

数据库的完整性

数据的正确性：数据符合现实世界语义

数据的相容性：数据库同一对象在不同关系表中的数据是符合逻辑的。

防止正确和不语义

为维护数据库完整性，DBMS必须

1. 提供完整性约束条件的执行
2. 提供完整性检查的方法
3. 违约处理 拒绝

实体完整性定义

级联 实体完整性 → 主码 { 唯一 主键 - 属性不为空 }

create table 中用 primary key 定义

属性构成的码有两种说明方法 { 列级约束事件
表级约束事件

多个属性构成码只有一种说明方法 → 定义表级约束事件

列级主码定义

```
create table student (  
  Sno char(9) primary key,  
  Sname char(20) not null,  
  Ssex char(2),  
  Sage smallint,  
  Sdept char(20)  
);
```

表级定义主码

```
create table student (  
  Sno char(9)  
  Sname char(20) not null,  
  Sex smallint,  
  Sdept char(20)  
  Primary key (Sno)  
);
```

```
create table SC (  
  Sno char(9) not null  
  Cno char(9) not null  
  Grade smallint  
  Primary key (Sno, Cno)  
);
```

参照完整性 → foreign key references
外键 → 默认为空

create table SC (

Sno char(9) not null

Cno char(4) not null

Grade smallint,

Primary key(Sno, Cno),

foreign key(Sno) references Student(Sno),

foreign key(Cno) references Course(Cno).

指明这些外键参照哪些表的主键

在自己这不是主键在别人那是
对参照完整性,除了定义外键,还应定义
外码列是否允许空值

表级定义完整性

外键要么为空,要么是另张
表中存在该外键

);

参照完整性违约处理 { no action (拒绝) → 缺省
级联 cascade
设置为空值 (set-null)

create table SC (

~~Sno not null.~~

Sno char(9) not null,

Cno char(4) not null,

Grade smallint,

Primary key(Sno, Cno),

Foreign key(Sno) references Student(Sno), set

on delete cascade → 级联删除相应的元组

on update cascade ✓

Foreign key(Cno) references Course(Cno)

on delete no action → 拒绝

on update cascade

);

修改了被参照表的一个元组致不致,
则将参照表中所有造成不致的元组
的对应属性设为空值.

用户自定义完整性

{ 属性上的约束条件

- not null → 非空
- unique → 唯一, 可以为空
- check → 检查列值是否满足条件表达式

若在表级定义实体完整性, 隐含了 Cno, Cnc 的 not null 可不写

```
create table dept (
```

```
  Deptno numeric(2)
```

```
  Dname char(9) unique not null
```

```
  location CHAR(10),
```

```
  Primary Key (Deptno)
```

```
);
```

```
create table student (
```

```
  Sno char(9) Primary key,
```

```
  Sname char(8) not null,
```

```
  Ssex char(2) check (Ssex IN ('男', '女'))
```

```
  Sage smallint,
```

```
  Sdept char(20)
```

```
);
```

```
  check (Grade >= 0 and Grade <= 100)
```

表级上的约束条件

```
che create table student (
```

```
  Sno
```

```
  Primary key (Sno)
```

```
  check (Ssex = '女' or Sname not like 'Mr.%')
```


完整性约束语句

constraint <完整性约束名称> <完整性约束条件>

↓
Not null, unique, primary key,
foreign key, check 语句等

触发器 trigger

用户定义在表上的一类由事件驱动的特殊过程

触发器保存在数据库服务器中

任何用户对表的增、删、改操作均由服务器自动激活相应的触发器。

触发器可以实施更为复杂的检查和操作，且

定义触发器

1. create trigger <触发器名>

2. {before|after} <触发事件> on <表名>

事件-条件-动作

3. referencing new|old row as <变量>

4. for each {ROW|STATEMENT}

5. [when <触发事件>] (<触发动作体>)

{表的拥有者可以创建触发器

{触发器名可以包含模式名,也可不包含

{触发器和表名须在同一模式下

{触发器只能定义在基本表上,不能定义在视图上 → 可使用 old 和 new 引用新旧元组

for each row → 行级触发器

for each statement → 语句级触发器 → 不可用 old 和 new

触发事件:

insert, delete, update, update of <触发列>

触发动作体:

referencing new table as delta

for each statement

before insert or update on Teacher

for each row

触发器是由触发事件激活的

删除触发器 → 需已存在

drop trigger <触发器名> on <表名>;

↓ 有权限才可删除