

《数据库原理与技术》上机报告

数据库设计

|  |  |  |
| --- | --- | --- |
| **姓名** |  |  |
| **学号** |  |  |
| **专业班级** |  |  |
| **指导教师** |  |  |
| **报告得分** |  |  |

2024年12月20日

**一、实验目的与要求**

实验运用数据库技术构建高校宿舍管理系统，实现对学生宿舍信息、学生信息、宿舍分配、报修管理等功能的有效管理，旨在帮助学生深入理解数据库管理系统的基本概念、原理和操作，熟练掌握数据库设计、创建、数据操作以及数据查询等关键技能，培养学生运用数据库技术解决实际问题的能力，并提高学生的动手实践和独立思考能力。

**二、实验内容与步骤**

**1、数据库设计**

1. 需求分析

**（一）学生信息管理**

* **学生基本信息录入**：包括姓名、性别、学号、专业、班级、联系方式等，支持批量导入功能。
* **信息查询与修改**：可根据学号、姓名等关键词快速查询学生信息，并对信息进行修改、更新，如联系方式变更、专业调整等。
* **学生入住与毕业退房办理**：记录学生入住时间、宿舍分配信息，在毕业退房时进行相关费用的查询和结算、信息登记。

**（二）宿舍信息管理**

* **宿舍基本信息维护**：如宿舍编号、楼层、房间类型（单人间、多人间等）、可容纳人数、家具及设施配备情况等。
* **宿舍分配与调整**：根据学生性别、专业、年级等因素进行宿舍分配，可手动调整宿舍安排，并记录分配历史。
* **空宿舍查询**：方便管理人员查看当前空宿舍情况，以便安排新生入住或进行宿舍调整。

**（三）设施设备管理**

* **设施设备登记**：对宿舍内的家具、电器、门窗等设施设备进行详细登记，包括品牌、型号、购置时间、使用状态等。
* **设施设备维修管理**：学生可在线申报设施设备损坏情况，管理人员接收申报后安排维修人员进行维修，并记录维修进度、维修费用、维修人员等信息，维修完成后进行验收。
* **设施设备报废处理**：对于无法修复或达到使用年限的设施设备，进行报废登记，记录报废原因、处理方式等。

**（四）费用管理**

* **费用项目设定**：包括住宿费、水电费、物业费等，可设置不同费用的收费标准、计费周期。
* **费用收缴记录**：记录学生缴费时间、缴费金额、缴费方式等信息，支持多种缴费方式（如线上支付、线下缴费）的对接与数据录入。
* **欠费提醒与催缴**：系统自动生成欠费学生名单，并通过短信、系统消息等方式提醒学生缴费，可打印催缴通知单。

**（五）访客管理**

* **访客登记**：访客进入公寓时，需登记姓名、身份证号、联系方式、访问对象、访问时间等信息，可通过扫描身份证或手动录入。
* **访客权限设置**：根据宿舍学生的授权，设置访客的访问权限，如可访问的楼层、宿舍范围、访问时长等。
* **访客记录查询与统计**：管理人员可查询历史访客记录，统计访客流量、访问高峰时段等信息，以便加强安全管理。

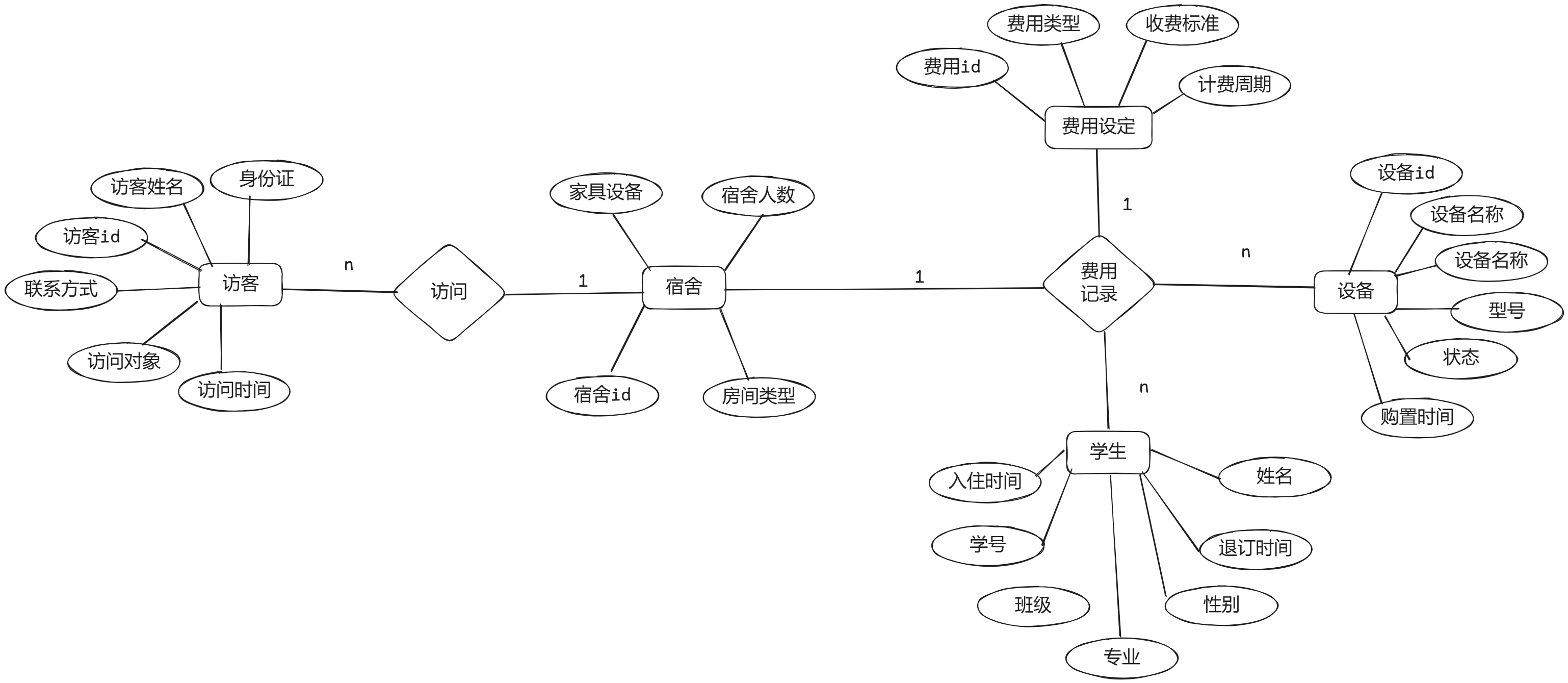
**（六）系统管理**

* **用户管理**：创建不同角色的用户账号，如管理员、宿管人员、学生等，并分配相应的操作权限，如管理员拥有系统的全面管理权限，宿管人员负责所在楼栋的日常管理操作，学生可进行个人信息查询、费用查询、维修申报等操作。
* **数据备份与恢复**：定期对系统数据进行备份，确保数据的安全性和完整性，在数据丢失或损坏时可进行数据恢复操作。
* **日志管理**：记录系统操作日志，包括用户登录时间、操作内容、操作结果等信息，以便进行系统操作审计与问题追溯。

