计算机科学技术学院实验报告

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **课程名称** | 数据库原理与应用 | | | **学 号** | 230511637 |
| **实验项目** | SQL SERVER2012代码编辑器的使用 | | | **姓 名** | 张世浩 |
| **学 时** | 1.5h | **项目性质** | 设计型 | **班 级** | 2305116 |
| **指导教师** | 李誌,颜雁 | **实验地点** | 实训楼430 | **日 期** | 2025年10月22日 |
| 1. **实验目的和要求**   一 实验题目： SQL SERVER2012代码编辑器的使用  二 实验目的：熟悉SQL SERVER2012 环境，熟悉代码编辑器的使用，能够熟练运用sql命令完成数据库，基本表、主码、外码和其它必要的约束条件的定义，此外利用ALTER语句完成数据库和基表的修改。（请依据实际题目完成 | | | | | |
| 1. **实验环境**   Sql sever  Navicat Premium Lite 17 | | | | | |
| 1. **实验内容与过程**   流程图:    实验代码:  -- 1. 重建数据库（确保环境干净）  DROP DATABASE IF EXISTS StudentInfoDB;  CREATE DATABASE StudentInfoDB;  USE StudentInfoDB;  -- 2. 创建表（含约束）  CREATE TABLE 学生 (  学号 CHAR(10) PRIMARY KEY,  年龄 INT NOT NULL CHECK (年龄 BETWEEN 16 AND 30),  性别 CHAR(2) NOT NULL CHECK (性别 IN ('男', '女')),  系号 CHAR(4) NOT NULL  ) ;  CREATE TABLE 课程 (  课号 CHAR(6) PRIMARY KEY,  课名 VARCHAR(50) NOT NULL,  学分 INT NOT NULL CHECK (学分 BETWEEN 1 AND 6),  学时 INT NOT NULL CHECK (学时 >= 16)  ) ;  CREATE TABLE 选课 (  学号 CHAR(10) NOT NULL,  课号 CHAR(6) NOT NULL,  成绩 DECIMAL(5,1) CHECK (成绩 BETWEEN 0 AND 100),  PRIMARY KEY (学号, 课号),  FOREIGN KEY (学号) REFERENCES 学生(学号) ON DELETE CASCADE,  FOREIGN KEY (课号) REFERENCES 课程(课号) ON DELETE CASCADE  ) ;  -- 3. 插入合法记录  INSERT INTO 学生 VALUES  ('S2024001', 20, '男', 'CS01'),  ('S2024002', 19, '女', 'MA01'),  ('S2024003', 22, '男', 'PH02');  INSERT INTO 课程 VALUES  ('C00001', '数据库系统', 4, 64),  ('C00002', '线性代数', 3, 48),  ('C00003', '大学物理', 5, 80);  INSERT INTO 选课 VALUES  ('S2024001', 'C00001', 88.5),  ('S2024001', 'C00002', 92.0),  ('S2024002', 'C00002', 85.0);  -- 4. 测试违反约束的记录（执行会报错）  /\*  INSERT INTO 学生 VALUES ('S2024001', 21, '女', 'CH03'); -- 主键重复  INSERT INTO 学生 VALUES ('S2024004', 15, '男', 'MA01'); -- 年龄超限  INSERT INTO 课程 VALUES ('C00004', '专业英语', 7, 32); -- 学分超限  INSERT INTO 选课 VALUES ('S2024999', 'C00001', 70.0); -- 外码不存在  INSERT INTO 选课 VALUES ('S2024003', 'C00003', 105.0); -- 成绩超限  \*/  -- 5. 修改日志文件配置（通过配置文件）  SHOW VARIABLES LIKE 'innodb\_log\_file\_size'; -- 查看当前日志大小  -- 6. 添加并删除约束（兼容低版本，不查询约束）  -- 添加系号格式约束  ALTER TABLE 学生  ADD CONSTRAINT CK\_系号格式 CHECK (系号 REGEXP '^[A-Za-z]{2}[0-9]{2}$');  -- （低版本兼容方案）通过查看表结构确认约束是否添加  SHOW CREATE TABLE 学生; -- 从结果中可找到添加的CHECK约束  -- 删除约束  ALTER TABLE 学生  DROP CONSTRAINT CK\_系号格式; | | | | | |
