山西工程技术学院

**实验报告**

（ 2024 -2025 学年第2学期）

课程名称： 大型数据库系统开发

专业班级： 22计算机科学与技术一班

学 号： 2210708130

学生姓名： 郝泓毅

任课教师： 王晓霞

2025年 月

实验报告

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 实验名称 | 数据库、表的创建与管理 | | | 指导教师 | 王晓霞 |
| 实验类型 | 验证型 | 实验学时 | **4** | 实验时间 |  |
| 1. **实验目的与要求**   1.掌握创建、修改数据库的方法及管理数据库的方法  2.掌握创建、修改表结构的方法及插入、更新和删除表数据的方法。 | | | | | |
| **二、实验环境**  **MySQL 8.0** | | | | | |
| **三、实验内容和步骤**  (1)利用MySQL8.0创建一个名为teaching数据库,初始大小为10MB,增长速度为10%,其他均采用默认设置。  (2)在查询编辑器中输入创建表的代码，分别创建student、course、score、teacher、 class、teach\_class这6张表  (3)分别对这6张表输入记录（每张表不少于5条记录）  (4)向student表插入、删除、修改一条记录  (1)  CREATE DATABASE teaching  DEFAULT CHARACTER SET utf8mb4  DEFAULT COLLATE utf8mb4\_general\_ci  DATAFILE 'teaching\_data.MYDATA'  SIZE 10MB  AUTO\_INCREMENT 10%  LOGFILE 'teaching\_log.MYLOG'  SIZE 5MB;  (2)  CREATE TABLE student (  studentno CHAR(11) NOT NULL COMMENT '学号',  sname CHAR(8) NOT NULL COMMENT '姓名',  sex ENUM('男', '女') DEFAULT '男' COMMENT '性别',  birthdate DATE NOT NULL COMMENT '出生日期',  entrance INT(3) NULL COMMENT '入学成绩',  phone VARCHAR(12) NOT NULL COMMENT '电话',  Email VARCHAR(20) NOT NULL COMMENT '电子信箱',  PRIMARY KEY (studentno)  );  CREATE TABLE course (  courseno CHAR(6) NOT NULL,  cname CHAR(6) NOT NULL,  type CHAR(8) NOT NULL,  period INT(2) NOT NULL,  exp INT(2) NOT NULL,  term INT(2) NOT NULL,  PRIMARY KEY (courseno)  );  CREATE TABLE score (  studentno CHAR(11) NOT NULL,  courseno CHAR(6) NOT NULL,  daily FLOAT(4, 1) DEFAULT 0,  final FLOAT(4, 1) DEFAULT 0,  PRIMARY KEY (studentno, courseno),  FOREIGN KEY (studentno) REFERENCES student(studentno),  FOREIGN KEY (courseno) REFERENCES course(courseno)  );  CREATE TABLE teacher (  teacherno CHAR(6) NOT NULL COMMENT '教师编号',  tname CHAR(8) NOT NULL COMMENT '教师姓名',  major CHAR(10) NOT NULL COMMENT '专业',  prof CHAR(10) NOT NULL COMMENT '职称',  department CHAR(16) NOT NULL COMMENT '部门',  PRIMARY KEY (teacherno)  );  CREATE TABLE class (  classno CHAR(6) NOT NULL COMMENT '班级编号',  classname CHAR(10) NOT NULL COMMENT '班级名称',  monitor CHAR(8) NULL COMMENT '班长姓名',  PRIMARY KEY (classno)  );  CREATE TABLE teach\_class (  teacherno CHAR(6) NOT NULL,  classno CHAR(6) NOT NULL,  PRIMARY KEY (teacherno, classno),  FOREIGN KEY (teacherno) REFERENCES teacher(teacherno),  FOREIGN KEY (classno) REFERENCES class(classno)  );  (3)  INSERT INTO student (studentno, sname, sex, birthdate, entrance, phone, Email)  VALUES  ('2023010001', '张三', '男', '2005-03-10', 500, '13812345678', 'zhangsan@example.com'),  ('2023010002', '李四', '女', '2005-05-20', 520, '13987654321', 'lisi@example.com'),  ('2023010003', '王五', '男', '2005-08-15', 550, '13666666666', 'wangwu@example.com'),  ('2023010004', '赵六', '女', '2005-11-08', 580, '13777777777', 'zhaoliu@example.com'),  ('2023010005', '孙七', '男', '2006-02-28', 600, '13555555555', 'sunqi@example.com');  INSERT INTO course (courseno, cname, type, period, exp, term)  VALUES  ('c00001', '数学', '必修', 64, 16, 1),  ('c00002', '语文', '必修', 64, 16, 1),  ('c00003', '英语', '必修', 64, 16, 1),  ('c00004', '物理', '必修', 48, 16, 2),  ('c00005', '化学', '必修', 48, 16, 2);  INSERT INTO score (studentno, courseno, daily, final)  VALUES  ('2023010001', 'c00001', 85.5, 90.0),  ('2023010002', 'c00001', 90.0, 88.0),  ('2023010001', 'c00002', 80.0, 85.0),  ('2023010003', 'c00002', 75.5, 82.0),  ('2023010004', 'c00003', 95.0, 92.0);  INSERT INTO teacher (teacherno, tname, major, prof, department)  VALUES  ('t00001', '陈老师', '数学', '教授', '数学学院'),  ('t00002', '林老师', '语文', '副教授', '文学院'),  ('t00003', '黄老师', '英语', '讲师', '外语学院'),  ('t00004', '周老师', '物理', '副教授', '理学院'),  ('t00005', '吴老师', '化学', '讲师', '理学院');  INSERT INTO class (classno, classname, monitor)  VALUES  ('c10001', '一班', '张三'),  ('c10002', '二班', '李四'),  ('c10003', '三班', '王五'),  ('c10004', '四班', '赵六'),  ('c10005', '五班', '孙七');  INSERT INTO teach\_class (teacherno, classno)  VALUES  ('t00001', 'c10001'),  ('t00002', 'c10002'),  ('t00003', 'c10003'),  ('t00004', 'c10004'),  ('t00005', 'c10005');  (4)  INSERT INTO student (studentno, sname, sex, birthdate, entrance, phone, Email)  VALUES ('2023010006', '钱八', '男', '2006-06-06', 560, '13444444444', 'qianba@example.com');  DELETE FROM student WHERE studentno = '2023010006';  UPDATE student  SET entrance = 530  WHERE studentno = '2023010001'; | | | | | |
| 1. **实验小结和思考** | | | | | |
| 实验成绩 |  | 批阅日期 |  | 批阅人 |  |

