

SQL完整性定义

约束类型

约束检查

其他约束形式

约束的对象级别

- 列级约束

- 列值范围

- 行级约束

- 同一行各列之间

- 表级约束

- 行间、表上、表间

分析下面约束的对象级别

- sno要求是8位整数，首位是0或1
- 飞行员的星级评定取决于其飞行里程
- 选课人数不能少于10人，多于100人
- 在本地纳税记录超过5年才有购房资格

约束类型

- **primary key**
- **foreign key**
- **unique**
- **default**
- **not null**
- **check**

这些约束的对象级别分别是什么？

存储约束的系统表：sysconstraints

列名	数据类型	描述
constid	int	约束号
id	int	拥有该约束的表 ID
colid	smallint	在其上定义约束的列 ID，如果是表约束则为 0
status	int	位图指示状态。可能的值包括： 1 = PRIMARY KEY 约束 2 = UNIQUE KEY 约束 3 = FOREIGN KEY 约束 4 = CHECK 约束 5 = DEFAULT 约束 16 = 列级约束 32 = 表级约束

约束命名及其定义

constraint 约束名 <约束条件>

sno char(8) **constraint** Sno_PK **primary key**

撤消约束: **alter table...drop constraint...**

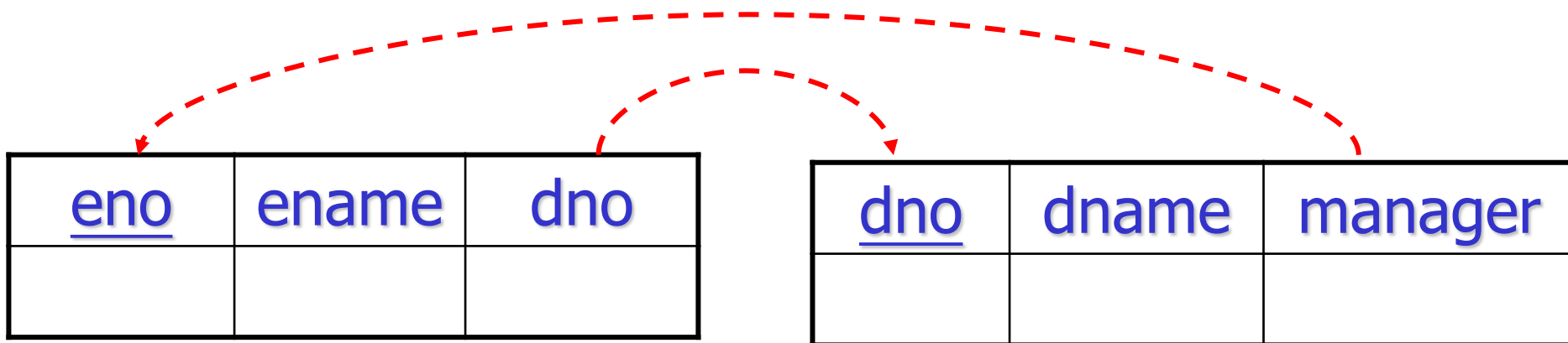
添加约束: **alter table...add constraint...**

alter table student **drop constraint** Sno_PK

alter table student **add constraint** Sex_CHECK **check**(sex in ('M', 'F'))

约束命名及其定义

如何定义两个相互参照的表？



- ① **create table** emp(eno primary key...)
- ② **create table** dept(dno primary key, manager foreign key...)
- ③ **alter table** emp **add constraint** dno foreign key...

primary key 与 unique

- 都是通过唯一性索引来支持，它们有何区别？
- 一个带有唯一性声明的属性（组）上有多少个null？

create table check_unique (

col1 int **unique**,

col2 int,

col3 int,

unique (col2, col3))

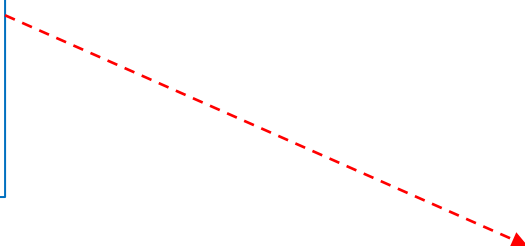
Col1	Col2	Col3
1	3	4
2	5	null
null	6	null

leader(teacher_no, project_no, start_time)

如何表达：每位老师在一定时间内只能负责一个项目？


primary key 与 unique 的背后

```
select  *  
  
from    INFORMATION_SCHEMA.table_constraints  
  
where   table_name = 'check_unique'
```



