# Introduction to Databases\* Lab Report $I_{\text{\tiny LAL}}$

\* Teacher: 张剡. TA: 助教哥

实验一:用SQL进行数据操作实验报告

张逸凯 171840708

Department of Computer Science and Technology Nanjing University

zykhelloha@gmail.com Tel: 18051988316

# 目录

#### 目录

#### 实验环境

#### 实验过程

- 1. 使用 SQL语句建立基本表 语句建立基本表 语句建立基本表 (Course, Student, SC, Teacher):
- 2. 使用 SQL语句修改基本表 语句修改基本表
  - 2.1 在 Student表中加入属性 表中加入属性 表中加入属性 age(smallint型)
    - 2.2 将 Student表中的属性 表中的属性 表中的属性 age 类型改为 类型改为 int 型
- 3. 使用 SQL 语句插入数据
  - 3.1. 向 Student 表插入不少于 10 条数据。
  - 3.2. 向 Course 表插入不少于 5 条数据。注意,应该包含计算机系 (CS) ,包含数据库课。
  - 3.3. 向 SC 表插入不少于 20 条数据。注意,应该包含 2018 年前后课程。
  - 3.4. 向 Teacher 表插入不少于 3 条数据。注意,应该包含姓胡的教师。

#### 当前表插入后结果

- 4. 查询
  - 4.1 找出所有至少选修了一门计算机系课程的学生姓名,保证结果中没有重复的姓名。
  - 4.2 找出所有姓胡的教师的姓名和院系。
  - 4.3 找出所有没有选修在 2018 年之前 (不含 2018 年) 开设的任何课程的学生的 ID 和姓名。
  - 4.4 找出每个系教师的最高工资值。可以假设每个系至少有一位教师。
- 5. 修改数据
  - 5.1. 将数据库课的学生成绩全部加 2 分。
- 6. 删除数据
  - 6.1. 删除均分不足 80 分的所有学生的选课记录。
- 7. 视图操作
  - 7.1. 创建一个视图,记录每位学生已修课程的总学分数。
- 8. 删除基本表
  - 8.1. 删除 SC 表。

实验中遇到的困难及解决办法

参考文献及致谢

# 实验环境

```
操作系统: Windows 10
```

软件环境: MySQL Workbench 8.0 CE, MySQL 8.0.18

# 实验过程

1. 使用 SQL语句建立基本表 语句建立基本表 语句建立基本表 (Course, Student, SC, Teacher):

```
create table Course(
   id int NOT NULL primary key,
   title char(20) NOT NULL,
    dept_name char(2),
    credit int
);
create table Student(
    id int NOT NULL primary key,
    name char(20) NOT NULL,
    dept_name char(2)
);
create table SC (
   student_id int NOT NULL,
    course_id int NOT NULL,
    primary key(student_id, course_id),
    year int,
    grade int
);
create table Teacher (
    id int NOT NULL primary key,
    name char(8) NOT NULL,
    dept_name char(2),
    salary int
);
```

- 2. 使用 SQL语句修改基本表 语句修改基本表
- 2.1 在 Student表中加入属性 表中加入属性 表中加入属性 age (smallint型)

```
alter table Student add age smallint;
```

2.2 将 Student表中的属性 表中的属性 表中的属性 age 类型改为 类型改为 int 型

```
alter table Student change age age int;
```

# 3. 使用 SQL 语句插入数据

3.1. 向 Student 表插入不少于 10 条数据。

```
insert into Student
values
(0, 'zhang', 'cs', 18),
(1, 'zhu', 'cs', 19),
(2, 'chen', 'cs', 18),
(3, 'li', 'cs', 20),
(4, 'zhang', 'cs', 18),
(5, 'zha', 'cs', 19),
(6, 'gao', 'cs', 18),
(7, 'wu', 'cs', 18),
(8, 'xie', 'cs', 18),
(9, 'zhang', 'cs', 18),
(11, 'liang', 'cs', 18),
(12, 'wang', 'cs', 18);
```

3.2. 向 Course 表插入不少于 5 条数据。注意,应该包含计算机系 (CS) ,包含数据库课。

```
insert into Course
values
(91, 'Database', 'cs', 3),
(92, 'DataMining', 'cs', 3),
(93, 'ML', 'cs', 3),
(94, 'MathAnalysis', 'ma', 5),
(95, 'Cv', 'cs', 4),
(96, 'Human', 'ch', 3),
(97, 'play', 'ha', 1);
```