**（七）数据加密**

* **数据加密：**对学生敏感信息（如身份证号、联系方式等）、费用信息等进行加密存储和传输，防止数据泄露。

1. E-R图



1. 逻辑结构设计

### **模块一：学生信息管理**

| **表名** | **字段名** | **数据类型** | **描述** |
| --- | --- | --- | --- |
| students | student\_id | INT (PK) AUTO\_INCREMENT | 学号，主键 |
|  | name | VARCHAR(50) | 学生姓名 |
|  | gender | ENUM('男', '女') | 性别 |
|  | major | VARCHAR(100) | 专业 |
|  | class | VARCHAR(50) | 班级 |
|  | contact | VARCHAR(15) | 联系方式 |
|  | dorm\_id | INT (FK) | 宿舍编号（外键） |
|  | check\_in\_date | DATE | 入住时间 |
|  | check\_out\_date | DATE | 退房时间 |
|  | encrypted\_contact | TEXT | 加密的联系方式 |

### **模块二：宿舍信息管理**

| **表名** | **字段名** | **数据类型** | **描述** |
| --- | --- | --- | --- |
| dormitories | dorm\_id | INT (PK) AUTO\_INCREMENT | 宿舍编号，主键 |
|  | floor | INT | 宿舍楼层 |
|  | room\_type | ENUM('单人间', '双人间', '多人间') | 房间类型 |
|  | capacity | INT | 可容纳人数 |
|  | current\_occupants | INT | 当前入住人数 |
|  | furniture\_info | TEXT | 家具及设施配备信息 |
|  | allocation\_history | TEXT | 分配历史记录 |

### **模块三：设施设备管理**

| **表名** | **字段名** | **数据类型** | **描述** |
| --- | --- | --- | --- |
| facilities | facility\_id | INT (PK) AUTO\_INCREMENT | 设备编号，主键 |
|  | dorm\_id | INT (FK) | 宿舍编号（外键） |
|  | name | VARCHAR(100) | 设备名称 |
|  | brand | VARCHAR(50) | 品牌 |
|  | model | VARCHAR(50) | 型号 |
|  | purchase\_date | DATE | 购置时间 |
|  | status | ENUM('正常', '损坏', '报废') | 设备状态 |

| **表名** | **字段名** | **数据类型** | **描述** |
| --- | --- | --- | --- |
| maintenance\_requests | request\_id | INT (PK) AUTO\_INCREMENT | 维修申请编号，主键 |
|  | facility\_id | INT (FK) | 设备编号（外键） |
|  | report\_date | DATE | 报修日期 |
|  | description | TEXT | 问题描述 |
|  | status | ENUM('待处理', '维修中', '已完成') | 维修状态 |
|  | repair\_date | DATE | 维修完成日期 |
|  | repair\_cost | DECIMAL(10, 2) | 维修费用 |
|  | repair\_person | VARCHAR(50) | 维修人员 |

| **表名** | **字段名** | **数据类型** | **描述** |
| --- | --- | --- | --- |
| disposal\_records | record\_id | INT (PK) AUTO\_INCREMENT | 报废记录编号，主键 |
|  | facility\_id | INT (FK) | 设备编号（外键） |
|  | disposal\_date | DATE | 报废日期 |
|  | reason | TEXT | 报废原因 |

### **模块四：费用管理**

| **表名** | **字段名** | **数据类型** | **描述** |
| --- | --- | --- | --- |
| fee\_settings | fee\_id | INT (PK) AUTO\_INCREMENT | 费用编号，主键 |
|  | fee\_type | ENUM('住宿费', '水电费', '物业费') | 费用类型 |
|  | amount | DECIMAL(10, 2) | 收费标准 |
|  | billing\_cycle | ENUM('按月', '按学期', '按年') | 计费周期 |

| **表名** | **字段名** | **数据类型** | **描述** |
| --- | --- | --- | --- |
| fee\_records | record\_id | INT (PK) AUTO\_INCREMENT | 缴费记录编号，主键 |
|  | student\_id | INT (FK) | 学号（外键） |
|  | fee\_id | INT (FK) | 费用编号（外键） |
|  | payment\_date | DATE | 缴费时间 |
|  | amount | DECIMAL(10, 2) | 缴费金额 |
|  | payment\_method | ENUM('线上支付', '线下缴费') | 缴费方式 |

### **模块五：访客管理**

| **表名** | **字段名** | **数据类型** | **描述** |
| --- | --- | --- | --- |
| visitors | visitor\_id | INT (PK) AUTO\_INCREMENT | 访客编号，主键 |
|  | name | VARCHAR(50) | 访客姓名 |
|  | id\_card | VARCHAR(18) | 身份证号 |
|  | contact | VARCHAR(15) | 联系方式 |
|  | visit\_target | INT (FK) | 访问对象（学生编号） |
|  | visit\_time | DATETIME | 访问时间 |
|  | access\_level | TEXT | 访问权限范围 |
|  | duration | INT | 访问时长（分钟） |

### **模块六：系统管理**

| **表名** | **字段名** | **数据类型** | **描述** |
| --- | --- | --- | --- |
| users | user\_id | INT (PK) AUTO\_INCREMENT | 用户编号，主键 |
|  | username | VARCHAR(50) | 用户名 |
|  | password\_hash | TEXT | 密码（哈希值） |
|  | role | ENUM('管理员', '宿管人员', '学生') | 用户角色 |
|  | permissions | TEXT | 用户权限列表 |

| **表名** | **字段名** | **数据类型** | **描述** |
| --- | --- | --- | --- |
| logs | log\_id | INT (PK) AUTO\_INCREMENT | 日志编号，主键 |
|  | user\_id | INT (FK) | 操作用户编号 |
|  | action | TEXT | 操作内容 |
|  | action\_time | DATETIME | 操作时间 |
|  | result | ENUM('成功', '失败') | 操作结果 |

1. 物理结构设计

**2、数据库创建与数据加载**

CREATE TABLE dormitories (

dorm\_id INT PRIMARY KEY AUTO\_INCREMENT COMMENT '宿舍编号',

floor INT NOT NULL COMMENT '楼层',

room\_type ENUM('单人间', '双人间', '多人间') NOT NULL COMMENT '房间类型',

capacity INT NOT NULL COMMENT '可容纳人数',

current\_occupants INT DEFAULT 0 COMMENT '当前入住人数',

furniture\_info TEXT COMMENT '家具及设施配备',

allocation\_history TEXT COMMENT '分配历史记录'

);

CREATE TABLE students (

student\_id INT PRIMARY KEY AUTO\_INCREMENT COMMENT '学号',

name VARCHAR(50) NOT NULL COMMENT '姓名',

gender ENUM('男', '女') NOT NULL COMMENT '性别',

major VARCHAR(100) COMMENT '专业',

class VARCHAR(50) COMMENT '班级',

contact VARCHAR(15) NOT NULL COMMENT '联系方式',

dorm\_id INT DEFAULT NULL COMMENT '宿舍编号',

check\_in\_date DATE DEFAULT NULL COMMENT '入住时间',

check\_out\_date DATE DEFAULT NULL COMMENT '退房时间',

encrypted\_contact TEXT COMMENT '加密的联系方式',

FOREIGN KEY (dorm\_id) REFERENCES dormitories(dorm\_id)

