

哈尔滨工业大学

<<数据库系统>>

实验报告二

(2023 年度春季学期)

姓名:	南宫材先
学号:	L190201201
学院:	计算学部
教师:	张浩

实验二

一、实验目的

在熟练掌握 MySQL 基本命令、SQL 语言以及用 C 语言编写 MySQL 操作程序的基础上，学习简单数据库系统的设计方法，包括数据库概要设计、逻辑设计

二、实验环境

Mysql5.0,
Spring boot 2.7.10,
Jpa,
Html,
Css,
Javascript

三、实验过程及结果

实体:

- (1) student: **student_id**(学号)、student_name、gender
- (2) class: **class_id**、class_name
- (3) dormitory: **dormitory_id**、dormitory_number
- (4) teacher: **teacher_id**、teacher_name
- (5) college: **college_id**、college_name
- (6) course: **course_id**、course_name
- (7) major: **major_id**、major_name
- (8) president: **president_id**、president_name

• 联系:

- (1) 一名学生可以选择一个宿舍、专业、学校、课程、选课教室。
- (2) 一个班可以有多名学生和多名老师。
- (3) 一个宿舍可以有好几个学生。
- (4) 一名教师可以属于一所学校、课程和课程教室。
- (5) 一所学校可以有多名学生和多名老师。 ;
- (6) 一个课程可以属于多名学生和多名老师。
- (7) 一个专业可以有好几个学生。
- (8) 一名校长只能属于一所学校。

• 表 1: Student:

列名	数据类型	完整性约束	含义
student_id	varchar(255)	主键	学生学号
student_name	varchar(255)	NOT NULL	学生姓名
gender	varchar(255)	NOT NULL	学生性别
college_id	bigint	外键, NOT NULL	学校号
major_id	bigint	外键, NOT NULL	专业号
dormitory_id	bigint	外键, NOT NULL	宿舍号
class_id	bigint	外键, NOT NULL	教室号

• 表 2: class

列名	数据类型	完整性约束	含义
class_id	bigint	主键, auto_increment	教师号
class_name	varchar(255)	NOT NULL	教师名字

• 表 3: dormitory

列名	数据类型	完整性约束	含义
dormitory_id	bigint	主键, auto_increment	宿舍号
dormitory_number	varchar(255)	NOT NULL	宿舍房间号

• 表 4: teacher

列名	数据类型	完整性约束	含义
teacher_id	bigint	主键, auto_increment	宿舍号
teacher_name	varchar(255)	外键, NOT NULL	宿舍房间号

college_id	bigint	外键, NOT NULL	学校号
course_id	bigint	外键, NOT NULL	课程号
class_id	bigint	外键, NOT NULL	教室号

• 表 5: course

列名	数据类型	完整性约束	含义
course_id	bigint	主键, auto_increment	课程号
course_name	varchar(255)	NOT NULL	课程名字

• 表 6: college

列名	数据类型	完整性约束	含义
college_id	bigint	主键, auto_increment	学校号
college_name	varchar(255)	NOT NULL	学习名字

• 表 7: major

列名	数据类型	完整性约束	含义
major_id	bigint	主键, auto_increment	专业号
major_name	varchar(255)	NOT NULL	专业名字

