第5章 数据库完整性

概述

1、定义(integrity)

——数据的正确性，有效性和相容性。

（防止不符合语义的数据的I/O）

2、功能

1）完整性需求的定义；

2）监督事务执行，测试是否违反完整性限制条件；

3）若发生违反情况，则进行相应处理（拒绝、报告、纠正）。

3、 完整性约束条件

1、基本概念

① 定义

——施加于数据库中数据上的语义限制条件

② 约束对象

列级、元组级、关系级

③ 约束对象状态

静态：反映数据库状态合理性的约束。

动态：反映数据库状态变迁的约束。

④ 约束时机（Immediate constraints）

（1）立即约束

——一条语句执行完后立即检查。

（2）延迟约束（deferred constraints）

——事务执行结束后检查。

转帐，从A到B后，帐才能平，才能进行检查。

5.1 实体完整性（entity integrity）

——对关系模式主属性施加的完整性控制。

不允许空，在关系中取值唯一

例： student (SNO,SNAME,SEX,BYEAR)

SNO不能为空且唯一

course(CNO,CTITLE) CNO不能为空且唯一

sc(SNO,CNO,GRADE) (SNO，CNO)不能为空且唯一

create table student (SNO Char (6) NOT NULL,…,);

PRIMARY KEY? DBMS的一般做法是为主码建立B+树索引，插入时遍历索引直到叶子节点，判断待插入主码值不存在则可以执行插入动作。

5.2 参照完整性（referential integrity）

——对外码施加的完整性控制。

回忆第二章定义

参照关系：外码所在关系，如SC

被参照关系：主码（同时又是另一关系中的外码）所在关系，如student，course。

外码：sc中的SNO，CNO

1）空值情况

① 空值表示外码属性不存在或者无法确定；

② 非空则对应被参照关系中该元组存在；

③ 是否可为空，依据应用语义确定。

DEPT（DNO，DNAME，LOCATION），

EMPL（ENO，DNO，ENAME）

EMPL中DNO可为空，表示该职员还未分配到任何部门工作。

2）删除被参照关系元组情况

① 捆绑删除（cascade）

——参照与被参照关系中相关数据一起删除。

被参照关系中外码元组删除

例如：删除99001号学生（或者01号课程）

删去student中SNO='99001'的元组

（删去course中CNO='01'的元组）

则捆绑删除SC中学号为99001（课号为01）的所有元组

→参照关系中与被参照关系中码值对应元组删除

可能层层牵连

若SC又是另一参照关系的被参照关系，则可能又删除之……。

② 受限删除（restricted/No action）

——参照关系没有一个外码与要删除的被参照关系的主码值相对应时才执行删除。

例如：若SC中外码值SNO，无一个与主码值(Student.SNO)对应时才删去student、course中相应元组。

③ 置空值删除（nullifies/set null）

删去被参照关系中元组；

参照关系中所有与被参照关系中已删去的主码值相等的外码值置为空值。

如：删去部门表中的某个部门，则职员表中原来属于该部门的职员的所属部门号置空。

上述三种，选择哪一种实施，视应用需求确定。

如：学籍管理中，学生离校了，删去选课及学生信息，故需捆绑删除（course不删）student信息。

DBMS提供相应选择机制。

3）修改被参照关系主码值情况

① 捆绑修改（cascade）

修改被参照关系中主码值。

如修改student中的SNO=99003改为SNO=99020

同时修改参照关系中相等外码值。

则同时SC中所有99003改为99020

可能逐层牵连。

② 受限修改（restricted/No action）

——仅当参照关系中没有一个外码值与被参照关系中某个元组主码值相等时才可修改被参照关系中的该元组主码值。

如：仅当SC中学生无99003时，才可修改student中SNO=99003。

③ 置空值修改（nullifies）

修改被参照关系中的主码值；

将参照关系中的与该主码值相等的外码置为空值。

④ 说明

具体应用中，根据应用需求选择上述方法执行。

DBMS提供机制支持用户选择。

5.3 用户定义完整性（integrity of user definition）

1）空值控制

——对给定属性施加不允许空值限制（NOT NULL）

2）单个属性控制

为：（GRADE is Null） OR (GRADE BETWEEN 0 AND 100)

