



数据库实验报告

实验题目  **数据库及基本表的建立**

学 院 计算机学院

专 业 计算机科学与技术3班

年 级 2020

姓 名 张俊鸿

指导教师 谢锐

（2022年 09 月）

实验一 数据库及基本表的建立

**实验项目名称:数据库及基本表的建立**

**实验项目性质:验证型**

**所属课程名称:《数据库应用》**

**实验计划学时:3**

# **实验目的**

1. 掌握SQL SERVER的查询分析器和企业管理器的使用;
2. 掌握创建数据库和表的操作;

# **实验内容和要求**

1. 分别使用SQL语句、企业管理器(EnterpriseManager)创建数据库;
2. 使用SQL语句、企业管理器(Enterprise Manager)创建数据库表;

# **实验主要仪器设备和材料**

1. 计算机及操作系统:PC机，Windows 11;
2. 数据库管理系统:MySQL

# **实验方法、步骤及结果测试**

创建一个教学管理数据库SC，其描述的信息有:学生信息、课程信息、教师信息、学生选课成绩、授课信息、班级信息、系部信息、专业信息。

创建:student表、course表、teacher表、 student\_course表、teacher\_course表等。

1. **创建数据库:**

* 确定数据库名称；

数据库用于学生管理，命名为homework。

·确定数据库的位置；

要求:数据文件和日志文件分别存储在自己的目录下。

(1)、使用SQL 语句指定参数创建数据库:

实现代码及截图：

|  |  |
| --- | --- |
| SQL语句 | 执行结果截图显示 |
| -- 创建数据库  CREATE DATABASE SC;  USE SC; |  |

实验结论：通过CREATE DATABASE语句，成功创建了SC数据库，实验结果验证了CREATE DATABASE语句的创建功能的正确性。

1. **创建基本表**

利用查询分析器，使用 SQL 语句方式创建方式将下面各表建立到教学管理数据库中。

1. **创建student表**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **代码** | **类型** | **约束** |
| **学号** | **s\_no** | **char(8)** | **主键** |
| **姓名** | **sname** | **char(8)** | **非空** |
| **性别** | **sex** | **char(2)** |  |
| **出生日期** | **sbirthday** | **Smalldatetime** |  |
| **学生所在院系编号** | **dno** | **char(6)** | **外键** |
| **专业代码** | **spno** | **char(8)** | **外键** |
| **班级编码** | **class\_no** | **char(4)** |  |

1. **创建Course表**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **代码** | **类型** | **约束** |
| **课程编号** | **cno** | **char(10)** | **主键** |
| **课程名称** | **cname** | **char(20)** | **非空** |
| **专业代码** | **spno** | **char(8)** | **外键** |
| **课程类型编号** | **ctno** | **tinyint** |  |
| **理论学时** | **lecture** | **tinyint** |  |
| **实验学时** | **experiment** | **tinyint** |  |
| **开课学期** | **semester** | **tinyint** |  |
| **课程学分** | **credit** | **tinyint** |  |

1. **创建student \_course表**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **代码** | **类型** | **约束** |
| **学号** | **s\_no** | **char(8)** | **主键,与student表中s\_no 外键关联,级联删除** |
| **上课编号** | **tcid** | **smallint** | **主键** |
| **学生成绩** | **score** | **tinyint** |  |

1. **创建teacher表**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **代码** | **类型** | **约束** |
| **教师编号** | **t\_no** | **char(8)** | **主键** |
| **教师姓名** | **t\_name** | **char(8)** | **非空** |
| **性别** | **t\_sex** | **char(2)** |  |
| **出生日期** | **t\_birthday** | **smalldatetime** |  |
| **教师所在院系编号** | **dno** | **char(6)** | **外键** |
| **职称** | **tech\_title** | **char(10)** |  |

1. **创建系部表(department)**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **代码** | **类型** | **约束** |
| **院系编号** | **dno** | **char(6)** | **主键** |
| **院系名称** | **dept\_name** | **char(20)** | **非空** |
| **院系负责人** | **header** | **char(8)** |  |

1. **创建专业信息表(speciality)**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **代码** | **类型** | **约束** |
| **专业代码** | **spno** | **char(8)** | **主键** |
| **院系编号** | **dno** | **char(6)** | **外键,非空** |
| **专业名称** | **spname** | **char(20)** | **非空** |

1. **创建teacher\_course表**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **代码** | **类型** | **约束** |
| **上课编号** | **tcid** | **smallint** | **主键** |
| **教师编号** | **t\_no** | **char(8)** | **外键** |
| **专业代码** | **spno** | **char(8)** | **外键** |
| **班级编码** | **class\_no** | **char(4)** |  |
| **课程编号** | **cno** | **char(10)** | **非空,外键** |
| **学期** | **semester** | **Char(6)** |  |
| **学年** | **schoolyear** | **Char(10)** |  |

