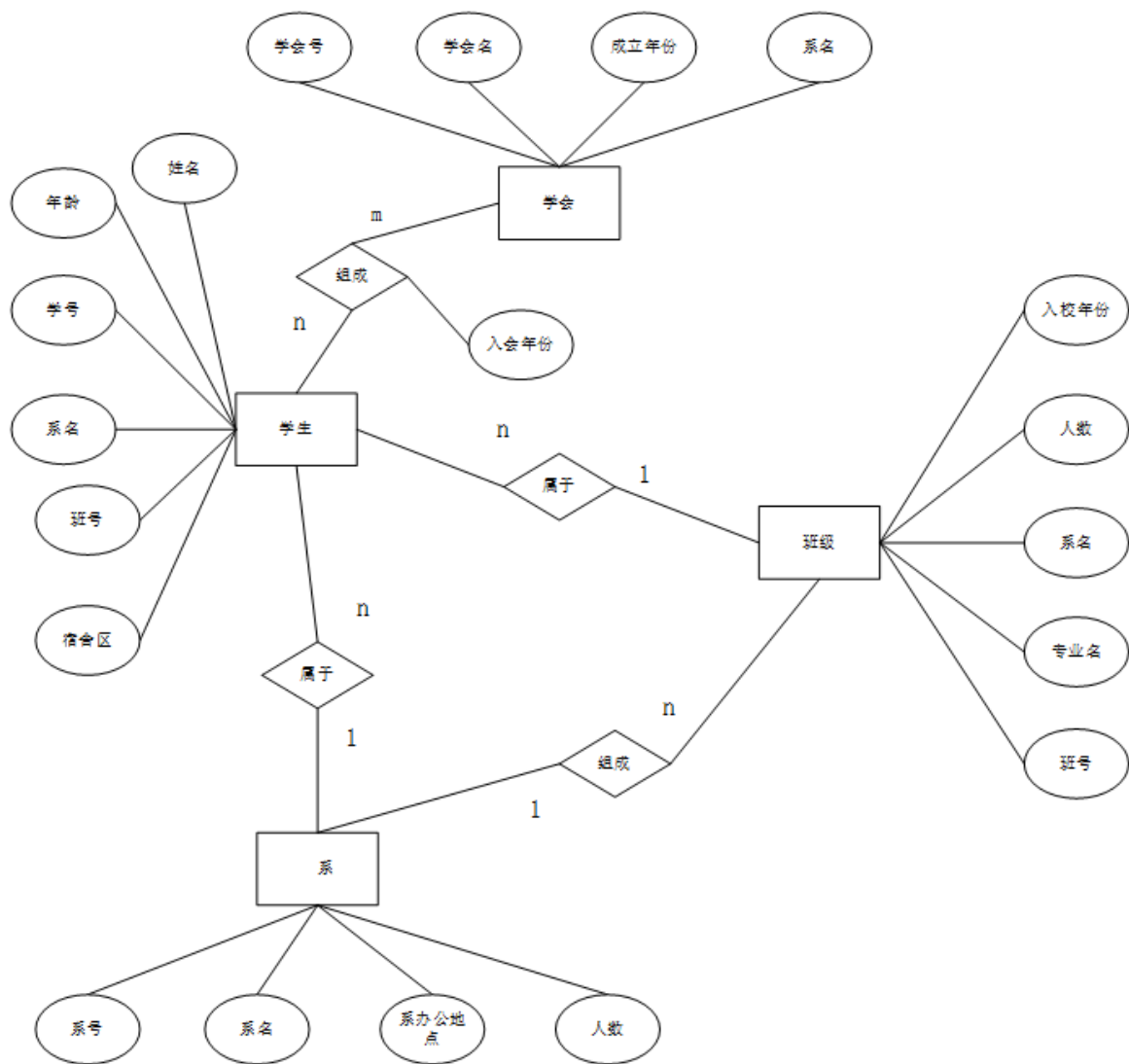
实验环境

Apache 2.4.12
PHP 5.6.11
MySQL Community Server (GPL) 5.6.25
Chrome

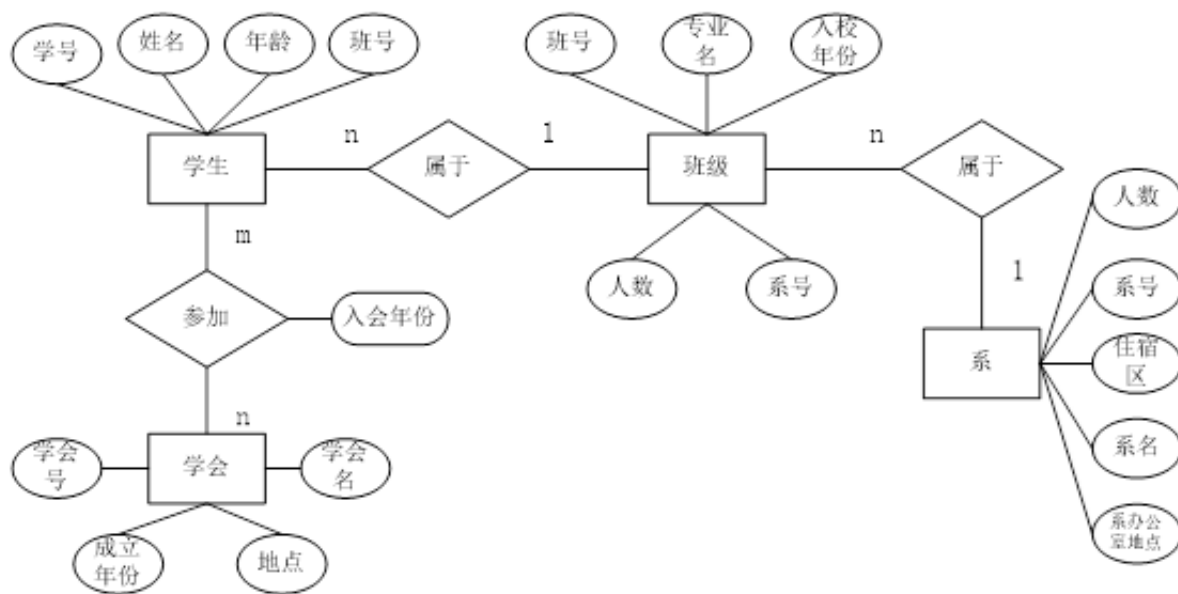
概念结构设计

画出E-R图。

原始E-R图



化简后E-R图



逻辑结构设计

把E-R图转为关系模式。

student<U, F>

$U = \{Sno, Sname, Sage, Sclass\}$

$F = \{Sno \rightarrow Sname, Sno \rightarrow Sage, Sno \rightarrow Sclass\}$

student 为描述学生的关系模式，**Sno** 为学号，**Sname** 为姓名，**Sage** 为年龄，**Sclass** 为班级。

class<U, F>

$U = \{Cno, Cmajor, Cyear, Cdept, Cstu\}$

$F = \{Cno \rightarrow Cmajor, Cno \rightarrow Cyear, Cno \rightarrow Cdept, Cno \rightarrow Cstu\}$

class 为描述班级的关系模式，**Cno** 为班号，**Cmajor** 为专业名，**Cyear** 为入校年份，**Cdept** 为系号，**Cstu** 为人数。

department<U, F>

$U = \{Dno, Dname, Daddr, Dstu, Ddorm\}$

$F = \{Dno \rightarrow Dname, Dno \rightarrow Daddr, Dno \rightarrow Dstu, Dno \rightarrow Ddorm\}$

department 为描述系的关系模式，**Dno** 为系号，**Dname** 为系名，**Daddr** 为系办公室地点，**Dstu** 为人数，**Ddorm** 为住宿区。