3）多属性控制

如：XB=‘男’ AND YL<=30

例1：CREATE TABLE DEPT
(DEPTNO INT PRIMARY KEY, //主码
DNAME CHAR(9) NOT NULL UNIQUE,
//不允许为空，且不能重复
LOCATION CHAR(10) NOT NULL);

例2：CREATE TABLE SC
(SNO INT,
CNO INT,
GRADE INT,
CHECK (GRADE>=0 AND GRADE<=100));

例3：CREATE TABLE STUDENT
(SNO INT PRIMARY KEY,
SNAME CHAR(30) NOT NULL,
SSEX CHAR(2),
SDEPT INT,
CHECK (SSEX='女' AND SNAME NOT LIKE 'Ms.%'));

例4：CREATE TABLE TEACHER
(ENO INT,
ENAME CHAR(30) NOT NULL,
SAL INT,
DEDUCT INT,
CONSTRAINT C1 CHECK (SAL+DEDUCT>=1000));

实发

应扣

例5：ALTER TABLE TEACHER
DROP CONSTRAINT C1;
ALTER TABLE TEACHER
ADD CONSTRAINT C2 CHECK
(SAL+DEDUCT>=2000 AND SAL>=1000);

5.4 完整性举例

1）实体完整性

```
CREATE TABLE student
(SNO NUMBER (8),
SNAME VARCHAR(8),
BYEAR NUMBER(3),
Constraint PK-XH PRIMARY KEY (SNO));
```

自动建立B+树索引

一旦定义了主码，则DBMS自动进行完整性检查：

主码不能为空；

主码值须唯一。

2）参照完整性

① 语句

FOREIGN KEY ：指定外码属性

REFERENCES：指定外码对应主码

ON DELETE CASCADE：指定捆绑删除要求

例：CREATE TABLE SC

(SNO NUMBER (8)，CNO NUMBER (3)，GRADE NUMBER (3)，

FOREIGN KEY (SNO) REFERENCES student (SNO)

ON UPDATE NO ACTION

ON DELETE NO ACTION,

FOREIGN KEY (CNO) REFERENCES course (CNO)

ON UPDATE NO ACTION

ON DELETE CASCADE)；

② 说明：

SC中外码有SNO和CNO。

SNO对应student主码为SNO。

CNO对应course主码为CNO。

当修改student中SNO时，先检查SC中有无元组SNO值与之相等，若有，则不能执行该修改；当修改course中CNO，先检查SC中有无元组的CNO值与之相等，若有，则不能执行该修改。

当删除student某元组时，则先在SC中找相应元组，若存在则拒绝执行删除。当删除course某元组时，则先在SC中找到相应元组，进行捆绑删除。

3）用户定义完整性

① 列值非空（NOT NULL）

② 列值唯一（UNIQUE）

```
CREAE TABLE COURSE
( CNO NUMBER (3),
CNM VARCHAR (20) CONSTRAINT U1 UNIQUE,
XS NUMBER (2));
```

其中：CONSTRAINT U1 UNIQUE：CNM唯一，约束名为U1。

③ 列值范围限制

例：．　．　．

```
    XB VARCHAR (2)
CONSTRAINT CNS_XB1 CHECK (XB IN('男', '女'));
```

数据库完整性机制在解决社会性突发问题起到的作用

- 近年来各类社会性问题频发，我们需要及时控制和妥善处理各类突发公共事件，提高快速反应和应急处理能力，建立健全应急机制，确保人们的生命安全。
- 以疫情期间在超市发现冻品表面新冠检测呈阳性为例