• 表 8: president

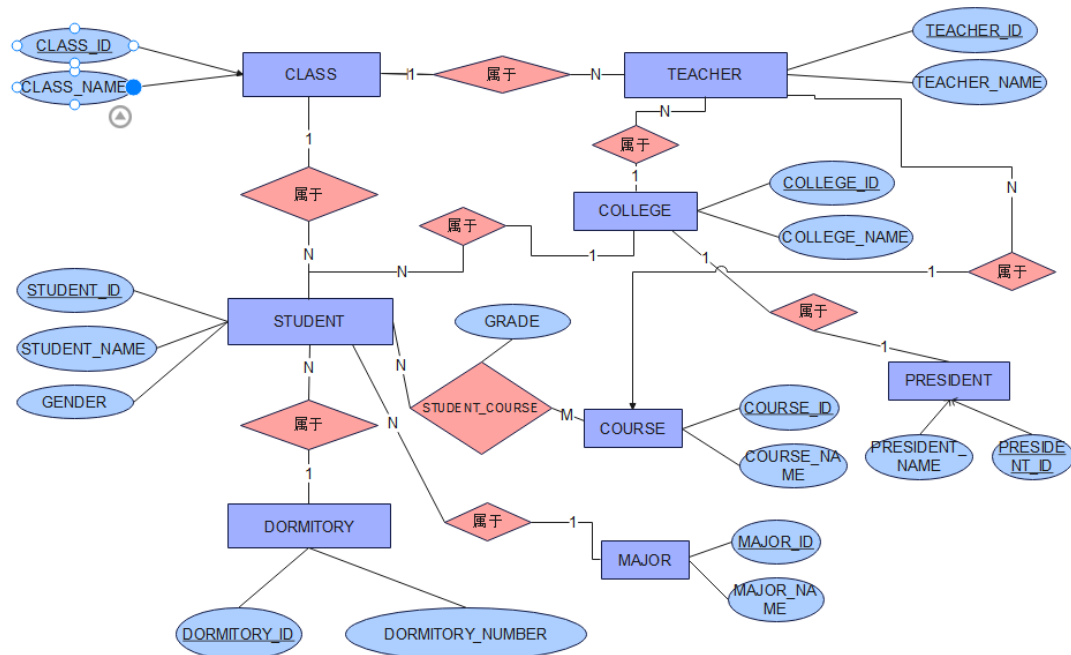
列名	数据类型	完整性约束	含义
president_id	bigint	主键, auto_increment	校长号
president_name	varchar(255)	NOT NULL	校长名字
college_id	bigint	外键, NOT NULL	学校号

• 表 9: studet_course

列名	数据类型	完整性约束	含义
student_course_id	bigint	主键, auto_increment	studet_course 号
grade	int	NULL	成绩
student_id	bigint	外键, NOT NULL	学生号
course_id	bigint	外键, NOT NULL	课程号

- Student 和 College 属于 N:1 联系
- Student 和 Major 属于 N:1 联系
- Student 和 Dormitory 属于 N:1 联系
- Student 和 Course 属于 N: M 联系
- Class 和 Teacher 属于 1:N 联系
- Teacher 和 College 属于 N:1 联系
- Teacher 和 Course 属于 N:1 联系
- College 和 President 属于 1:1 联系

绘制 E-R 图



设计逻辑数据库

Student(student_id, student_name, gender)

Class(class_id, class_name)

Dormitory(dormitory_id, dormitory_name)

Teacher(teacher_id, teacher_name)

College (college_id, college_name)

Course (course_id, course_name)

Major(major_id, major_name)

President(president_id, president_name)

Student_Course(Student_id, Course_id, gender)

