

计算机系《数据库原理》实验报告-实验五

姓名：马星 学号：5418122020 专业班级：计算机科学与技术(卓越计划) 221 班

一、实验名称：

数据库完整性设置

二、实验目的：

掌握实体完整性的定义和维护方法

三、实验要求：

1、实验要求：理解数据库的实体和参照完整性的概念，掌握设置主键和外键的完整性约束的 SQL 语句，理解在数据库操作中，如果违反了完整性约束的设置数据库系统会如何处理；

2、实验内容

本实验为应用综合实验,要求同学能够在一定的应用场景下设计 SQL 实现数据库的主键和外键完整性约束条件,并查看数据库对违反操作的处理。

1) 完成 P173: 练习 6 中数据库表格设计及对应要求的完整性约束语句

练习 6. 假设有下面两个关系模式:

职工(职工号, 姓名, 年龄, 职务, 工资, 部门号), 其中职工号为主码:

部门(部门号, 名称, 经理名, 电话), 其中部门号为主码。

用 SQL 语言定义这两个关系模式, 要求在模式中完成以下完整性约束条件的定义:

(1) 定义每个模式的主码: (2) 定义参照完整性: (3) 定义职工年龄不得超过 60 岁。

2) 完成 p173: 练习 7 中违反操作的反应 (需要分别对练习 6 中 3 中约束执行一次错误的操作, 并将操作反应截图到实验报告中并添加说明)

练习 7. 在关系系统中, 当操作违反实体完整性、参照完整性和用户定义的完整性约束条件时, 一般是如何分别进行处理的?

四、实验过程和结果：

练习六:

首先使用 CREATE SCHEMA 创建一个 SCHEMA, 然后使用 CREATE TABLE 建表

定义主码使用 PRIMARY KEY 关键字

创建表后, 再定义参照完整性, 用 ALTER TABLE 对表字段进行操作, ADD CONSTRAINT 语句对职工表添加约束, 职工表中部门号与部门表中的部门号对应, 所以对职工表的部门号字段设置为外键约束

最后定义职工年龄不得超过 60 岁, 使用 CHECK 指定约束条件

```
# 练习六
-- 定义职工和部门关系模式
CREATE SCHEMA 公司;
use 公司;

CREATE TABLE 职工
(
    职工号 INT PRIMARY KEY, # 定义主码
    姓名 VARCHAR(255),
    年龄 INT,
    职务 VARCHAR(255),
    工资 DECIMAL(10, 2),
    部门号 INT
);

CREATE TABLE 部门
(
    部门号 INT PRIMARY KEY, # 定义主码
    名称 VARCHAR(255),
    经理名 VARCHAR(255),
    电话 VARCHAR(255)
);

-- 定义参照完整性
ALTER TABLE 职工
    ADD CONSTRAINT 部门号 FOREIGN KEY (部门号) REFERENCES 部门 (部门号);

-- 定义年龄限制约束
ALTER TABLE 职工
    ADD CONSTRAINT 年龄限制 CHECK (年龄 <= 60);
```



练习七:

尝试插入一条违反完整性约束的记录, 观察数据库如何处理。

```
1 # 练习七
2 -- 插入一条主码缺失的职工记录
3 ! INSERT INTO 职工 (姓名, 年龄, 职务, 工资, 部门号)
4 VALUES ('张三', 30, '员工', 5000, 1);
5
6 VALUES ('张三', 30, '员工', 5000, 1)
7 [2024-05-18 15:24:41] [HY000][1364] Field '职工号' doesn't have a default value
```

在尝试插入一条主码缺失的记录时, 数据库报错, 主码没有默认值, 在缺失的情况下, 无法进行插入

```
-- 插入一条年龄超过60岁的职工记录
❗ INSERT INTO 职工 (职工号, 姓名, 年龄, 职务, 工资, 部门号)
VALUES (1, '张三', 65, '经理', 10000, 1);
```

```
公司> INSERT INTO 职工 (职工号, 姓名, 年龄, 职务, 工资, 部门号)
VALUES (1, '张三', 65, '经理', 10000, 1)
[2024-05-18 15:26:29] [HY000][3819] Check constraint '年龄限制' is violated.
```

在尝试插入年龄超出限制的记录时，数据库检查到，违反了“年龄限制”这条约束，导致插入失败

五、总结和感想:

本次实验的主要目的是掌握实体完整性的定义和维护方法，并深入理解数据库的实体完整性、参照完整性以及用户定义完整性的概念。通过实践，我们学习了如何在 SQL 中设置主键和外键的完整性约束，并观察了当操作违反这些完整性约束时，数据库系统如何处理这些情况。

在实验的第一部分，我们设计了两个关系模式：职工和部门，并为这两个模式定义了主键和参照完整性约束。具体来说，我们为职工表定义了主键为职工号，并设置了部门号作为外键，引用部门表的主键部门号。此外，我们还添加了一个用户定义的完整性约束，即职工年龄不得超过 60 岁。

在实验的第二部分，我们尝试执行了一些违反上述完整性约束的操作。例如，尝试插入一条年龄超过 60 岁的职工记录，以观察数据库系统如何响应。结果表明，当操作违反了实体完整性或参照完整性约束时，数据库管理系统会拒绝执行该操作，或者在某些情况下，可能会采取级联操作来维持数据的一致性。

通过本次实验，我们不仅学习了如何使用 SQL 语言实现不同类型的完整性约束，还了解了数据库管理系统在数据操作过程中如何检查和维护这些约束。这些知识和技能对于维护数据库的可靠性和正确性至关重要。

总的来说，这次实验使我们对数据库完整性有了更深入的理解，并能够在实际的数据库设计和管理中应用所学知识，确保数据的准确性和一致性。