);

CREATE TABLE facilities (

facility\_id INT PRIMARY KEY AUTO\_INCREMENT COMMENT '设备编号',

dorm\_id INT NOT NULL COMMENT '宿舍编号',

name VARCHAR(100) NOT NULL COMMENT '设备名称',

brand VARCHAR(50) COMMENT '品牌',

model VARCHAR(50) COMMENT '型号',

purchase\_date DATE COMMENT '购置时间',

status ENUM('正常', '损坏', '报废') DEFAULT '正常' COMMENT '状态',

FOREIGN KEY (dorm\_id) REFERENCES dormitories(dorm\_id)

);

CREATE TABLE maintenance\_requests (

request\_id INT PRIMARY KEY AUTO\_INCREMENT COMMENT '申请编号',

facility\_id INT NOT NULL COMMENT '设备编号',

report\_date DATE NOT NULL COMMENT '报修日期',

description TEXT COMMENT '问题描述',

status ENUM('待处理', '维修中', '已完成') DEFAULT '待处理' COMMENT '状态',

repair\_date DATE DEFAULT NULL COMMENT '完成日期',

repair\_cost DECIMAL(10, 2) DEFAULT NULL COMMENT '维修费用',

repair\_person VARCHAR(50) DEFAULT NULL COMMENT '维修人员',

FOREIGN KEY (facility\_id) REFERENCES facilities(facility\_id)

);

CREATE TABLE fee\_settings (

fee\_id INT PRIMARY KEY AUTO\_INCREMENT COMMENT '费用编号',

fee\_type ENUM('住宿费', '水电费', '物业费') NOT NULL COMMENT '费用类型',

amount DECIMAL(10, 2) NOT NULL COMMENT '收费标准',

billing\_cycle ENUM('按月', '按学期', '按年') NOT NULL COMMENT '计费周期'

);

CREATE TABLE fee\_records (

record\_id INT PRIMARY KEY AUTO\_INCREMENT COMMENT '记录编号',

student\_id INT NOT NULL COMMENT '学号',

fee\_id INT NOT NULL COMMENT '费用编号',

payment\_date DATE DEFAULT NULL COMMENT '缴费时间',

amount DECIMAL(10, 2) NOT NULL COMMENT '缴费金额',

payment\_method ENUM('线上支付', '线下缴费') NOT NULL COMMENT '缴费方式',

FOREIGN KEY (student\_id) REFERENCES students(student\_id),

FOREIGN KEY (fee\_id) REFERENCES fee\_settings(fee\_id)

);

CREATE TABLE visitors (

visitor\_id INT PRIMARY KEY AUTO\_INCREMENT COMMENT '访客编号',

name VARCHAR(50) NOT NULL COMMENT '姓名',

id\_card VARCHAR(18) NOT NULL COMMENT '身份证号',

contact VARCHAR(15) NOT NULL COMMENT '联系方式',

visit\_target INT NOT NULL COMMENT '访问对象（学生编号）',

visit\_time DATETIME NOT NULL COMMENT '访问时间',

access\_level TEXT COMMENT '访问权限',

duration INT COMMENT '访问时长（分钟）',

FOREIGN KEY (visit\_target) REFERENCES students(student\_id)

);

CREATE TABLE users (

user\_id INT PRIMARY KEY AUTO\_INCREMENT COMMENT '用户编号',

username VARCHAR(50) NOT NULL COMMENT '用户名',

password\_hash TEXT NOT NULL COMMENT '密码（哈希值存储）',

role ENUM('管理员', '宿管人员', '学生') NOT NULL COMMENT '角色',

permissions TEXT COMMENT '权限列表'

);

CREATE TABLE logs (

log\_id INT PRIMARY KEY AUTO\_INCREMENT COMMENT '日志编号',

user\_id INT NOT NULL COMMENT '操作用户编号',

action TEXT NOT NULL COMMENT '操作内容',

action\_time DATETIME NOT NULL COMMENT '操作时间',

result ENUM('成功', '失败') NOT NULL COMMENT '操作结果',

FOREIGN KEY (user\_id) REFERENCES users(user\_id)

);

INSERT INTO dormitories (floor, room\_type, capacity, current\_occupants, furniture\_info, allocation\_history)

VALUES

(1, '单人间', 1, 0, '床, 书桌, 衣柜', ''),

(1, '双人间', 2, 1, '床, 书桌, 衣柜', '张三'),

(2, '多人间', 4, 2, '床, 书桌, 衣柜', '张三, 李四'),

(3, '单人间', 1, 0, '床, 书桌, 衣柜', ''),

(3, '双人间', 2, 2, '床, 书桌, 衣柜', '王五, 赵六'),

(4, '多人间', 6, 4, '床, 书桌, 衣柜', '多人'),

(5, '单人间', 1, 1, '床, 书桌, 衣柜', '李七'),

(6, '双人间', 2, 0, '床, 书桌, 衣柜', ''),

(7, '多人间', 8, 6, '床, 书桌, 衣柜', '多人'),

(8, '单人间', 1, 0, '床, 书桌, 衣柜', ''),

(9, '双人间', 2, 1, '床, 书桌, 衣柜', '刘八'),

(10, '多人间', 4, 3, '床, 书桌, 衣柜', '多人'),

(11, '单人间', 1, 1, '床, 书桌, 衣柜', '陈九'),

(12, '双人间', 2, 2, '床, 书桌, 衣柜', '周十, 钱十一'),

(13, '多人间', 6, 4, '床, 书桌, 衣柜', '多人'),

(14, '单人间', 1, 0, '床, 书桌, 衣柜', ''),

(15, '双人间', 2, 1, '床, 书桌, 衣柜', '孙十二'),

(16, '多人间', 8, 6, '床, 书桌, 衣柜', '多人'),

(17, '单人间', 1, 0, '床, 书桌, 衣柜', ''),

(18, '双人间', 2, 2, '床, 书桌, 衣柜', '李十三, 王十四');

INSERT INTO students (name, gender, major, class, contact, dorm\_id, check\_in\_date, check\_out\_date, encrypted\_contact)

VALUES

('张三', '男', '计算机科学', '2023级1班', '12345678901', 1, '2023-09-01', NULL, 'encrypted\_1'),

('李四', '女', '软件工程', '2023级2班', '12345678902', 2, '2023-09-02', NULL, 'encrypted\_2'),

('王五', '男', '土木工程', '2023级3班', '12345678903', 3, '2023-09-03', NULL, 'encrypted\_3'),

('赵六', '女', '机械工程', '2023级4班', '12345678904', 3, '2023-09-03', NULL, 'encrypted\_4'),