数据库系统实现

Create table

```
create table class (
  class_id bigint not null auto_increment,
  class_name varchar(255) not null,
  primary key (class_id)
) engine=InnoDB
2023-03-30 09:41:17.166 INFO 10284 --- [ restartedMain] p6spy : statement 20ms
create table college (
  college_id bigint not null auto_increment,
  college_name varchar(255) not null,
  primary key (college_id)
) engine=InnoDB
2023-03-30 09:41:17.184 INFO 10284 --- [ restartedMain] p6spy : statement 17ms
create table course (
  course_id bigint not null auto_increment,
  course_name varchar(255) not null,
  primary key (course_id)
) engine=InnoDB
2023-03-30 09:41:17.202 INFO 10284 --- [ restartedMain] p6spy : statement 17ms
create table dormitory (
  dormitory_id bigint not null auto_increment,
  dormitory_number varchar(255) not null,
  primary key (dormitory_id)
) engine=InnoDB
2023-03-30 09:41:17.218 INFO 10284 --- [ restartedMain] p6spy : statement 16ms
```

```
create table major (
  major_id bigint not null auto_increment,
  major_name varchar(255) not null,
  primary key (major_id)
) engine=InnoDB
2023-03-30 09:41:17.234 INFO 10284 --- [ restartedMain] p6spy : statement 15ms
create table president (
  president_id bigint not null auto_increment,
  president_name varchar(255) not null,
  college_id bigint not null,
  primary key (president_id)
) engine=InnoDB
2023-03-30 09:41:17.252 INFO 10284 --- [ restartedMain] p6spy : statement 17ms
create table student (
  student_id varchar(255) not null,
  gender varchar(255) not null,
  student_name varchar(255) not null,
  college_id bigint not null,
  dormitory_id bigint not null,
  major_id bigint not null,
  class_id bigint not null,
  primary key (student_id)
) engine=InnoDB
2023-03-30 09:41:17.269 INFO 10284 --- [ restartedMain] p6spy : statement 15ms
```

```
create table student_course (
  student_course_id bigint not null auto_increment,
  grade integer,
  course_id bigint,
  student_id varchar(255),
  primary key (student_course_id)
) engine=InnoDB
2023-03-30 09:41:17.297 INFO 10284 --- [ restartedMain] p6spy
create table teacher (
  teacher_id bigint not null auto_increment,
  teacher_name varchar(255) not null,
  college_id bigint not null,
  course_id bigint not null,
  class_id bigint not null,
  primary key (teacher_id)
) engine=InnoDB
2023-03-30 09:41:17.345 INFO 10284 --- [ restartedMain] p6spy
alter table president
add constraint FK2dl05ew8d2ka0q8esunq676cr
foreign key (college_id)
references college (college_id)
2023-03-30 09:41:17.398 INFO 10284 --- [ restartedMain] p6spy
alter table student
add constraint FKlq9gp46xirggtyslqs0wkowu5
foreign key (college_id)
references college (college_id)
2023-03-30 09:41:17.452 INFO 10284 --- [ restartedMain] p6spy
```

```
alter table student
add constraint FKlq9gp46xirggtyslqs0wkowu5
foreign key (college_id)
references college (college_id)
2023-03-30 09:41:17.452 INFO 10284 --- [ restartedMain] p6spy
alter table student
add constraint FKcfl9c70hkakrryim7tafbtoao
foreign key (dormitory_id)
references dormitory (dormitory_id)
2023-03-30 09:41:17.507 INFO 10284 --- [ restartedMain] p6spy
alter table student
add constraint FKcml1vvjs3bcqyxdcprjrjd2o0
foreign key (major_id)
references major (major_id)
2023-03-30 09:41:17.578 INFO 10284 --- [ restartedMain] p6spy
alter table student
add constraint FKdwhkib64u47wc4yo4hk0cub90
foreign key (class_id)
references class (class_id)
2023-03-30 09:41:17.637 INFO 10284 --- [ restartedMain] p6spy
alter table student_course
add constraint FKejrkh4gv81qgmspsanaj190ws
foreign key (course_id)
references course (course_id)
2023-03-30 09:41:17.687 INFO 10284 --- [ restartedMain] p6spy
```

```
2023-03-30 09:41:17.687 INFO 10284 --- [ restartedMain] p6spy
alter table student_course
add constraint FKq7yw2wg9wlt2cnj480hcdn6dq
foreign key (student_id)
references student (student_id)
2023-03-30 09:41:17.746 INFO 10284 --- [ restartedMain] p6spy
alter table teacher
add constraint FK763n3dvgpu6dL9swenb5pc1hh
foreign key (college_id)
references college (college_id)
2023-03-30 09:41:17.797 INFO 10284 --- [ restartedMain] p6spy
alter table teacher
add constraint FKf75wvkk4ch3gnhje998bpq0lcid
foreign key (course_id)
references course (course_id)
2023-03-30 09:41:17.866 INFO 10284 --- [ restartedMain] p6spy
alter table teacher
add constraint FK6ehoggb8t4jrwovqt1w3v5a6w
foreign key (class_id)
references class (class_id)
```

[Home](#)

学校信息

学生登入

学生信息

老师登入

老师信息

校长登入

校长信息

按下重复检查时, 用 Student__id 在数据库中查询值。 如果有值, 就判断学生已经存在, 发送警告窗口, 如果没有值, 就通过 insert 门输入学生信息, 移动到 home 页面。如有空白, 出示警告窗口后, 防止使用 insert 语法

The figure consists of four screenshots of a web application interface, arranged in a 2x2 grid. Each screenshot shows a registration form with fields for '学号' (Student ID), '学生姓名' (Student Name), '性别' (Gender), '班名字' (Class Name), '学效名字' (Academic Name), '宿舍' (Dormitory), and '专业' (Major). A 'Submit' button is at the bottom of the form. A '重复检查' (Repeat Check) button is located above the '学号' field.

- Top-left screenshot:** The '学号' field contains 'L190201201'. The '重复检查' button is highlighted in blue. The form is otherwise empty.
- Top-right screenshot:** A modal window titled 'localhost:9090 내용:' is displayed over the form. It contains the text '已经存在的学号.' (Student ID already exists.) and a green '확인' (Confirm) button.
- Bottom-left screenshot:** The '学号' field contains 'L190201202'. The '重复检查' button is highlighted in blue. A modal window titled 'localhost:9090 내용:' is displayed, containing the text '可以使用学号' (Can use student ID) and a green '확인' (Confirm) button.
- Bottom-right screenshot:** The '学号' field contains 'L190201201'. The '重复检查' button is highlighted in blue. A modal window titled 'localhost:9090 내용:' is displayed, containing the text '오류' (Error) and a green '확인' (Confirm) button.