实验报告

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 实验名称 | 数据完整性的设置 | | | 指导教师 | 王晓霞 |
| 实验类型 | 验证型 | 实验学时 | **4** | 实验时间 |  |
| 1. **实验目的与要求**   创建外键、UNIKUE约束  创建check约束、规则  掌握数据完整性的类型和实现机制  (2)掌握约束、规则对象的创建和修改 | | | | | |
| **二、实验环境**  **MySQL 8.0** | | | | | |
| **三、实验内容和步骤**  (1)利用MySQL将teaching数据库中score表的courseno列设置为引用表course的外键；  (2)在teaching数据库中class表的classname创建UNIQUE约束；  (3)为teaching数据库中student表的birthday列创建check约束,规定学生的年龄在17~25之间,为course表的credit列创建check约束,规定学分的取值范围为1~6,删除check约束；  (4)为teaching数据库创建规则prof\_rule,规定教师职称取值只能为”助教”、“讲师”、“副教授”、“教授”，并将其绑定到teacher表的Prof列，删除创建的规则。  (1)  ALTER TABLE score  ADD CONSTRAINT fk\_score\_courseno  FOREIGN KEY (courseno) REFERENCES course(courseno);  (2)  ALTER TABLE class  ADD CONSTRAINT unique\_classname  UNIQUE (classname);  (3)  ALTER TABLE student  ADD CONSTRAINT check\_student\_age  CHECK (YEAR(CURRENT\_DATE) - YEAR(birthdate) BETWEEN 17 AND 25);    ALTER TABLE course  ADD CONSTRAINT check\_course\_credit  CHECK (credit BETWEEN 1 AND 6);  ALTER TABLE student  DROP CONSTRAINT check\_student\_age;  ALTER TABLE course  DROP CONSTRAINT check\_course\_credit;  (4)  ALTER TABLE teacher  MODIFY COLUMN prof ENUM('助教', '讲师', '副教授', '教授'); | | | | | |
| 1. **实验小结和思考** | | | | | |
| 实验成绩 |  | 批阅日期 |  | 批阅人 |  |