institute<U, F>

$U = \{Ino, Iname, Iyear, Iaddr\}$

$F = \{Ino \rightarrow Iname, Ino \rightarrow Iyear, Ino \rightarrow Iaddr\}$

`institute` 为描述学会的关系模式，`Ino` 为学会号，`Iname` 为学会名，`Iyear` 为成立时间，`Iaddr` 为地点。

si<U, F>

$U = \{Sno, Ino, Jyear\}$

$F = \{(Sno, Ino) \rightarrow Jyear\}$

`si` 为描述学生加入学会的关系模式，`Sno` 为学号，`Ino` 为系号，`Jyear` 为加入时间。

每个表均不存在非主属性对码的部分依赖，不存在非主属性对码的传递依赖，不存在主属性对码的部分和传递依赖，符合BCNF的要求。

具体实现

根据关系模式创建数据库。表名和属性名用英文，属性的数据类型根据上面的描述自己定义。

```
/* 系表 */
CREATE TABLE IF NOT EXISTS `department` (
  `Dno` int(2) NOT NULL COMMENT '系号',
  `Dname` varchar(30) NOT NULL COMMENT '系名',
  `Daddr` text NOT NULL COMMENT '系办公室地点',
  `Dstu` int(5) unsigned NOT NULL COMMENT '人数',
  `Ddorm` text NOT NULL COMMENT '宿舍区',
  PRIMARY KEY (`Dno`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='系';

/* 班级表 */
CREATE TABLE IF NOT EXISTS `class` (
  `Cno` char(7) NOT NULL COMMENT '班号',
  `Cmajor` varchar(30) NOT NULL COMMENT '专业名',
  `Cyear` int(4) NOT NULL COMMENT '入校年份',
  `Cdept` int(11) NOT NULL COMMENT '系号',
  `Cstu` int(3) NOT NULL COMMENT '人数',
  CHECK(`Cyear` > 1970 AND `Cyear` < 2100),
  PRIMARY KEY (`Cno`),
  FOREIGN KEY (`Cdept`) REFERENCES department(Dno)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='班级';

/* 学生表 */
CREATE TABLE IF NOT EXISTS `student` (
  `Sno` char(11) NOT NULL COMMENT '学号',
  `Sname` varchar(30) NOT NULL COMMENT '姓名',
```

```

`Sage` tinyint(3) unsigned NOT NULL COMMENT '年龄',
`Sclass` char(7) NOT NULL COMMENT '班号',
CHECK(`Sage` > 0 AND `Sage` < 100),
PRIMARY KEY (`Sno`),
FOREIGN KEY (`Sclass`) REFERENCES class(Cno)
    ON UPDATE CASCADE
    ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='学生';

/* 学会表 */
CREATE TABLE IF NOT EXISTS `institute` (
    `Ino` int(5) unsigned NOT NULL COMMENT '学会号',
    `Iname` varchar(30) NOT NULL COMMENT '学会名',
    `Iyear` int(4) NOT NULL COMMENT '成立年份',
    `Iaddr` text NOT NULL COMMENT '地点',
    CHECK(`Iyear` > 1970 AND `Iyear` < 2100),
    PRIMARY KEY (`Ino`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='学会';

/* 学生参加学会表 */
CREATE TABLE IF NOT EXISTS `si` (
    `Sno` char(11) NOT NULL COMMENT '学号',
    `Ino` int(11) unsigned NOT NULL COMMENT '学会号',
    `Jyear` int(4) NOT NULL COMMENT '入会年份',
    CHECK(`Jyear` > 1970 AND `Jyear` < 2100),
    PRIMARY KEY (`Sno`, `Ino`),
    FOREIGN KEY (`Sno`) REFERENCES student(Sno),
    FOREIGN KEY (`Ino`) REFERENCES institute(Ino)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='参加';