3.3. 向 SC 表插入不少于 20 条数据。注意,应该包含 2018 年前后课程。

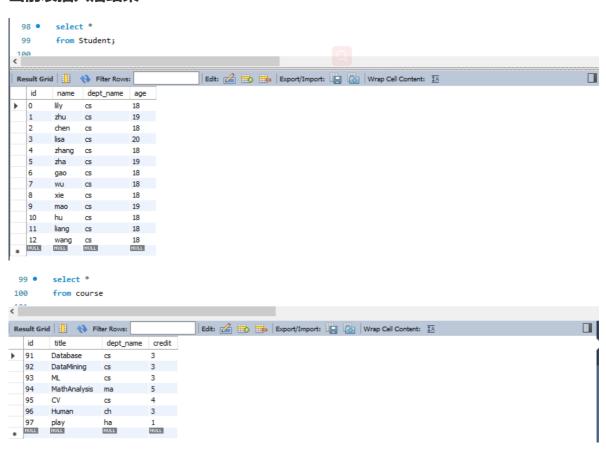
```
insert into SC
values
(0, 91, 2018, 98),
(0, 92, 2017, 98),
(12, 93, 2016, 60),
(5, 94, 2016, 95),
(5, 95, 2016, 58),
(11, 96, 2018, 95),
(5, 91, 2017, 85),
(1, 91, 2018, 85),
(1, 92, 2016, 85),
(4, 91, 2019, 85),
(4, 94, 2018, 85),
(4, 95, 2018, 85),
(9, 91, 2019, 85),
(4, 97, 2016, 75),
(2, 92, 2019, 75),
(2, 91, 2019, 75),
(2, 94, 2018, 75),
```

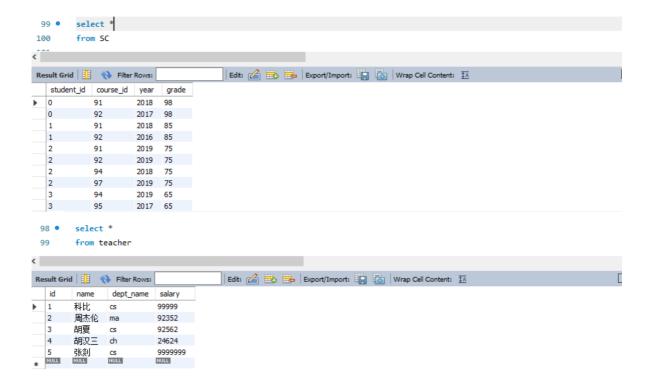
```
(2, 97, 2019, 75),
(3, 97, 2015, 75),
(6, 97, 2018, 75),
(8, 97, 2019, 65),
(6, 96, 2019, 65),
(3, 94, 2019, 65),
(3, 95, 2017, 65),
(7, 91, 2016, 65),
(7, 95, 2018, 18),
(7, 93, 2014, 21),
(10, 94, 2016, 29),
(10, 91, 2018, 95);
```

#### 3.4. 向 Teacher 表插入不少于 3 条数据。注意,应该包含姓胡的教师。

```
insert into teacher
values
(1, '科比', 'cs', 99999),
(2, '周杰伦', 'ma', 92352),
(3, '胡夏', 'cs', 92562),
(4, '胡汉三', 'ch', 24624),
(5, '张剡', 'cs', 9999999);
```

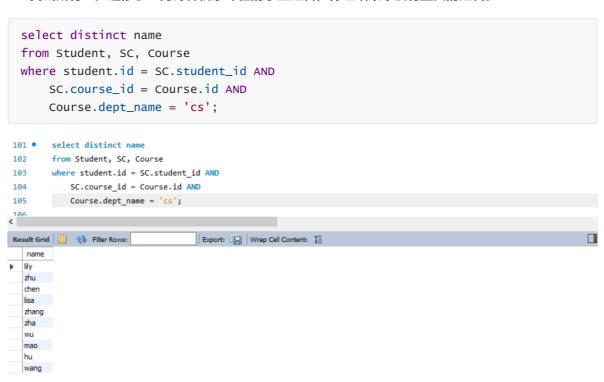
# 当前表插入后结果





## 4. 查询

4.1 找出所有至少选修了一门计算机系课程的学生姓名,保证结果中没有重复的姓名。

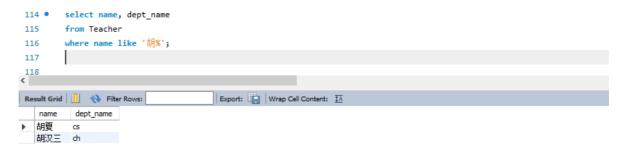


#### 让我们来验证答案

筛选出某位同学选了哪些院系的课程:

```
select distinct Course.dept_name
 from Student, SC, Course
 where Student.id = SC.student_id AND
      Course.id = SC.course_id AND
      Student.name = 'zhang';
107 • select distinct Course.dept_name
108
      from Student, SC, Course
    where Student.id = SC.student id AND
109
    Course.id = SC.course_id AND
111
       Student.name = 'zhang';
| Export: | | Wrap Cell Content: IA
dept_name
▶ cs
  ma
ma
ha
```

#### 4.2 找出所有姓胡的教师的姓名和院系。



显然可以注意到结果是正确的 🖨

### 4.3 找出所有没有选修在 2018 年之前 (不含 2018 年) 开设的任何课程的学生的 ID 和姓名。

```
select id, name
from Student
where id not in (
    select distinct SC.student_id
    from SC
    where year < 2018
);</pre>
```

```
118 • select id, name
119
      from Student
120 \ominus where id not in (
121
          select distinct SC.student_id
122
123
           where year < 2018
     );
124
                                     | Edit: 🔏 📆 🖶 | Export/Import: 📳 🚡 | Wrap Cell Content: 🏗
Result Grid 🔠 💎 Filter Rows:
  id name
  2
  6 gao
       xie
  9 mao
11 liang
```