重复检查时使用的 Select 文法


```
select
  *
from
  student
where
  student_id = 'L190201201'
```

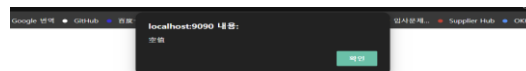
```
select
  *
from
  student
where
  student_id = 'L190201202'
```

信息输入完成后，将使用 insert 文法。

```
select
  student0_.student_id as student_1_6_0_,
  student0_.college_id as college_4_6_0_,
  student0_.dormitory_id as dormitor5_6_0_,
  student0_.gender as gender2_6_0_,
  student0_.major_id as major_id6_6_0_,
  student0_.class_id as class_id7_6_0_,
  student0_.student_name as student_3_6_0_
from
  student student0_
where
  student0_.student_id='L190201202'
2023-03-30 15:11:41.483 INFO 10284 --- [nio-9090-exec-9] p6spy
insert
into
  student
  (college_id, dormitory_id, gender, major_id, class_id, student_name, student_id)
values
  (3, 2, '男子', 2, 2, '南宮材先', 'L190201202')
```

进入学生信息,使用 select 和 Join Query 文法。使用 joinquery 一次性查询 student、class、college、dormitory、major。如果没有值,则显示警告窗口,然后转到 home 页面。

专业 GroupBy						
学号	学生姓名	性别	班名字	学效名字	宿舍	专业
111115550	홍길동	남자	计算机C班	哈尔滨工业大学	A301	计算机
201115066	李	女子	计算机B	哈尔滨工业大学	B220	计算机
556165151	남궁재선	男子	计算机D班	哈尔滨工业大学	B101	中文
L18001515	金啊啊	男子	中文C班	哈尔滨工业大学	A202	中文
L190201202	南宮材先	男子	计算机A班	哈尔滨工业大学	A301	计算机



```
select
  student_id as studentId,
  student_name as studentName ,
  gender,
  class_name as className,
  dormitory_number as dNumber,
  college_name as collegeName,
  major_name as majorName
from
  student s
join
  class c
    on s.class_id= c.class_id
join
  dormitory d
    on s.dormitory_id = d.dormitory_id
join
  college co
    on s.college_id = co.college_id
join
  major m
    on s.major_id = m.major_id
```

点击"专业 Groupby"，使用 GroupBy 文法按组查询专业学生人数。

专业	总共
计算机	3
中文	2

```
select
    count(*) as count,
    m.major_name as majorName
from
    student s
join
    major m
    on s.major_id = m.major_id
group by
    m.major_name
```

点击专业按钮, 使用嵌套查询文法, 查询专业的学生姓名。

[Home](#)

专业	学号	姓名
计算机	111115550	홍길동
计算机	201115066	李
计算机	L190201202	南宫材先

```
select
    student_id as studentId,
    student_name as studentName
from
    student
where
    major_id in(
        select
            major_id
        from
            major
        where
            major_name = '计算机'
    )
```

按下 delete, delete query 文法被使用, 学生信息被删除。

专业 GroupBy

学号	学生姓名	性别	班名字	学效名字	宿舍	专业	
201115066	李	女子	计算机B班	哈尔滨工业大学	B220	计算机	delete
556165151	남궁재선	남자	计算机D班	哈尔滨工业大学	B101	中文	delete
L18001515	李明明	男子	中文C班	哈尔滨工业大学	A202	中文	delete
L190201202	南宫材先	男子	计算机A班	哈尔滨工业大学	A301	计算机	delete

```
delete
from
    major
where
    major_id=1
2023-03-30 15:47:53.605 INFO 4920 --- [nio-9090-exec-8] pdsy : statement 0ms

delete
from
    class
where
    class_id=1
2023-03-30 15:47:53.606 INFO 4920 --- [nio-9090-exec-8] pdsy : statement 0ms

delete
from
    college
where
    college_id=1
```

四、实验心得

通过使用数据库,我感受到了很多实际数据库如何使用,并了解到使用复杂的Query 可以获得更详细的信息。。