```

创建一个视图，能显示每个学会的学会名，学生数(实际不存在，也不能增加)。

```

CREATE VIEW v_inststu(Iname, Istu)
AS
SELECT Iname, COUNT(Sno)
FROM si, institute
WHERE si.Ino = institute.Ino
GROUP BY institute.Ino;

```

创建一个触发器，能根据每个班的学生变动情况自动增减班级表和系表的人数字段的值。

```

DELIMITER $
CREATE TRIGGER t_insert_stu
AFTER INSERT ON student
FOR EACH ROW
BEGIN
UPDATE class
SET Cstu = (
    SELECT COUNT(*)
    FROM student
    WHERE Sclass = class.Cno);
UPDATE department
SET Dstu = (
    SELECT SUM(Cstu)
    FROM class
    WHERE Cdept = department.Dno);
END
$

CREATE TRIGGER t_delete_stu
AFTER DELETE ON student
FOR EACH ROW
BEGIN
UPDATE class
SET Cstu = (
    SELECT COUNT(*)
    FROM student
    WHERE Sclass = class.Cno);
UPDATE department
SET Dstu = (
    SELECT SUM(Cstu)
    FROM class
    WHERE Cdept = department.Dno);
END
$
DELIMITER ;

```

创建一个存储过程，实现如下功能：给定一个班的旧班号和新班号，把所有相关表中此班的旧班号改为新班号，并返回此班的人数（使用输出参数）。

```

DELIMITER $
CREATE PROCEDURE changeCno(in oCno char(7), in nCno char(7), out total int)
BEGIN
UPDATE class
SET Cno = nCno
WHERE Cno = oCno;
SELECT COUNT(*)
INTO total
FROM student
WHERE Sclass = nCno;
END
$
DELIMITER ;

```

编写一段脚本，使用游标完成如下功能：确定系表中人数字段的值与实际学生数是否相符。如果不相符，把人数字段的值改为实际数，并在窗口打印此系的系号、系名、原人数、实际人数。

```

DELIMITER $
/* 创建临时存储过程 */
CREATE PROCEDURE checkstu()
BEGIN
/* 定义要使用的变量 */
DECLARE vDno int(11);
DECLARE vDname varchar(30);
DECLARE vOriginal int(11);
DECLARE vActual int(11);
DECLARE done INT DEFAULT 0;
/* 定义游标 */
DECLARE cur CURSOR FOR
SELECT department.Dno, department.Dname, department.Dstu, COUNT(*)
FROM student, class, department
WHERE student.Sclass = class.Cno AND
class.Cdept = department.Dno
GROUP BY department.Dno;
/* 处理遍历结束 */
DECLARE CONTINUE HANDLER FOR SQLSTATE '02000' SET done=1;
DECLARE CONTINUE HANDLER FOR SQLSTATE '23000' SET done=1;
/* 创建临时表记录有变动的系 */
CREATE TABLE IF NOT EXISTS `fixstulog`(
`Dno` int(11) NOT NULL COMMENT '系号',
`Dname` varchar(30) NOT NULL COMMENT '系名',
`Original` int(11) unsigned NOT NULL COMMENT '原始学生数',
`Actual` int(11) NOT NULL COMMENT '实际学生数',

```



```

PRIMARY KEY (`Dno`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='检查学生数操作日志';
/* 打开游标 */
OPEN cur;
REPEAT
  IF NOT done THEN
    FETCH NEXT FROM cur INTO vDno, vDname, vOriginal, vActual;
    /* 原始学生数与实际学生数不一致，加表更新并显示。 */
    IF vOriginal <> vActual THEN
      INSERT INTO fixstulog (`Dno`, `Dname`, `Original`, `Actual`)
VALUES (vDno, vDname, vOriginal, vActual);
      UPDATE department
      SET Dstu = vActual
      WHERE Dno = vDno;
    END IF;
  END IF;
UNTIL done END REPEAT;
SELECT *
FROM fixstulog;
/* 关闭游标 */
CLOSE cur;
/* 删除临时表 */
DROP TABLE fixstulog;
END
$
DELIMITER ;
/* 调用临时存储过程 */
CALL checkstu();
/* 删除临时存储过程 */
DROP PROCEDURE checkstu;