| 1. **实验结果与分析**   实验结果截图:    实验结果分析:  **一**、数据库与表创建阶段   1. 数据库创建通过 DROP DATABASE IF EXISTS 和 CREATE DATABASE StudentInfoDB 语句，成功创建了全新的数据库，执行 USE StudentInfoDB 可正常切换，无旧数据干扰，数据库创建流程正常。 2. 表结构与约束定义学生表、课程表、选课表均成功创建，所有约束生效：    * 主键约束：学生.学号、课程.课号 及选课表的复合主键 (学号, 课号) 均正确定义，确保记录唯一性。    * 外键约束：选课表的 学号 关联学生表、课号 关联课程表，并设置 ON DELETE CASCADE（主表记录删除时级联删除子表关联记录），保证数据关联性。    * CHECK 约束：年龄（16-30）、学分（1-6）、成绩（0-100）等范围约束均生效，可通过 sp\_help 表名 查看表结构及约束详情。   二、合法数据插入阶段   1. 数据插入结果向三张表插入的合法记录均成功执行（提示 “(3 行受影响)”），无报错。例如：    * 学生表的 3 条记录学号唯一，年龄、性别均符合约束；    * 选课表的学号和课号均对应学生表、课程表中已存在的记录，外键关联有效。 2. 数据验证执行 SELECT \* FROM 学生、SELECT \* FROM 课程、SELECT \* FROM 选课 可查询到所有插入数据，字段值均在约束范围内，数据完整性得到保障。   三、约束冲突测试阶段  测试 5 种违反约束的场景，SQL Server 均返回明确报错，验证了约束有效性：   1. 学生表学号重复：违反主键约束，报错 “违反了 PRIMARY KEY 约束... 不能在对象中插入重复键”，拒绝重复录入。 2. 学生年龄 = 15：违反 CHECK 约束，报错 “CHECK 约束失败... 该值必须在 16 到 30 之间”，阻止非法年龄录入。 3. 课程学分 = 7：违反 CHECK 约束，报错 “CHECK 约束失败... 学分必须在 1 到 6 之间”，限制学分范围。 4. 选课学号不存在：违反外键约束，报错 “INSERT 语句与 FOREIGN KEY 约束冲突... 找不到匹配的记录”，确保关联数据存在。 5. 成绩 = 105：违反 CHECK 约束，报错 “CHECK 约束失败... 成绩必须在 0 到 100 之间”，拦截超范围成绩。   四、日志文件与约束管理阶段   1. 日志文件配置执行 SELECT name, size FROM sys.database\_files WHERE type\_desc = 'LOG' 可查看当前日志文件信息（size 单位为 8KB 页）。通过 ALTER DATABASE StudentInfoDB MODIFY FILE (NAME = 'StudentInfoDB\_log', SIZE = 100MB) 成功修改日志文件初始大小，SQL Server 即时生效（无需重启服务）。 2. 约束动态管理    * 添加约束：执行 ALTER TABLE 学生 ADD CONSTRAINT CK\_系号格式 CHECK (系号 LIKE '[A-Za-z][A-Za-z][0-9][0-9]') 成功为 “系号” 添加格式约束（2 字母 + 2 数字），通过 sp\_helpconstraint 学生 可查看约束已生效。    * 删除约束：执行 ALTER TABLE 学生 DROP CONSTRAINT CK\_系号格式 成功移除约束，再次查询约束列表时该约束已消失，支持动态调整。   五、实验结论   1. SQL Server 的主键、外键、CHECK 约束能有效保障数据完整性，阻止重复、非法及关联不一致的数据录入。 2. 日志文件大小可通过 ALTER DATABASE 语句直接修改，操作便捷，适合根据业务需求调整存储配置。 3. 约束支持动态添加和删除，灵活性高，可适配业务规则的变化。 4. 所有 SQL 操作在 SQL Server 环境中均按预期执行，无语法错误，验证了代码的正确性和约束机制的有效性**。** | | | | | |