在设计超市数据库时，要考虑在若干关键表之间建立关联性

例：会员表

会员卡编号	姓名	身份证号	电话号码	会员等级	。。。。

商品

商品编号	商品名	单价	数量	。。。。

销售单据

会员卡编号	商品编号	。。。。		

数据库完整性机制在解决社会性突发问题起到的作用

例：会员表

会员卡编号	姓名	身份证号	电话号码	会员等级	。。。。

商品

商品编号	商品名	单价	数量	。。。。

销售单据

会员卡编号	商品编号	。。。。		

若某超市在售的某个冻品被发现表面新冠检测呈阳性。

↓

（1）立即下架该商品；

（2）通过销售单据表找到买此类商品的会员卡编号，排查与此关联的会员表，找到具体的购物人。

这些查找可通过定义上述三张表间的外码联系来实现。

疾控人员可及时地对经超市数据库排查出的密接人员进行核酸检测、隔离疑似人员等措施，尽可能减少对社会不良影响。

5.5 域中的完整性控制

可基于域定义关系的属性，可对域定义约束。

例1：域的使用

```
CREATE DOMAIN GenderDomain CHAR(2)
    CHECK (VALUE IN('男','女'));
```

或者

```
CREATE DOMAIN GenderDomain CHAR(2)
    CONSTRAINT C3 CHECK (VALUE IN('男','女'));
```

之后在某建表语句中：

……ssex GenderDomain,……

例2：

```
ALTER DOMAIN GenderDomain
    DROP CONSTRAINT C3;
```

5.6 断言（ASSERTION）

指定更具一般性的、较为复杂的约束：涉及多个表或聚集操作。

任何对断言中所涉及关系的操作都会触发对断言的检查，若断言为假则拒绝执行。

CREATE ASSERTION <断言名> <CHECK 子句>

例：限制数据库课程最多60名学生选修。

```
CREATE ASSERTION ASSE_SC_DB_NUM
    CHECK(60>=
        (SELECT COUNT(*)
         FROM COURSE,SC
         WHERE SC.SNO=COURSE.SNO
         AND COURSE.CNAME='数据库'
         )
    );
```

删除断言：DROP ASSERTION <断言名>;

例：限制每一门课程最多60名学生选修

```
CREATE ASSERTION ASSE_SC_DB_NUM
    CHECK(60>=ALL (SELECT COUNT(*)
          FROM SC
          GROUP BY CNO
          )
    );
```

每一门课程每学期最多60名学生选修

```
CREATE ASSERTION ASSE_SC_DB_NUM
    CHECK(60>=ALL (SELECT COUNT(*)
          FROM SC
          GROUP BY CNO,TERM)    );
```

5.7 触发器

触发器在SQL99之后才写入SQL标准，但很多关系数据库管理系统很早就支持触发器，因此不同的系统实现的触发器语法不尽相同，实际使用需参考系统的使用手册。

“事件-条件-动作”规则
Event-condition-action

例：
```
CREATE TRIGGER UPDATE-SC
BEFORE INSERT OR UPDATE ON SC
FOR EACH ROW
WHEN (:NEW.CNO='001')
AS BEGIN
IF  :NEW.GRADE>50
  THEN
        :NEW.GRADE  :=50;
ENDIF
END
```

说明：

- 利用触发器，当对SC中进入插入元组和修改GRADE值时，若为‘001’号课程，则GRADE>50时一律自动改为50分。
- 一般定义触发器语句为CREATE OR REPLACE TRIGGER。（先定义后使用）
- CREATE TRIGGER语句定义触发器的约束条件。

关于触发器的概念

1.触发器名

2.表名

3.触发事件

INSERT/DELETE/UPDATE(OF<触发列，…>)…

4.触发时机

BEFORE，AFTER

5.触发器类型

FOR EACH ROW，FOR EACH STATEMENT

6.触发条件

WHEN

7.触发动作体

AS BEGIN…END

触发器的作用？维护数据完整性　维护系统安全性

创建触发器

CREATE TRIGGER <触发器名>
{BEFORE|AFTER}<触发事件>ON<表名>
REFERENCING NEW|OLD ROW AS <变量>
FOR EACH {ROW|STATEMENT}
[WHEN<触发条件>] <触发动作体>