1. **创建班级表(class)**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **代码** | **类型** | **约束** |
| **专业代码** | **spno** | **char(8)** | **主键，与 speciality 表中spno 外键关联,** |
| **班级编码** | **class\_no** | **char(4)** | **主键,** |
| **班负责人** | **header** | **char(8)** |  |

1. **查看各数据表之间的关系，生成数据库关系图。**

|  |
| --- |
| **生成数据库关系图截图显示** |
| **实验结论：通过Navicat的逆向数据库到模型，生成数据库关系截图，可以显示分析不同表之间的外键关联。** |

1. **查看各数据表之间的关系，生成数据库关系图。**

|  |  |
| --- | --- |
| **SQL语句** | **执行结果截图显示** |
| -- 创建表  CREATE TABLE student (  s\_no char(10) PRIMARY KEY,-- 学号  sname char(8) NOT NULL,-- 姓名  sex char(2),-- 性别  sbirthday datetime,-- 出生日期  dno char(6),-- 学生所在院系编号  spno char(8),-- 专业代码  class\_no char(4)-- 班级编号  );  CREATE TABLE Course (  cno char(10) PRIMARY KEY,-- 课程编号  cname char(20) NOT NULL,-- 课程名称  spno char(8),-- 专业代码  ctno tinyint,-- 课程类型编号  lecture tinyint,-- 理论学时  experiment tinyint,-- 理论学时  semester tinyint,-- 理论学时  credit tinyint-- 理论学时  );  CREATE TABLE student\_course (  s\_no char(10),-- 学号  tcid smallint,-- 上课编号  score tinyint,-- 学生成绩  PRIMARY KEY (`s\_no`,`tcid`)  );  CREATE TABLE teacher (  t\_no char(8) PRIMARY KEY,-- 教室编号  t\_name char(8) NOT NULL,-- 教师姓名  t\_sex char(2),-- 性别  t\_birthday datetime,-- 出生日期  dno char(6),-- 教师所在院系编号  tech\_title char(10)-- 职称  );  CREATE TABLE department (  dno char(6) PRIMARY KEY,-- 院系编号  dept\_name char(20) NOT NULL,-- 院系名称  header char(8)-- 院系负责人  );  CREATE TABLE speciality (  spno char(8) PRIMARY KEY,-- 专业代码  dno char(6) NOT NULL,-- 院系编号  spname char(20) NOT NULL-- 专业名称  );  CREATE TABLE teacher\_course (  tcid smallint PRIMARY KEY,-- 上课编号  t\_no char(8),-- 教师编号  spno char(8),-- 专业代码  class\_no char(4),-- 班级编码  cno char(10) NOT NULL,-- 课程编号  semester char(6),-- 学期  schoolyear char(10)-- 学年  );  CREATE TABLE class (  spno char(8), -- 专业代码  class\_no char(4), -- 班级编码  header char(8) ,-- 班负责人  PRIMARY KEY (`spno`,`class\_no`)  );  -- 添加外键  ALTER TABLE class  ADD CONSTRAINT `class\_ibk\_1` FOREIGN KEY (`spno`) REFERENCES `speciality` (`spno`);  ALTER TABLE course  ADD CONSTRAINT `Course\_ibk\_1` FOREIGN KEY (`spno`) REFERENCES `speciality` (`spno`);    ALTER TABLE speciality  ADD CONSTRAINT `speciality\_ibk\_1` FOREIGN KEY (`dno`) REFERENCES `department` (`dno`);    ALTER TABLE student  ADD CONSTRAINT `student\_ibk\_1` FOREIGN KEY (`dno`) REFERENCES `department` (`dno`),  ADD CONSTRAINT `student\_ibk\_2` FOREIGN KEY (`spno`) REFERENCES `speciality` (`spno`);    ALTER TABLE student\_course  ADD CONSTRAINT `student \_course\_ibk\_1`  FOREIGN KEY (`s\_no`)  REFERENCES `student` (`s\_no`)  ON DELETE CASCADE;  ALTER TABLE teacher  ADD CONSTRAINT `teacher\_ibk\_1` FOREIGN KEY (`dno`) REFERENCES `department` (`dno`);    ALTER TABLE teacher\_course  ADD CONSTRAINT `teacher\_course\_ibk\_1` FOREIGN KEY (`t\_no`) REFERENCES `teacher` (`t\_no`),  ADD CONSTRAINT `teacher\_course\_ibk\_2` FOREIGN KEY (`spno`) REFERENCES `speciality` (`spno`),  ADD CONSTRAINT `teacher\_course\_ibk\_3` FOREIGN KEY (`cno`) REFERENCES `course` (`cno`); |  |