| 1. **实验心得**   本次数据库实验让我从理论走向实践，完整体验了从数据库搭建到约束管理的全流程。过程中不仅解决了多个语法和环境适配问题，更对数据库设计的核心逻辑有了深刻体会，以下是具体的实操心得：  一、数据库与表创建  在实验第一步，我先执行了DROP DATABASE IF EXISTS StudentInfoDB，再创建新数据库并切换使用。一开始没加IF EXISTS，遇到了 “数据库已存在” 的报错，后来才明白这行语句的作用 —— 清除旧环境残留，避免后续表创建时出现冲突，这让我意识到数据库操作必须考虑 “兼容性”，不能默认环境是全新的。  创建表时，我重点关注了约束定义：给学生表的 “学号”、课程表的 “课号” 设为主键，选课表用 “学号 + 课号” 作为复合主键，还给选课表添加了关联学生表和课程表的外键，并设置ON DELETE CASCADE。一开始我忘了给外键加级联删除，后来测试删除主表记录时，子表关联数据删不掉才发现问题。这让我明白，主键是确保记录唯一的 “身份证”，外键是维持表间关联的 “纽带”，而约束不是随便加的，要根据实际业务需求设计，否则会影响数据操作的合理性。  给字段加CHECK约束时，比如年龄限制在 16-30、学分 1-6，我一开始觉得这些限制 “多余”，但后来测试时才发现，正是这些约束能从源头挡住非法数据，这也让我理解了 “数据完整性” 的重要性 —— 数据库不仅要存数据，还要保证数据是可靠、有效的。  二、数据插入：  插入合法数据时，我采用了批量插入的方式，把多条记录写在一个INSERT语句里，执行效率更高。但一开始插入选课数据时，不小心用了不存在的学号，直接触发了外键约束报错，这才意识到 “选课表的学号和课号必须在学生表、课程表中已存在” 这个关联规则。后来调整了插入顺序，先插学生表和课程表，再插选课表，数据就顺利插入了。  通过查询语句验证插入结果时，我逐一核对了每条记录的字段值，确保没有违反CHECK约束，比如年龄都在 16-30 之间，成绩在 0-100 之间。这一步让我养成了 “操作后验证” 的习惯，数据库操作不能想当然，必须通过查询确认结果正确，否则后续实验会基于错误数据展开，导致更多问题。  三、约束冲突测试：  这是我觉得最有收获的一步。我故意插入违反约束的记录，比如重复的学号、年龄 15、学分 7 等，观察系统的报错信息。当看到 “主键约束冲突”“CHECK 约束失败”“外键引用失败” 这些明确的提示时，我对每种约束的作用有了直观认识 —— 原来主键是防止重复，CHECK是限制范围，外键是保证关联。  之前学理论时，总觉得约束是抽象的概念，通过这次主动试错，我清楚地知道了 “如果不遵守规则，数据库会如何反应”。比如测试成绩 105 时，系统直接拒绝插入，这让我明白约束不是 “摆设”，而是实实在在保护数据质量的 “防线”。如果没有这些约束，数据库里可能会出现年龄为负数、选不存在课程的无效数据，后续做统计分析时就会出错。  四、日志配置与约束管理：灵活适配需求  查看日志文件大小时，我通过SHOW VARIABLES语句了解了当前的配置，也知道了 MySQL 和 SQL Server 修改日志大小的不同方式 ——MySQL 需要改配置文件，SQL Server 可以用ALTER DATABASE语句直接修改。这让我意识到，数据库不是 “一成不变” 的，要根据实际需求调整配置，比如数据量大会需要更大的日志文件，保证数据恢复时不丢失信息。  动态添加和删除约束时，我先给 “系号” 加了 “2 字母 + 2 数字” 的格式约束，通过SHOW CREATE TABLE验证约束生效后，又用DROP CONSTRAINT删除了该约束。一开始在低版本 MySQL 中，因为不支持DROP CONSTRAINT IF EXISTS，还遇到了语法报错，后来通过调整语句解决了问题。这让我明白，不同数据库版本的语法支持有差异，实际开发中要考虑兼容性；同时约束是可以灵活调整的，当业务需求变化时，比如 “系号格式修改”，不需要重建表，只需动态修改约束，就能适配新规则。  **总结**：  这次实验让我深刻体会到 “数据库设计的核心是规范”。从数据库创建时的环境清理，到表结构设计时的约束定义，再到数据操作时的顺序和验证，每一步都需要遵循规范。以前觉得枯燥的 SQL 语句，在实践中变得生动起来，每一个语法的背后都有其用途，比如IF EXISTS是为了兼容性，ON DELETE CASCADE是为了关联数据的一致性。  同时，实验中遇到的报错也让我学会了 “排查问题”—— 先看报错信息，定位问题类型（是语法错误还是约束冲突），再针对性解决。这不仅提高了我的 SQL 操作能力，还培养了我的逻辑思维和问题解决能力。 | | | | | |
| 1. **教师评语** | | | | | |
| 1. **实验成绩**   教师签名： 李誌,颜雁 批阅日期： 2025年10月22日 | | | | | |

注：项目性质为 演示型、验证型、设计型、综合型和创新型。