CONSTRAINT_NAME	TABLE_SCHEMA	TABLE_NAME	CONSTRAINT_TYPE
col1	my_db	check_unique	UNIQUE
col2	my_db	check_unique	UNIQUE

```
select  *  
  
from    INFORMATION_SCHEMA.innodb_indexes  
  
where   table_id=1225
```



INDEX_ID	NAME	TABLE_ID	TYPE
360	GEN_CLUST_INDEX	1225	1
361	col1	1225	2
362	col2	1225	2

foreign key

基本关系：主码所在的关系

依赖关系：外码所在的关系

- 为保证参照完整性，仅仅监视依赖关系上的修改操作是不完备的
- 如果基本关系中的行可以随意修改也会破坏参照完整性

sno	sname	dno
s1	Tom	d1
s2	Bob	d2

dno	dname
d1	CS
d2	MATH



sno	sname	dno
s1	Tom	d1
s2	Bob	d2

dno	dname
d2	MATH

foreign key: 三种定义方式

RESTRICT方式:

- 只有当依赖关系中没有一个外码值与要删除（更新）的基本关系的主码值相对应时，才可以删除（更新）该行（的主码），否则系统拒绝此删除操作

CASCADE方式:

- 将依赖关系中所有外码值与基本关系中要删除的主码值所对应的行一起删除（将依赖关系中所有与基本关系中要修改的主码值所对应的外码值一起修改为新值）

SET NULL方式:

- 删除（更新）基本关系中的行时，将依赖关系中与基本关系中被删（更新）主码值相对应的外码值置为空值

伐冰之家 不畜牛羊

delete d1

sno	sname	dno
s1	Tom	d1
s2	Bob	d2

dno	dname
d1	CS
d2	MATH

on delete RESTRICT

sno	sname	dno
s1	Tom	d1
s2	Bob	d2

dno	dname
d1	CS
d2	MATH

on delete CASCADE

sno	sname	dno
s2	Bob	d2

dno	dname
d2	MATH

on delete SET NULL

sno	sname	dno
s1	Tom	null
s2	Bob	d2

dno	dname
d2	MATH

表级约束：涉及多行或多表之间的影响

```
create table SC (  
    sno    char(8),  
    cno    char(10),  
    grade  smallint,  
    primay key (sno, cno),  
    check( sno in (select sno from student) ),  
    check( cno in (select cno from course) ) )
```

- 上述的check定义是否等价于外码约束？
- 如果student中删除行，不会触发check检查，只有SC表的更新才会触发

断言 **assertion** : **not exists**

create assertion AtoB **check**

(not exists

这个断言所
维护的约束
是什么?

(select A

from R

group by A

having count (distinct B) > 1)

)

很多数据库不支持断言，以及带子查询的check，怎么办？

函数约束：要求每门课的选课人数不能多于100人

```
create function fn_registerCount ( @cno varchar )  
returns int as  
begin  
return  
    (select count(*)  
    from    sc  
    where cno = @cno )  
End
```

为表sc列添加如下**check**约束：

```
alter table sc add constraint CK_registerCount check ( fn_registerCount(cno) <= 100 )
```

函数约束：要求优先使用最小的空缺序列号

noGap(no int **primary key**, name varchar(10))

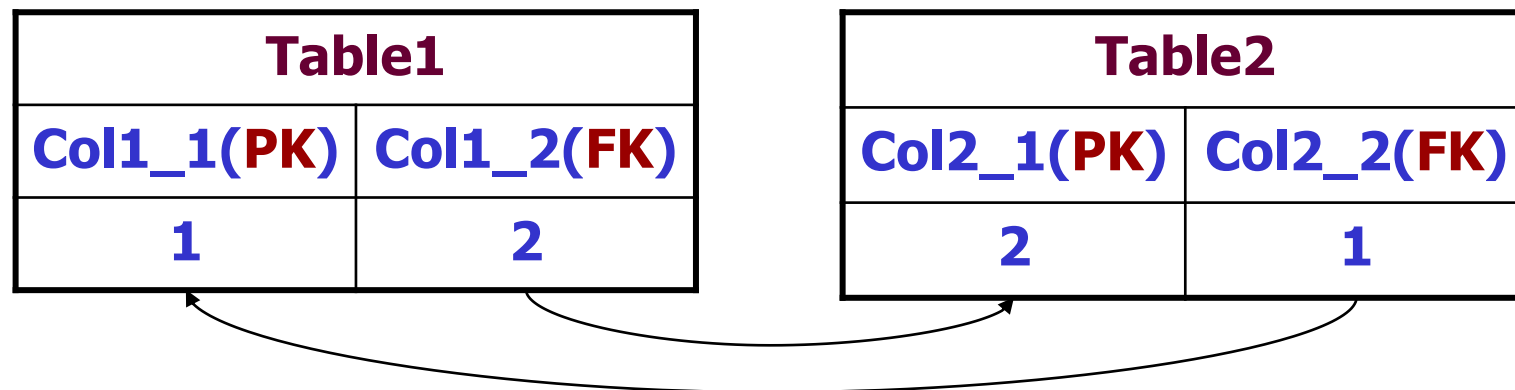
```
create function fn_noGap_minUnusedKey()
returns int as
begin
    if ( not exists ( select * from noGap where no = 1 )) return 1
    declare @min_unused_no int
    select    @min_unused_no = min(no + 1)
    from      noGap NO1
    where     not exists
                (select *
                 from   noGap NO2
                 where  NO2.no = NO1.no+1)
    return @min_unused_no
end
```