实验结论：通过CREATE TABLE语句，创建出所需要8个表，明确了不同数据类型的定义，同时还以ALTER TABLE语句在创建表之后给不同的表加上外键。

1. **利用查询分析器修改上述各表。**
2. **用INSERT语句向各个表中插入数据录入5条记录。录入时注意体会外键约束。**

**实现代码及截图：**

|  |  |
| --- | --- |
| SQL语句 | 实验结果截图显示 |
| -- 插入数据  INSERT into sc.department VALUES  ('00001','计算机学院','张三'),  ('00002','自动化学院','李四'),  ('00003','外国语学院','王五'),  ('00004','物理学院','赵四'),  ('00005','土木学院','张五'),  ('00006','光电学院','小刘')  INSERT into sc.teacher VALUES  ('00000001','张老师','男','2002-10-13','00001','教授'),  ('00000002','俊老师','女','2001-10-13','00001','副教授'),  ('00000003','鸿老师','男','2003-10-13','00001','讲师'),  ('00000004','章老师','女','2004-10-13','00001','讲师'),  ('00000005','骏老师','男','2005-10-13','00001','教授'),  ('00000006','红老师','女','2006-10-13','00001','副教授')  INSERT into sc.speciality VALUES  ('11111111','00001','计科'),  ('11111112','00001','网工'),  ('11111113','00001','信安'),  ('11111114','00001','软工'),  ('11111115','00001','人工智能')  INSERT into sc.course VALUES  ('00001','毛概','11111111',1,48,0,1,6),  ('00002','软件工程','11111111',2,40,0,1,3),  ('00003','数据库系统','11111111',3,48,24,1,3),  ('00004','信息安全概论','11111111',4,32,0,1,3),  ('00005','工程管理','11111111',5,24,0,1,1)  INSERT into sc.student VALUES  ('3120005043','张俊鸿','男','2002-10-13','00001','11111111','3'),  ('2120005043','张俊白','女','2003-10-13','00001','11111111','3'),  ('4120005043','张俊红','男','2004-10-13','00001','11111111','3'),  ('5120005043','张俊绿','男','2005-10-13','00001','11111111','3'),  ('6120005043','张俊黑','女','2006-10-13','00001','11111111','3')  INSERT into sc.`student\_course` VALUES  ('3120005043',1,98),  ('3120005043',2,99),  ('3120005043',3,97),  ('3120005043',4,96),  ('3120005043',5,95)    INSERT into sc.class VALUES  ('11111111','3','张俊鸿'),  ('11111112','4','张俊白'),  ('11111113','6','张俊红'),  ('11111114','7','张俊绿'),  ('11111115','8','张俊黑')  INSERT into sc.teacher\_course VALUES  (1,'00000001','11111111','3','00001','001','0001'),  (2,'00000002','11111111','3','00002','002','0002'),  (3,'00000001','11111111','3','00003','003','0003'),  (4,'00000004','11111111','3','00004','004','0004'),  (5,'00000001','11111111','3','00005','005','0005') |  |

实验结论：通过insert插入语句，在每个表中插入5条数据，插入数据要明确先后顺序，先插入无外键的，再插入需要外键关联的数据。实验结果验证了INSERT – VALUES语句的插入功能的正确性。

1. **用UPDATE语句更改student表中数据；**

**实现代码：**

|  |
| --- |
| -- 修改数据  UPDATE sc.student SET sname='张俊粉' WHERE s\_no = '6120005043'; |

|  |  |
| --- | --- |
| **student表更改前的内容截图显示查询分析** | **student表更改后的内容截图显示** |
|  |  |

实验结论：通过update更新语句，可以更新STUDENT表中某一条记录，实验结果验证了UPDATE SET语句的更新功能的正确性。

1. **用DELETE语句删除student表中数据；**

|  |
| --- |
| -- 删除数据  DELETE FROM sc.student WHERE s\_no = '6120005043'; |

|  |  |
| --- | --- |
| **student表更改前的内容截图显示查询分析** | **student表更改后的内容截图显示** |
|  |  |

实验结论：通过delete删除语句，可以删除STUDENT表中某一条记录，实验结果验证了DELETE FROM语句的更新功能的正确性。

# **实验中出现的问题及解决方案**

1. **问题：**创建表无法按照所给的表顺序创建，因为存在外键约束，不能将属性关联到未创建的表上。

**解决方案：**先不进行外键约束，将8个表依次创建完毕之后，使用ALTER语句进行外键关联约束。

1. 问题：所给的表设计存在缺陷，student\_course上课编号tcid没有与Course表的课程编号cno进行外键关联。

**解决方案**：修改student\_course的tcid为cno，类型为char(10)，并进行外键约束。

# **思考题**

1. **说明数据库中的表和数据文件的关系。**

表是一个用来管理数据存储的逻辑概念，表只是和数据文件发生关系，数据文件是物理的，一个表可以包含多个数据文件，而一个数据文件只能隶属一个表空间。

1. **数据库中的日志文件能否单独修改?**

不能，因为数据库中的日志文件记录了数据库的日志信息，单独修改日志文件可能会造成数据库以及日志文件两者的不一致性