实验报告

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 实验名称 | 单表数据检索 | | | 指导教师 | 王晓霞 |
| 实验类型 | 验证型 | 实验学时 | **4** | 实验时间 |  |
| 1. **实验目的与要求**   (1)会使用SELECT语句对单表进行查询,会使用LIKE进行模糊查询,会用WHERE加学号、班级进行查询  (2)会用聚合函数COUNT进行统计、MAX求最大值、MIN求最小值、TOP5求成绩前5名学生信息，会用ORDER进行排序，GROUP BY按学号分组。  (2)掌握约束、规则对象的创建和修改  (1)掌握SELECT各个字句的功能和检索数据的方法  (2)掌握WHERE子句中LIKE、IN、BETWEEN、IS等逻辑运算符的使用  (3)掌握聚合函数的使用 | | | | | |
| **二、实验环境**  **MySQL 8.0** | | | | | |
| **三、实验内容和步骤**  (1)查询所有课程的课程编号、课程名和学分，查询160501班所有学生的基本信息  (2)查询student表中所有年龄大于20岁的男生的姓名和年龄  (3)查询计算机学院教师的专业名称  (4)查询每名学生的学号、选修课程数目、总成绩，并将查询结果存放到生成的“学生选课统计表”  (5)查询student表中所有学生的基本信息,查询结果按班级号classno升序排序,同一班级中的学生按入学成绩point降序排列  (6)查询各班学生的人数(按班级分组),查询各班期末成绩的最高分和最低分  (7)查询教授两门及以上课程的教师编号、课程编号和任课班级  (8)查询课程编号以c05开头,被三名及以上学生选修,且期末成绩的平均分高于75分的课程号、选修人数和期末成绩平均分，并按平均分降序排序。  (1)  SELECT courseno, cname, credit FROM course;  SELECT \*FROM student WHERE classno = '160501';  (2)  SELECT sname, YEAR(CURRENT\_DATE) - YEAR(birthdate) AS age  FROM student  WHERE sex = '男' AND YEAR(CURRENT\_DATE) - YEAR(birthdate) > 20;  (3)  SELECT major  FROM teacher  WHERE department = '计算机学院';  (4)  CREATE TABLE 学生选课统计表 (      studentno CHAR(11),      course\_count INT,      total\_score DECIMAL(10, 2)  );  INSERT INTO 学生选课统计表 (studentno, course\_count, total\_score)  SELECT      s.studentno,      COUNT(sc.courseno) AS course\_count,      SUM(sc.daily + sc.final) AS total\_score  FROM      student s  LEFT JOIN      score sc ON s.studentno = sc.studentno  GROUP BY      s.studentno;  (5)  SELECT \* FROM student ORDER BY classno ASC, entrance DESC;  (6)  SELECT classno, COUNT(studentno) AS student\_count FROM student GROUP BY classno;  (7)  SELECT      t.teacherno,      c.courseno,      tc.classno  FROM      teacher t  JOIN      teach\_class tc ON t.teacherno = tc.teacherno  JOIN      score sc ON tc.classno = sc.studentno      course c ON sc.courseno = c.courseno  GROUP BY      t.teacherno, c.courseno, tc.classno  HAVING      COUNT(c.courseno) >= 2;  (8)  SELECT      sc.courseno,      COUNT(DISTINCT sc.studentno) AS student\_count,      AVG(sc.final) AS average\_final\_score  FROM      score sc  WHERE      sc.courseno LIKE 'c05%'  GROUP BY      sc.courseno  HAVING      COUNT(DISTINCT sc.studentno) >= 3 AND AVG(sc.final) > 75  ORDER BY      average\_final\_score DESC; | | | | | |
| 1. **实验小结和思考** | | | | | |
| 实验成绩 |  | 批阅日期 |  | 批阅人 |  |