alter table noGap **add constraint** DF_no **default** (fn_noGap_minUnusedKey ()) **for** no

约束检查

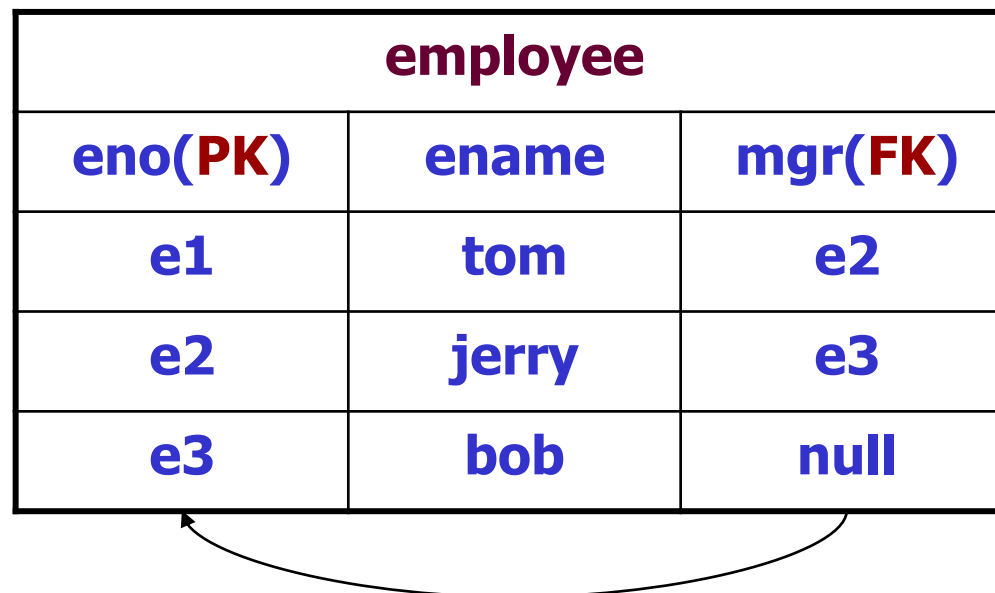
相互参照的表，
如何插入行？

互参照表



如何往自参照的表里
批量插入数据？

自参照表



标准SQL中的延迟约束

如何设置延迟约束？

延迟约束

deferred constraints

- 将多个更新操作语句放入一个事务，在提交时才检查约束

在约束创建时：

[not] deferrable

initially deferred [immediate]

对现有约束：

set constraint 约束名 deferred

延迟约束检查

```
create table emp(  
    eno      char(10) primary key,  
    ename    char(20),  
    mgr      char(10)  
    constraint FK_Constraint foreign key references emp(eno)  
    deferrable initially immediate
```

设置延迟约束: **set constraint FK_Constraint deferred**

SQL Server的约束开关

添加未验证的约束(with nocheck)

```
create table t1 (col_a INT)
```

```
insert into t1 values (-1)
```

```
alter table t1 with nocheck add constraint skip_check check (col_a > 1)
```

```
insert into t1 values (-2)
```

禁用并重新启用一个约束(nocheck)

```
alter table t1 nocheck constraint skip_check
```

```
insert into t1 values (-2)
```

```
alter table t1 check constraint skip_check
```

dbcc checkconstraints

Table	Constraint	Where
t2	skip_check	col_a = '-1'
t2	skip_check	col_a = '-2'