('李七', '男', '金融学', '2023级5班', '12345678905', 5, '2023-09-04', NULL, 'encrypted\_5'),

('刘八', '女', '法学', '2023级6班', '12345678906', 9, '2023-09-05', NULL, 'encrypted\_6'),

('陈九', '男', '电子信息', '2023级7班', '12345678907', 11, '2023-09-06', NULL, 'encrypted\_7'),

('周十', '女', '医学', '2023级8班', '12345678908', 12, '2023-09-07', NULL, 'encrypted\_8'),

('钱十一', '男', '护理学', '2023级9班', '12345678909', 12, '2023-09-08', NULL, 'encrypted\_9'),

('孙十二', '女', '会计学', '2023级10班', '12345678910', 15, '2023-09-09', NULL, 'encrypted\_10'),

('李十三', '男', '教育学', '2023级11班', '12345678911', 18, '2023-09-10', NULL, 'encrypted\_11'),

('王十四', '女', '心理学', '2023级12班', '12345678912', 18, '2023-09-11', NULL, 'encrypted\_12'),

('王十五', '男', '化学', '2023级13班', '12345678913', NULL, NULL, NULL, 'encrypted\_13'),

('张十六', '女', '物理学', '2023级14班', '12345678914', NULL, NULL, NULL, 'encrypted\_14'),

('李十七', '男', '数学', '2023级15班', '12345678915', NULL, NULL, NULL, 'encrypted\_15'),

('赵十八', '女', '哲学', '2023级16班', '12345678916', NULL, NULL, NULL, 'encrypted\_16'),

('陈十九', '男', '历史学', '2023级17班', '12345678917', NULL, NULL, NULL, 'encrypted\_17'),

('周二十', '女', '生物学', '2023级18班', '12345678918', NULL, NULL, NULL, 'encrypted\_18'),

('孙二十一', '男', '地理学', '2023级19班', '12345678919', NULL, NULL, NULL, 'encrypted\_19'),

('钱二十二', '女', '天文学', '2023级20班', '12345678920', NULL, NULL, NULL, 'encrypted\_20');

INSERT INTO facilities (dorm\_id, name, brand, model, purchase\_date, status)

VALUES

(1, '空调', '格力', 'KFR-35GW', '2022-01-01', '正常'),

(2, '热水器', '美的', 'JSQ22-12HWB', '2022-02-01', '正常'),

(3, '洗衣机', '海尔', 'XQG70-1010', '2022-03-01', '正常'),

(4, '冰箱', '西门子', 'BCD-610W', '2022-04-01', '损坏'),

(5, '电风扇', '艾美特', 'FSW65T', '2022-05-01', '正常'),

(6, '微波炉', '松下', 'NN-DF383B', '2022-06-01', '正常'),

(7, '电视', '创维', '55G8S', '2022-07-01', '正常'),

(8, '吸尘器', '戴森', 'V10', '2022-08-01', '报废'),

(9, '路由器', '华为', 'AX3', '2022-09-01', '正常'),

(10, '打印机', '佳能', 'G3800', '2022-10-01', '正常'),

(1, '空调', '格力', 'KFR-35GW', '2022-01-01', '正常'),

(2, '热水器', '美的', 'JSQ22-12HWB', '2022-02-01', '正常'),

(3, '洗衣机', '海尔', 'XQG70-1010', '2022-03-01', '正常'),

(4, '冰箱', '西门子', 'BCD-610W', '2022-04-01', '损坏'),

(5, '电风扇', '艾美特', 'FSW65T', '2022-05-01', '正常'),

(6, '微波炉', '松下', 'NN-DF383B', '2022-06-01', '正常'),

(7, '电视', '创维', '55G8S', '2022-07-01', '正常'),

(8, '吸尘器', '戴森', 'V10', '2022-08-01', '报废'),

(9, '路由器', '华为', 'AX3', '2022-09-01', '正常'),

(10, '打印机', '佳能', 'G3800', '2022-10-01', '正常'),

INSERT INTO students (name, gender, major, class, contact, dorm\_id, check\_in\_date, check\_out\_date, encrypted\_contact)

VALUES

('张三', '男', '计算机科学', '2023级1班', '12345678901', 1, '2023-09-01', NULL, 'encrypted\_1'),

('李四', '女', '软件工程', '2023级2班', '12345678902', 2, '2023-09-02', NULL, 'encrypted\_2'),

('王五', '男', '土木工程', '2023级3班', '12345678903', 3, '2023-09-03', NULL, 'encrypted\_3'),

('赵六', '女', '机械工程', '2023级4班', '12345678904', 3, '2023-09-03', NULL, 'encrypted\_4'),

('李七', '男', '金融学', '2023级5班', '12345678905', 5, '2023-09-04', NULL, 'encrypted\_5'),

('刘八', '女', '法学', '2023级6班', '12345678906', 9, '2023-09-05', NULL, 'encrypted\_6'),

('陈九', '男', '电子信息', '2023级7班', '12345678907', 11, '2023-09-06', NULL, 'encrypted\_7'),

('周十', '女', '医学', '2023级8班', '12345678908', 12, '2023-09-07', NULL, 'encrypted\_8'),

('钱十一', '男', '护理学', '2023级9班', '12345678909', 12, '2023-09-08', NULL, 'encrypted\_9'),

('孙十二', '女', '会计学', '2023级10班', '12345678910', 15, '2023-09-09', NULL, 'encrypted\_10'),

('李十三', '男', '教育学', '2023级11班', '12345678911', 18, '2023-09-10', NULL, 'encrypted\_11'),

('王十四', '女', '心理学', '2023级12班', '12345678912', 18, '2023-09-11', NULL, 'encrypted\_12'),

('王十五', '男', '化学', '2023级13班', '12345678913', NULL, NULL, NULL, 'encrypted\_13'),

('张十六', '女', '物理学', '2023级14班', '12345678914', NULL, NULL, NULL, 'encrypted\_14'),

('李十七', '男', '数学', '2023级15班', '12345678915', NULL, NULL, NULL, 'encrypted\_15'),

('赵十八', '女', '哲学', '2023级16班', '12345678916', NULL, NULL, NULL, 'encrypted\_16'),

('陈十九', '男', '历史学', '2023级17班', '12345678917', NULL, NULL, NULL, 'encrypted\_17'),

('周二十', '女', '生物学', '2023级18班', '12345678918', NULL, NULL, NULL, 'encrypted\_18'),

('孙二十一', '男', '地理学', '2023级19班', '12345678919', NULL, NULL, NULL, 'encrypted\_19'),

('钱二十二', '女', '天文学', '2023级20班', '12345678920', NULL, NULL, NULL, 'encrypted\_20');

INSERT INTO maintenance\_requests (facility\_id, report\_date, description, status, repair\_date, repair\_cost, repair\_person)