4.4 找出每个系教师的最高工资值。可以假设每个系至少有一位教师。

```
select name, MAX(salary)
from Teacher
group by dept_name;

126  select name, MAX(salary)
127  from Teacher
128  group by dept_name;
129

Result Grid  WFilter Rows:

name MAX(salary)

科比 9999999
周杰伦 92352
胡汉三 24624
```

# 5. 修改数据

5.1. 将数据库课的学生成绩全部加 2 分。

```
update SC
set grade = grade + 2
where course_id in (
    select course.id
    from course
    where course.title = 'Database'
);
```

```
133 • update SC
134
    set grade = grade + 2
136
       select course.id
137
       from course
138
        where course.title = 'Database'
139
140
141 • select *
142 from SC;
| Edit: 🔏 📆 📙 | Export/Import: 📳 🚡 | Wrap Cell Content: 🏗
  student_id course_id year grade
               2018 100
            2017 98
 0
       92
        91
               2018 87
      92 2016 85
  1
        91
               2019 77
      92 2019 75
  2
               2018 75
       97 2019 75
  2
  3
        94
               2019 65
      95 2017 65
  3
        97
               2015 75
       97 2015 /5
91 2019 87
  4
               2018 85
  4
      95 2018 85
        97
               2016 75
       91 2017 87
  5
               2016 95
            2016 58
```

可以发现数据库课的学生在 sc 表里成绩被加了2分.

# 6. 删除数据

6.1. 删除均分不足 80 分的所有学生的选课记录。

为了验证结果的正确,首先我们可以看看均分不足80的同学有哪些

```
133 •
     select student_id, AVG(grade)
134
         from SC
135
        group by student_id
            having AVG(grade) < 80
136
<
Export: Wrap Cell Content: IA
  student_id AVG(grade)
 2
          77,0000
       70.3333
  3
          72,0000
        36.6667
  7
          67.0000
  10 64.0000
  12
          62.0000
```

## 接下来本题答案为:

```
delete from SC
where student_id in (
    select student_id from (
        select student_id
        from SC
        group by student_id
        having AVG(grade) < 80
    ) t
)</pre>
```

```
138 • delete from SC
140
         select student_id from (
141
            select student_id
142
            from SC
            group by student_id
143
            having AVG(grade) < 80
144
145
        ) t
146
147
148 •
      select *
149
       from SC
<
Result Grid 🔢 🙌 Filter Rows:
                               | Edit: 🚣 📆 🛗 | Export/Import: 🏭 🚡 | Wrap Cell Content: 🏗
  student_id course_id year grade
  0
          91
                 2018 100
               2017 100
  0
        92
                 2018 87
  1
         92
             2016 87
              2018 87
  4
                 2018 87
  4
        97
             2016 77
                 2017 87
          91
  5
             2016 97
        94
  5
          95
                 2016 60
        91 2019 87
  9
  11
        96 2018 97
```

## 7. 视图操作

#### 7.1. 创建一个视图,记录每位学生已修课程的总学分数。

```
create view Student_credit(Student_id, Student_name, sum_credit)
       as select Student.id, Student.name, SUM(Course.credit)
            from Student, SC, Course
            where Student.id = SC.student_id AND
                 SC.course_id = Course.id
            group by Student.id;
  select *
  from student_credit;
151 • create view Student_credit(Student_id, Student_name, sum_credit)
152
       as select Student.id, Student.name, SUM(Course.credit)
153
           from Student, SC, Course
           where Student.id = SC.student_id AND
             SC.course_id = Course.id
155
156
            group by Student.id;
157
158 • select *
from student_credit;
<
Export: Wrap Cell Content: 🔣
  Student_id Student_name sum_credit
        zhu
                 6
         zhang
                   13
         mao
  11
         liang
```

可以发现之前被删除的同学没有出现在这里, 易验证这里所修课程的总学分是正确的.

# 8. 删除基本表

#### 8.1. 删除 SC 表。



# 实验中遇到的困难及解决办法

复习老师上课的PPT<sup>1</sup>,以及参考教材<sup>2</sup>和其他练习题,完成本次实验不是很困难,比较不容易的部分在 *删除均分不足80 分的所有学生的选课记录*部分,因为对 having 语句的掌握不够,刚开始用了复杂的嵌套语句实现本题,后来复习了PPT以及查找练习之后找到了解决方法.

本次实验难度不是很大,建议下一次能否给出数据库,因为自己构造数据库鲁棒性不够,而且对于助教哥的批改,如果给出了数据库之后就像给出了测试用例,这样大家就有一个标准答案也比较好批改.

# 参考文献及致谢

- [1] 南京大学 2019年春季 数据库概论 课程PPT
- [2] 徐洁磐, 柏文阳, 刘奇志. 数据库系统实用教程. 高等教育出版社, 2006

谢谢助教哥的批改~