```

调试分析

1. 在MySQL中一个触发器只能在 `DELETE` 和 `UPDATE` 操作中选择其一，故要创建两个触发器监视人数的增减。
2. 在 `student` 表中引用了 `class` 的外键，在对 `class` 表进行更新或者删除操作不成功，要对 `student` 表中的外键选择 `CASCADE` 级联操作。
3. 对年龄，年份等字段加上取值范围的约束防止信息误录。
4. 将数据库的编码设置为UTF-8保证对中文的兼容性。
5. 在游标循环遍历过程中，设置Handler标记循环结束。

实验结果

数据库测试

测试视图

```
mysql> SELECT * FROM v_inststu;
+-----+-----+
| Iname          | Istu |
+-----+-----+
| 西电微软学生俱乐部 |    1 |
| 西电开源社区   |    1 |
+-----+-----+
2 rows in set (0.00 sec)
```

学会总人数字段没有存在数据库中，通过创建视图成功获取结果。

测试触发器

```
mysql> SELECT * FROM class WHERE Cno=1303018;
+-----+-----+-----+-----+-----+
| Cno    | Cmajor          | Cyear | Cdept | Cstu |
+-----+-----+-----+-----+-----+
| 1303018 | 计算机科学与技术 | 2013  | 3     | 30   |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

目前1303018班共有30名同学。

```
mysql> DELETE FROM student WHERE Sno=13030188006;
Query OK, 1 row affected (0.31 sec)
```

删除学号为 13030188006 的同学。

```
mysql> SELECT * FROM class WHERE Cno=1303018;
+-----+-----+-----+-----+-----+
| Cno    | Cmajor          | Cyear | Cdept | Cstu |
+-----+-----+-----+-----+-----+
| 1303018 | 计算机科学与技术 | 2013  | 3     | 29   |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

目前1303018班只有29名同学，删除触发成功。

```
mysql> INSERT INTO `student` (`Sno`, `Sname`, `Sage`, `Sclass`) VALUES ('13030188006', '姜喆', 21, '1303018');
Query OK, 1 row affected (0.08 sec)
```

将该同学加回表中。

```
mysql> SELECT * FROM class WHERE Cno=1303018;
+-----+-----+-----+-----+-----+
| Cno    | Cmajor          | Cyear | Cdept | Cstu |
+-----+-----+-----+-----+-----+
| 1303018 | 计算机科学与技术 | 2013  | 3     | 30   |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

目前1303018班有30名同学，更新触发成功。

测试存储过程

```
mysql> SET @total = 0;  
Query OK, 0 rows affected (0.00 sec)
```

声明变量 `total` 用于存放存储过程返回的值。

```
mysql> CALL changeCno(1303018, 1303038, @total);  
Query OK, 1 row affected (0.49 sec)
```

调用存储过程 `changeCno`，将班号 `1303018` 改为 `1303038`。

```
mysql> SELECT @total;  
+-----+  
| @total |  
+-----+  
|      30 |  
+-----+  
1 row in set (0.00 sec)
```

返回值显示该班学生为30人。

测试脚本

```
mysql> UPDATE department SET Dstu=1000;
Query OK, 3 rows affected (0.34 sec)
Rows matched: 3  Changed: 3  Warnings: 0
```

虚假设置每个系有1000人。