VALUES

(1, '2023-10-01', '空调不制冷', '待处理', NULL, NULL, NULL),

(2, '2023-10-02', '洗衣机漏水', '维修中', NULL, NULL, '李师傅'),

(3, '2023-10-03', '热水器无法加热', '已完成', '2023-10-05', 150.00, '张师傅'),

(4, '2023-10-04', '冰箱噪音大', '维修中', NULL, NULL, '王师傅'),

(4, '2023-10-04', '冰箱噪音大', '维修中', NULL, NULL, '王师傅'),

(1, '2023-10-01', '空调不制冷', '待处理', NULL, NULL, NULL),

(2, '2023-10-02', '洗衣机漏水', '维修中', NULL, NULL, '李师傅'),

(3, '2023-10-03', '热水器无法加热', '已完成', '2023-10-05', 150.00, '张师傅'),

(1, '2023-10-01', '空调不制冷', '待处理', NULL, NULL, NULL),

(2, '2023-10-02', '洗衣机漏水', '维修中', NULL, NULL, '李师傅'),

(3, '2023-10-03', '热水器无法加热', '已完成', '2023-10-05', 150.00, '张师傅'),

(4, '2023-10-04', '冰箱噪音大', '维修中', NULL, NULL, '王师傅'),

(1, '2023-10-01', '空调不制冷', '待处理', NULL, NULL, NULL),

(2, '2023-10-02', '洗衣机漏水', '维修中', NULL, NULL, '李师傅'),

(3, '2023-10-03', '热水器无法加热', '已完成', '2023-10-05', 150.00, '张师傅'),

(4, '2023-10-04', '冰箱噪音大', '维修中', NULL, NULL, '王师傅'),

(1, '2023-10-01', '空调不制冷', '待处理', NULL, NULL, NULL),

(2, '2023-10-02', '洗衣机漏水', '维修中', NULL, NULL, '李师傅'),

(3, '2023-10-03', '热水器无法加热', '已完成', '2023-10-05', 150.00, '张师傅'),

(4, '2023-10-04', '冰箱噪音大', '维修中', NULL, NULL, '王师傅'),

(5, '2023-10-05', '电风扇无法启动', '已完成', '2023-10-06', 50.00, '刘师傅');

INSERT INTO fee\_settings (fee\_type, amount, billing\_cycle)

VALUES

('住宿费', 1200.00, '按学期'),

('水电费', 50.00, '按月'),

('物业费', 30.00, '按月'),

('住宿费', 1200.00, '按学期'),

('水电费', 50.00, '按月'),

('物业费', 30.00, '按月'),

('住宿费', 2400.00, '按年'),

('住宿费', 1200.00, '按学期'),

('水电费', 50.00, '按月'),

('物业费', 30.00, '按月'),

('住宿费', 2400.00, '按年'),

('住宿费', 1200.00, '按学期'),

('水电费', 50.00, '按月'),

('物业费', 30.00, '按月'),

('住宿费', 2400.00, '按年'),

('住宿费', 1200.00, '按学期'),

('水电费', 50.00, '按月'),

('物业费', 30.00, '按月'),

('住宿费', 2400.00, '按年'),

('住宿费', 2400.00, '按年'),

('水电费', 100.00, '按学期');

INSERT INTO fee\_records (student\_id, fee\_id, payment\_date, amount, payment\_method)

VALUES

(1, 1, '2023-09-01', 1200.00, '线上支付'),

(2, 2, '2023-09-02', 50.00, '线下缴费'),

(3, 3, '2023-09-03', 30.00, '线上支付'),

(4, 1, '2023-09-04', 1200.00, '线下缴费'),

(1, 1, '2023-09-01', 1200.00, '线上支付'),

(2, 2, '2023-09-02', 50.00, '线下缴费'),

(3, 3, '2023-09-03', 30.00, '线上支付'),

(4, 1, '2023-09-04', 1200.00, '线下缴费'),

(1, 1, '2023-09-01', 1200.00, '线上支付'),

(2, 2, '2023-09-02', 50.00, '线下缴费'),

(3, 3, '2023-09-03', 30.00, '线上支付'),

(4, 1, '2023-09-04', 1200.00, '线下缴费'),

(1, 1, '2023-09-01', 1200.00, '线上支付'),

(2, 2, '2023-09-02', 50.00, '线下缴费'),

(3, 3, '2023-09-03', 30.00, '线上支付'),

(4, 1, '2023-09-04', 1200.00, '线下缴费'),

(1, 1, '2023-09-01', 1200.00, '线上支付'),

(2, 2, '2023-09-02', 50.00, '线下缴费'),

(3, 3, '2023-09-03', 30.00, '线上支付'),

(4, 1, '2023-09-04', 1200.00, '线下缴费'),

(5, 4, NULL, 2400.00, '线上支付');

INSERT INTO visitors (name, id\_card, contact, visit\_target, visit\_time, access\_level, duration)

VALUES

('李明', '110101199001011234', '13812345678', 1, '2023-10-10 10:00:00', '宿舍楼1', 60),

('王芳', '120202198902021234', '13987654321', 2, '2023-10-11 14:00:00', '宿舍楼2', 120),

('张强', '130303197003031234', '13765432109', 3, '2023-10-12 09:00:00', '宿舍楼3', 30),

('赵云', '140404196004041234', '13876543210', 4, '2023-10-13 15:00:00', '宿舍楼4', 90),

('李明', '110101199001011234', '13812345678', 1, '2023-10-10 10:00:00', '宿舍楼1', 60),

('王芳', '120202198902021234', '13987654321', 2, '2023-10-11 14:00:00', '宿舍楼2', 120),

('张强', '130303197003031234', '13765432109', 3, '2023-10-12 09:00:00', '宿舍楼3', 30),

('赵云', '140404196004041234', '13876543210', 4, '2023-10-13 15:00:00', '宿舍楼4', 90),

('李明', '110101199001011234', '13812345678', 1, '2023-10-10 10:00:00', '宿舍楼1', 60),

('王芳', '120202198902021234', '13987654321', 2, '2023-10-11 14:00:00', '宿舍楼2', 120),

('张强', '130303197003031234', '13765432109', 3, '2023-10-12 09:00:00', '宿舍楼3', 30),

('赵云', '140404196004041234', '13876543210', 4, '2023-10-13 15:00:00', '宿舍楼4', 90),

('李明', '110101199001011234', '13812345678', 1, '2023-10-10 10:00:00', '宿舍楼1', 60),

('王芳', '120202198902021234', '13987654321', 2, '2023-10-11 14:00:00', '宿舍楼2', 120),

('张强', '130303197003031234', '13765432109', 3, '2023-10-12 09:00:00', '宿舍楼3', 30),

('赵云', '140404196004041234', '13876543210', 4, '2023-10-13 15:00:00', '宿舍楼4', 90),

('李明', '110101199001011234', '13812345678', 1, '2023-10-10 10:00:00', '宿舍楼1', 60),

('王芳', '120202198902021234', '13987654321', 2, '2023-10-11 14:00:00', '宿舍楼2', 120),

('张强', '130303197003031234', '13765432109', 3, '2023-10-12 09:00:00', '宿舍楼3', 30),

('赵云', '140404196004041234', '13876543210', 4, '2023-10-13 15:00:00', '宿舍楼4', 90),