实验报告

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 实验名称 | 多表查询与子查询 | | | 指导教师 | 王晓霞 |
| 实验类型 | 综合设计型 | 实验学时 | **4** | 实验时间 |  |
| 1. **实验目的与要求**   (1)会使用JOIN ON连接两个及以上表  (2)会在SELECT查询语句中再嵌套一个SELECT语句  (3)会使用游标对查询结果集进行处理(查询结果集必须是大于等于两条记录)  (1)掌握多表连接查询及多表连接的各种方法,包括内连接、外连接  (2)掌握子查询的方法，包括相关子查询和不相关子查询  (3)掌握游标处理结果集的基本过程 | | | | | |
| **二、实验环境**  **MySQL 8.0** | | | | | |
| **三、实验内容和步骤**  (1)查询所有班级的期末成绩平均分,并按照平均分降序排序  (2)查询教师基本信息和教授课程信息,其中包括未分配课程的教师信息  (3)查询两门及以上课程的期末成绩超过80分的学生姓名及其平均成绩  (4)查询没有被任何学生选修的课程编号、课程名称和学分（子查询）  (5)查询入学成绩最高的学生学号、姓名和入学成绩（子查询）  (6)查询同时教授c05127号和c05109号课程的教师信息(子查询)  (7)查询每门课程的课程号、课程名和选修该课程的学生人数，并按所选人数升序排序  (8)使用游标输出学生姓名、选修课程名称和期末考试成绩  (1)  SELECT class.classno, AVG(score.final) AS avg\_score  FROM class           JOIN teach\_class ON class.classno = teach\_class.classno           JOIN score ON teach\_class.courseno = score.courseno  GROUP BY class.classno  ORDER BY avg\_score DESC;  (2)  SELECT teacher.\*, course.courseno, course.cname  FROM teacher           LEFT JOIN teach\_class ON teacher.teacherno = teach\_class.teacherno           LEFT JOIN course ON teach\_class.courseno = course.courseno;  (3)  SELECT student.sname, AVG(score.final) AS avg\_score  FROM student           JOIN score ON student.studentno = score.studentno  WHERE score.final > 80  GROUP BY student.sname  HAVING COUNT(score.courseno) >= 2;  (4)  SELECT courseno, cname, credit  FROM course  WHERE courseno NOT IN (SELECT DISTINCT courseno FROM score); | | | | | |
| 1. **实验小结和思考** | | | | | |
| 实验成绩 |  | 批阅日期 |  | 批阅人 |  |

实验报告

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 实验名称 | 单表数据检索 | | | 指导教师 | 王晓霞 |
| 实验类型 | 验证型 | 实验学时 | **4** | 实验时间 |  |
| 1. **实验目的与要求**   (1)会使用SELECT语句对单表进行查询,会使用LIKE进行模糊查询,会用WHERE加学号、班级进行查询  (2)会用聚合函数COUNT进行统计、MAX求最大值、MIN求最小值、TOP5求成绩前5名学生信息，会用ORDER进行排序，GROUP BY按学号分组。  (2)掌握约束、规则对象的创建和修改  (1)掌握SELECT各个字句的功能和检索数据的方法  (2)掌握WHERE子句中LIKE、IN、BETWEEN、IS等逻辑运算符的使用  (3)掌握聚合函数的使用 | | | | | |
| **二、实验环境**  **MySQL 8.0** | | | | | |
| **三、实验内容和步骤**  (1)查询所有课程的课程编号、课程名和学分，查询160501班所有学生的基本信息  (2)查询student表中所有年龄大于20岁的男生的姓名和年龄  (3)查询计算机学院教师的专业名称  (4)查询每名学生的学号、选修课程数目、总成绩，并将查询结果存放到生成的“学生选课统计表”  (5)查询student表中所有学生的基本信息,查询结果按班级号classno升序排序,同一班级中的学生按入学成绩point降序排列  (6)查询各班学生的人数(按班级分组),查询各班期末成绩的最高分和最低分  (7)查询教授两门及以上课程的教师编号、课程编号和任课班级  (8)查询课程编号以c05开头,被三名及以上学生选修,且期末成绩的平均分高于75分的课程号、选修人数和期末成绩平均分，并按平均分降序排序。  (1) | | | | | |
| 1. **实验小结和思考** | | | | | |
| 实验成绩 |  | 批阅日期 |  | 批阅人 |  |