```
mysql> SELECT * FROM department;
+----+-----+-----+-----+-----+
| Dno | Dname          | Daddr | Dstu | Ddorm |
+----+-----+-----+-----+-----+
| 1   | 通信工程学院   | 行政楼 | 1000 | 8号楼 |
| 2   | 电子工程学院   | 行政楼 | 1000 | 9号楼 |
| 3   | 计算机学院     | 行政楼 | 1000 | 10号楼 |
+----+-----+-----+-----+-----+
3 rows in set (0.02 sec)
```

系表中人数字段更新为虚假人数。

```
+----+-----+-----+-----+
| Dno | Dname          | Original | Actual |
+----+-----+-----+-----+
| 1   | 通信工程学院   | 1000     | 1      |
| 2   | 电子工程学院   | 1000     | 1      |
| 3   | 计算机学院     | 1000     | 61     |
+----+-----+-----+-----+
3 rows in set (0.58 sec)
```

执行脚本后，返回虚假人数和真实人数。

```
mysql> SELECT * FROM DEPARTMENT;
+----+-----+-----+-----+-----+
| Dno | Dname          | Daddr | Dstu | Ddorm |
+----+-----+-----+-----+-----+
| 1   | 通信工程学院   | 行政楼 | 1     | 8号楼 |
| 2   | 电子工程学院   | 行政楼 | 1     | 9号楼 |
| 3   | 计算机学院     | 行政楼 | 61    | 10号楼 |
+----+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

系表中人数字段更新为真实人数。

Web前端测试

首页

主页面，各个模块功能简介。

学生管理系统 首页 学生 班级 系 学会 高级

学生管理系统

这是一个面向西安电子科技大学学生的学生管理系统。

现在开始

学生

添加学生或者根据学号查询某同学的详细信息。

查看详情 »

班级

查看各个班级专业、人数等信息。

查看详情 »

系

查看各个系办公地点、总人数等信息。

查看详情 »

学会

查看学校内各个学会的介绍、人数等信息。

查看详情 »

高级

一些高级功能。

查看详情 »

© 阮思捷 姜喆 苏奇 穆丹 2015

学生

通过学号搜索学生信息。

学生管理系统 首页 学生 班级 系 学会

学生

+ 添加学生

🔍 查询学生

请输入学号

查询

添加学生信息

添加学生信息。

学生管理系统 首页 学生 班级 系 学会

添加

学号

13030188013

姓名

阮思捷

年龄

21

班级

1303018

确定

查询学生信息

查看学生信息及所加学会，可以编辑或者删除。

查询结果

学号
13030188013

姓名
阮思捷

年龄
21

系名
计算机学院

班号
1303038

宿舍区
10号楼

学会
西电微软学生俱乐部
西电开源社区

编辑

删除

编辑学生信息

编辑学生信息。

编辑

学号
13030188013

姓名
阮思捷

年龄
21

班级
1303018

确定

班级

查看班级信息。

班级

班号	专业名	入校年份	系名	人数
1303018	计算机科学与技术	2013	计算机学院	30
1303028	网络工程	2013	计算机学院	28
1303041	物联网工程	2013	计算机学院	3

系

查看系信息。

系

系号	系名	系办公室地点	人数
1	通信工程学院	行政楼	0
2	电子工程学院	行政楼	0
3	计算机学院	行政楼	61

学会

查看学会信息。

学会

学会号	学会名	成立年份	地点	人数
1	西电微软学生俱乐部	2007	1号楼1区323	1
2	西电开源社区	2008	E楼	1

高级

通过调用存储过程修改旧班号为新班号。

高级

批量修改班号

1303018

1303038

修改

高级

批量修改班号

成功将班号1303018修改为班号1303038。
该班人数共计30人。

总结

通过本次实验，我们组四个人对数据库的设计有了更加深入的思考，锻炼了我们的实践能力和合作能力。我们尝试了对数据库建表、创建视图、创建触发器、创建存储过程和编写脚本等等功能。对关系数据理论有了感性和理性的认识，收获很大。实验过程中遇到了一些问题，与课本上稍有出入，通过查找资料，官方文档解决，对我们解决问题的能力也有很大的提高。