('钱坤', '150505199505051234', '13654321098', 5, '2023-10-14 13:00:00', '宿舍楼5', 45);

INSERT INTO users (username, password\_hash, role, permissions)

VALUES

('admin', 'HASH12345', '管理员', '全权限'),

('dorm\_manager1', 'HASH67890', '宿管人员', '管理宿舍1'),

('dorm\_manager2', 'HASH11111', '宿管人员', '管理宿舍2'),

('student1', 'HASH22222', '学生', '查看个人信息'),

('student2', 'HASH33333', '学生', '查看个人信息');

INSERT INTO logs (user\_id, action, action\_time, result)

VALUES

(1, '创建学生数据', '2023-10-15 10:00:00', '成功'),

(2, '修改宿舍分配', '2023-10-16 11:00:00', '成功'),

(3, '记录访客信息', '2023-10-17 12:00:00', '成功'),

(4, '查看费用记录', '2023-10-18 13:00:00', '成功'),

(5, '维修申报', '2023-10-19 14:00:00', '失败');

**3、数据操作与查询**

**1. 简单查询**

**查询学生表中的所有数据：**

SELECT \* FROM students;

**查询学生表中专业为“计算机科学”的学生：**

SELECT student\_id, name, gender, major, class

FROM students

WHERE major = '计算机科学';

**查询宿舍当前入住人数小于最大容量的宿舍（空宿舍）：**

SELECT dorm\_id, floor, room\_type, capacity, current\_occupants

FROM dormitories

WHERE current\_occupants < capacity;

**查询设施设备状态为“损坏”的所有设备：**

SELECT facility\_id, name, brand, model, status

FROM facilities

WHERE status = '损坏';

**2. 连接查询**

**查询学生及其对应的宿舍信息（内连接）：**

SELECT s.student\_id, s.name, s.gender, s.major, d.dorm\_id, d.room\_type, d.floor

FROM students s

INNER JOIN dormitories d ON s.dorm\_id = d.dorm\_id;

**查询宿舍及其设备信息（内连接）：**

SELECT d.dorm\_id, d.room\_type, d.floor, f.name AS facility\_name, f.status

FROM dormitories d

INNER JOIN facilities f ON d.dorm\_id = f.dorm\_id;

**查询所有学生及其费用缴费记录（左连接）：**

SELECT s.student\_id, s.name, fr.payment\_date, fr.amount, fr.payment\_method

FROM students s

LEFT JOIN fee\_records fr ON s.student\_id = fr.student\_id;

**查询访客和被访问学生的信息（连接查询）：**

SELECT v.visitor\_id, v.name AS visitor\_name, v.contact AS visitor\_contact,

s.name AS student\_name, s.dorm\_id

FROM visitors v

INNER JOIN students s ON v.visit\_target = s.student\_id;

**3. 嵌套查询**

**1.查询学生表中入住人数最多的宿舍中的学生信息**

-- 查询入住人数最多的宿舍的学生信息

SELECT student\_id, name, dorm\_id

FROM students

WHERE dorm\_id = (

SELECT dorm\_id

FROM dormitories

ORDER BY current\_occupants DESC

LIMIT 1

);

**注释：**

1. **外层查询**：
   * SELECT student\_id, name, dorm\_id: 查询学生表中的学号、姓名以及宿舍编号。
   * WHERE dorm\_id = (...)：限制只查询入住人数最多的宿舍中的学生。
2. **子查询**：
   * SELECT dorm\_id FROM dormitories ORDER BY current\_occupants DESC LIMIT 1：
     + 从 dormitories 表中查询入住人数（current\_occupants）最多的宿舍编号。
     + 使用 ORDER BY current\_occupants DESC 按照人数从多到少排序。
     + LIMIT 1 限制结果为单条记录，获取入住人数最多的宿舍编号。
3. **作用**：
   * 子查询先获取宿舍编号，作为外层查询的过滤条件，确保只查询与该宿舍相关的学生信息。

**预期结果**

查询出当前入住人数最多的宿舍中所有学生的学号、姓名及宿舍编号。

**2.查询费用记录中缴费金额高于平均缴费金额的学生信息**

-- 查询缴费金额高于平均缴费金额的学生信息

SELECT student\_id, payment\_date, amount

FROM fee\_records

WHERE amount > (

SELECT AVG(amount)

FROM fee\_records

);

**注释：**

1. **外层查询**：
   * SELECT student\_id, payment\_date, amount: 查询缴费金额高于平均值的学生 ID、缴费日期和缴费金额。
   * WHERE amount > (...)：限制只查询缴费金额大于平均缴费金额的记录。
2. **子查询**：
   * SELECT AVG(amount) FROM fee\_records：
     + 使用 AVG() 函数计算 fee\_records 表中所有缴费记录的平均缴费金额。
3. **作用**：
   * 子查询计算出平均缴费金额，外层查询用这个结果与每条缴费记录的金额进行比较，筛选出高于平均值的缴费记录。

**预期结果**

查询出缴费金额高于平均值的所有学生及相关缴费信息。

**4. 数据删除**

1、删除 facilities 表中状态为 '报废' 的所有记录。

DELETE FROM facilities

WHERE status = '报废';

2、级联删除

删除 dormitories 表中宿舍编号为 1 的宿舍，同时删除关联的 facilities 表记录（如果设置了外键约束（ON DELETE CASCADE），会级联删除关联表中的记录。）。

DELETE FROM dormitories

WHERE dorm\_id = 1;

**5. 数据更新**

1、将宿舍编号为 1 的宿舍楼层从 1 修改为 2。

UPDATE dormitories

SET floor = 2

WHERE dorm\_id = 1;

2、将 dormitories 表中所有当前入住人数为 0 的记录的房间类型改为 '多人间'。

UPDATE dormitories

SET room\_type = '多人间'

WHERE current\_occupants = 0;

**6、数据操作脚本**

以下脚本涵盖插入、更新和删除操作，适用于完整操作流程。

-- 插入操作

INSERT INTO dormitories (floor, room\_type, capacity, current\_occupants, furniture\_info, allocation\_history)

VALUES (2, '双人间', 2, 1, '床, 书桌, 衣柜', '张三');

-- 结果说明: 新增了一条宿舍记录，楼层为 2，房间类型为双人间。

INSERT INTO students (name, gender, major, class, contact, dorm\_id, check\_in\_date, encrypted\_contact)

VALUES ('张三', '男', '计算机科学', '2023级1班', '12345678901', 2, '2023-09-01', '加密数据1');

-- 结果说明: 新增了一条学生记录，姓名为张三。

-- 更新操作

UPDATE dormitories

SET room\_type = '单人间'

WHERE dorm\_id = 2;

-- 结果说明: 将宿舍编号为 2 的房间类型修改为单人间。

UPDATE students

SET dorm\_id = NULL

WHERE name = '张三';

-- 结果说明: 将姓名为张三的学生宿舍编号清空，表示退宿。

-- 删除操作

DELETE FROM students

WHERE student\_id = 1;

-- 结果说明: 删除了学号为 1 的学生记录。

DELETE FROM facilities

WHERE status = '报废';