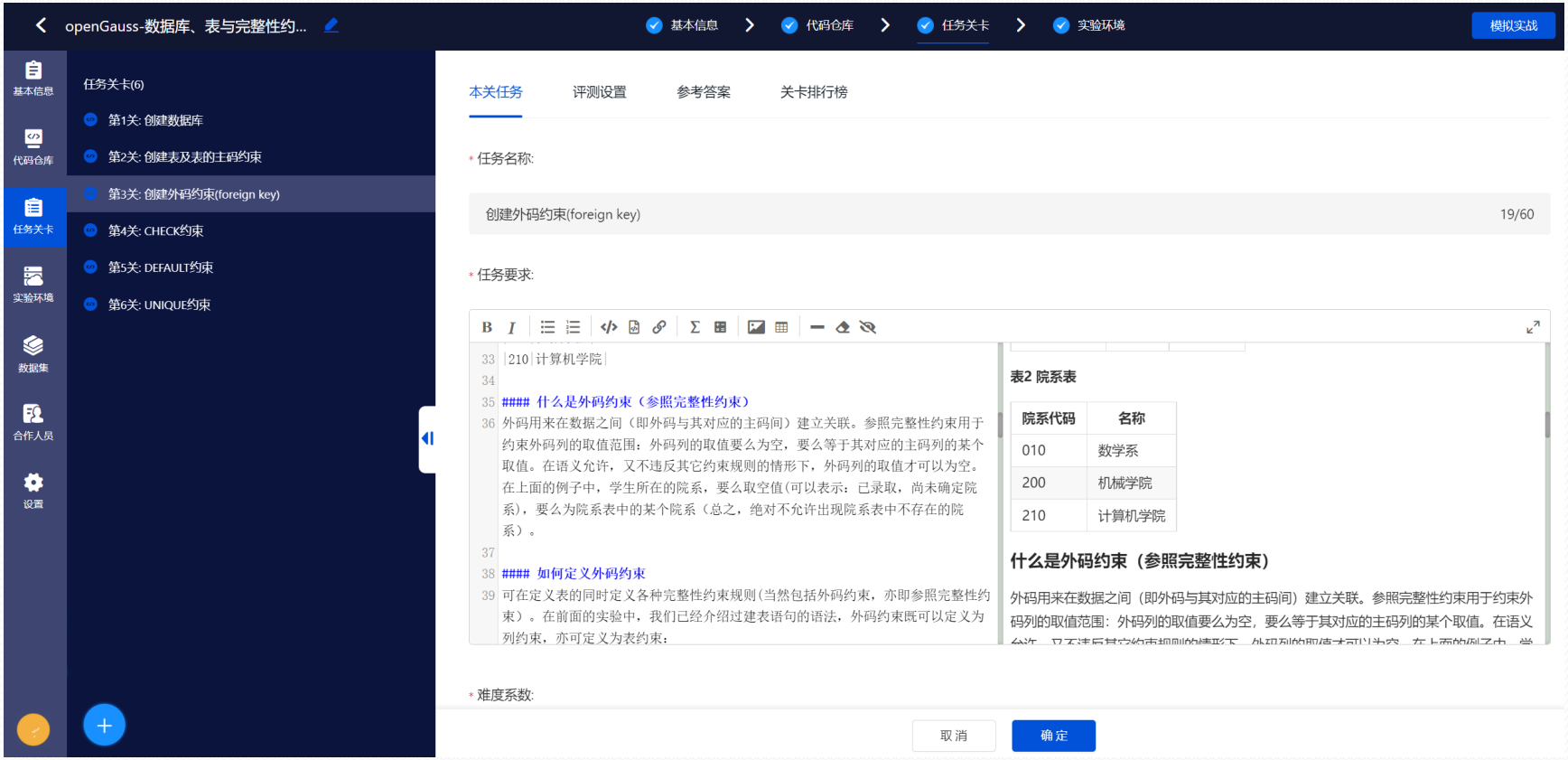
删除触发器

DROP TRIGGER 触发器名 ON 表名

触发器的激活

先执行BEFORE型触发器，再执行SQL语句，再执行AFTER型触发器。同类型触发器按创建时间先后顺序执行。

国产数据库完整性约束
——openGuass之头歌实验（拓展学习）



慕课讨论题

- 有哪些技术可以提升数据库完整性保护能力？

大家学过的哪些技术可以提高数据库系统的完整性保护能力？请举例并解释。