-- 结果说明: 删除了状态为报废的所有设备记录。

**7、 ORDER BY 子句对查询结果排序**

-- 查询宿舍表中的所有记录，并按照楼层从低到高排序

SELECT dorm\_id, floor, room\_type, capacity, current\_occupants

FROM dormitories

ORDER BY floor ASC;

-- 查询目的: 获取所有宿舍记录，按照楼层升序排列，便于查看分布情况。

-- 预期结果: 所有宿舍按楼层从低到高依次排列。

**8、使用 DISTINCT 去除重复记录**

查询所有宿舍中存在的唯一房间类型

-- 查询宿舍表中所有不同的房间类型，去除重复记录

SELECT DISTINCT room\_type

FROM dormitories;

-- 查询目的: 获取宿舍表中存在的房间类型，避免重复显示。

-- 预期结果: 返回“单人间”、“双人间”、“多人间”这三种房间类型，每种仅显示一次。

**4、数据库完整性与安全性**

4.1 完整性约束验证

**1. 完整性约束验证**

完整性约束用于确保数据库中的数据符合规则和业务逻辑，主要包括以下四种类型：

**1.1 实体完整性**

* **定义**：实体完整性确保每个表的主键列具有唯一性，且不能为空。
* **实现方法**：
  + 在表的主键列上添加 PRIMARY KEY 约束。

CREATE TABLE students (

student\_id INT PRIMARY KEY AUTO\_INCREMENT COMMENT '学号',

name VARCHAR(50) NOT NULL COMMENT '姓名',

gender ENUM('男', '女') NOT NULL COMMENT '性别',

contact VARCHAR(15) NOT NULL COMMENT '联系方式'

);

**1.2 域完整性**

* **定义**：域完整性确保字段值必须满足定义的类型范围和约束条件。
* **实现方法**：
  + 使用字段的数据类型（如 VARCHAR, INT 等）和约束（如 NOT NULL, DEFAULT, ENUM）。

CREATE TABLE dormitories (

dorm\_id INT PRIMARY KEY AUTO\_INCREMENT COMMENT '宿舍编号',

floor INT NOT NULL COMMENT '楼层',

room\_type ENUM('单人间', '双人间', '多人间') NOT NULL COMMENT '房间类型'

);

**1.3 参照完整性**

* **定义**：参照完整性确保外键的值必须在主表中存在，或者为空。
* **实现方法**：
  + 使用 FOREIGN KEY 约束，设置外键关联。

CREATE TABLE students (

student\_id INT PRIMARY KEY AUTO\_INCREMENT COMMENT '学号',

dorm\_id INT COMMENT '宿舍编号',

FOREIGN KEY (dorm\_id) REFERENCES dormitories(dorm\_id)

);

**1.4 故意违反参照性约束**

1: 插入违反主键约束的记录

-- 向 students 表中插入两条记录，尝试使用相同的主键 student\_id

INSERT INTO students (student\_id, name, gender, major, class, contact, dorm\_id, check\_in\_date, encrypted\_contact)

VALUES (1, '张三', '男', '计算机科学', '2023级1班', '12345678901', 1, '2023-09-01', '加密数据1');

INSERT INTO students (student\_id, name, gender, major, class, contact, dorm\_id, check\_in\_date, encrypted\_contact)

VALUES (1, '李四', '女', '软件工程', '2023级2班', '12345678902', 2, '2023-09-02', '加密数据2');

**预期结果**

第二条记录的插入应失败，因为 student\_id 是主键，主键值必须唯一。

**实际结果**

数据库报错信息如下：

ERROR 1062 (23000): Duplicate entry '1' for key 'PRIMARY'

**分析**

* **原因**: student\_id 列被定义为主键（PRIMARY KEY），主键约束要求列值唯一且非空。第二次插入尝试使用了重复的主键值 1，违反了主键约束。
* **结论**: 数据库完整性约束生效，阻止了重复主键值的插入，数据一致性得到了保证。

2: 插入违反外键约束的记录

-- 向 students 表插入记录，其中 dorm\_id 指向 dormitories 表中不存在的宿舍编号

INSERT INTO students (name, gender, major, class, contact, dorm\_id, check\_in\_date, encrypted\_contact)

VALUES ('赵五', '男', '机械工程', '2023级3班', '12345678903', 999, '2023-09-03', '加密数据3');

**预期结果**

插入操作应失败，因为 dorm\_id 列是一个外键，且值 999 不存在于 dormitories 表的主键 dorm\_id 中。

**实际结果**

数据库报错信息如下：

ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails (`xsss`.`students`, CONSTRAINT `students\_ibfk\_1` FOREIGN KEY (`dorm\_id`) REFERENCES `dormitories` (`dorm\_id`))

**分析**

* **原因**: dorm\_id 被定义为外键，引用了 dormitories 表的主键列 dorm\_id。外键约束要求引用的值必须在父表（dormitories）中存在。值 999 不存在于 dormitories 表中，因此插入操作失败。
* **结论**: 数据库外键约束生效，阻止了无效外键值的插入，保证了数据的引用完整性。

**完整性约束验证报告**

| **操作编号** | **操作描述** | **预期结果** | **实际结果** | **分析与结论** |
| --- | --- | --- | --- | --- |
| 1 | 插入重复主键值 | 插入失败 | 报错：ERROR 1062 (23000): Duplicate entry '1' for key 'PRIMARY' | 主键约束生效，阻止了重复主键值的插入，保证了主键列的唯一性。 |
| 2 | 插入无效外键值 | 插入失败 | 报错：ERROR 1452 (23000): Cannot add or update a child row... FOREIGN KEY ... REFERENCES | 外键约束生效，阻止了无效外键值的插入，确保引用的外键值在父表中存在。 |

4.2 用户管理与权限设置

2.1 用户管理

1、创建用户：

使用 CREATE USER 创建数据库用户。

CREATE USER 'admin\_user'@'localhost' IDENTIFIED BY 'Admin@123';

CREATE USER 'dorm\_manager'@'localhost' IDENTIFIED BY 'Dorm@123';

CREATE USER 'student\_user'@'localhost' IDENTIFIED BY 'Student@123';

2.2 权限设置

分配权限：

根据用户角色分配权限，确保最小权限原则。

管理员权限（全面管理权限）：

GRANT ALL PRIVILEGES ON dorm\_management\_system.\* TO 'admin\_user'@'localhost';

宿管人员权限（管理宿舍和设备相关表）：

GRANT SELECT, INSERT, UPDATE, DELETE ON dorm\_management\_system.dormitories TO 'dorm\_manager'@'localhost';

GRANT SELECT, INSERT, UPDATE ON dorm\_management\_system.facilities TO 'dorm\_manager'@'localhost';

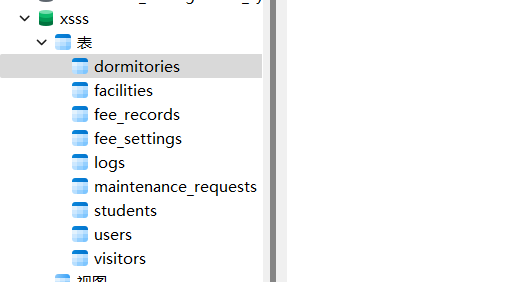
学生权限（只能查看和更新自己的数据）：

GRANT SELECT, UPDATE ON dorm\_management\_system.students TO 'student\_user'@'localhost';

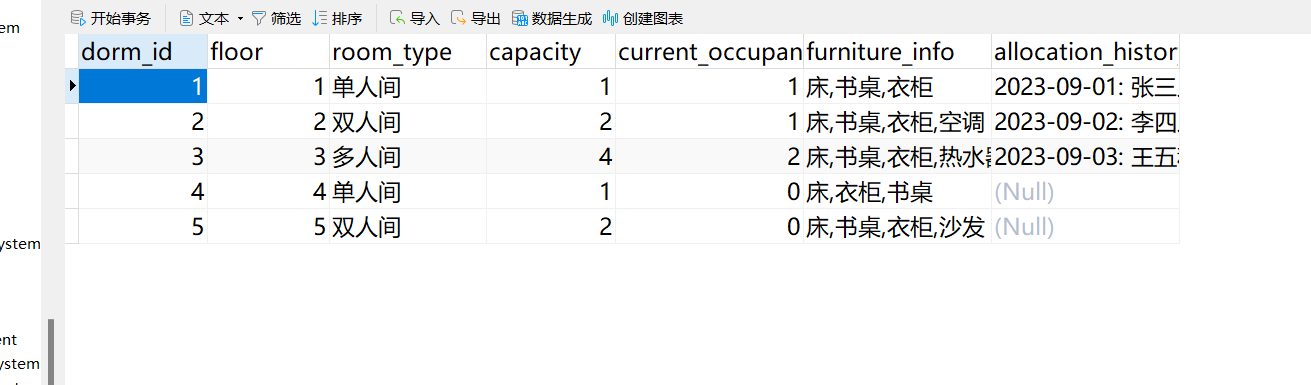
GRANT SELECT ON dorm\_management\_system.fee\_records TO 'student\_user'@'localhost';

**三、实验结果与分析**

1、建立表结构：



2、插入数据：



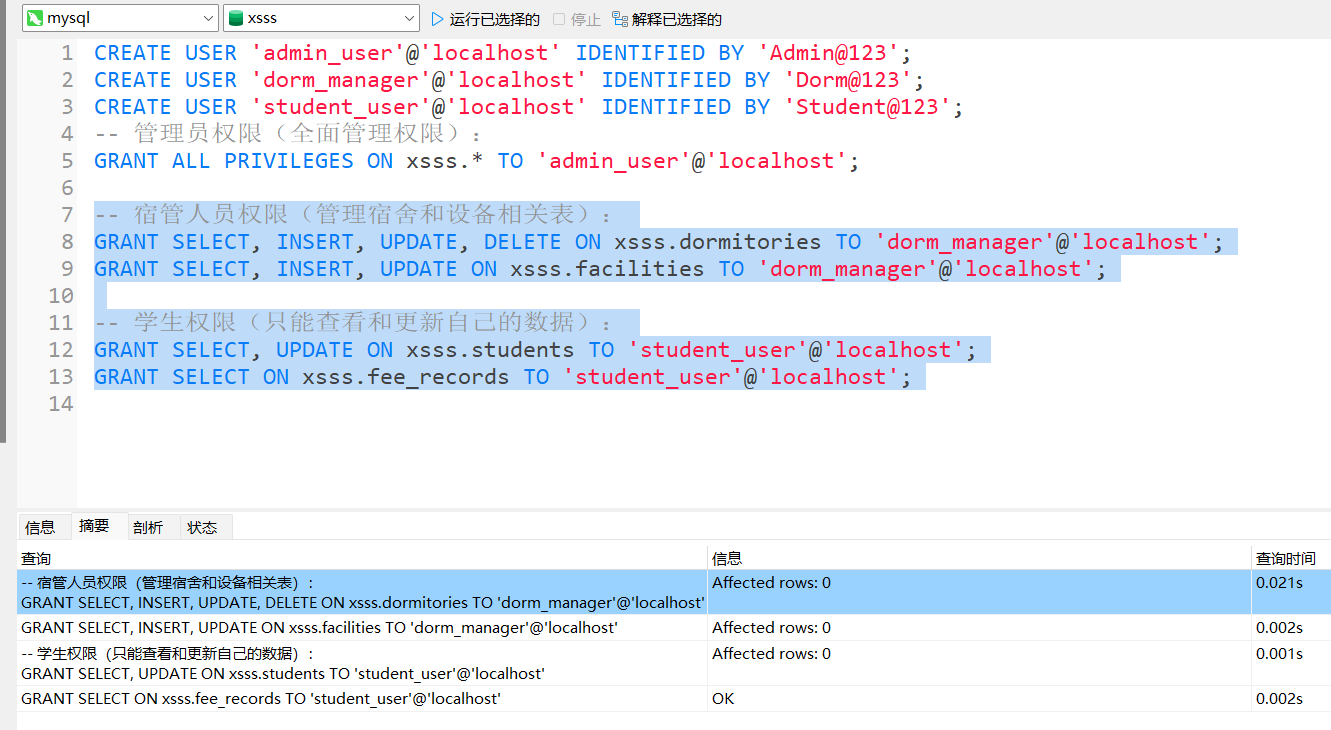
3、连接查询



4、嵌套查询



5、权限分配：



**四、实验总结与体会**

宿舍管理系统的设计与开发旨在实现宿舍管理信息化、提高效率，并减少人工管理中的错误和冗余。以下是本系统的主要功能模块与成果总结：

1. **功能模块概述**：
   * **学生信息管理**：实现学生基本信息的录入、查询、修改及宿舍入住/退房管理。
   * **宿舍信息管理**：维护宿舍基本信息（如楼层、房间类型等）、分配宿舍及空宿舍查询。
   * **设施设备管理**：记录设施设备信息，提供报修、维修和报废管理功能。
   * **费用管理**：设定住宿、水电、物业等费用标准，支持缴费记录管理和欠费提醒。
   * **访客管理**：实现访客登记、访问权限设置和访客记录查询功能。
   * **系统管理**：通过角色分配和权限管理确保数据安全，并记录用户操作日志。
   * **数据安全与完整性**：通过加密、外键约束、触发器等手段保证数据的安全性与一致性。
2. **数据库设计成果**：
   * 规范化的表结构设计，确保数据冗余最小化和存储效率最优。
   * 外键和约束设计有效保证了数据的完整性。
   * 用户权限设置和操作日志记录强化了系统的安全性与可审计性。
3. **技术实现**：
   * 采用 MySQL 作为数据库后端，确保系统性能和稳定性。
   * 使用 SQL 完成增删改查功能，并结合触发器和嵌套查询实现复杂的业务逻辑。
   * 数据加密技术保护敏感信息（如联系方式、身份证号等），实现了数据的隐私保护。