

数据库系统实验 3 实验报告

数据科学与计算机学院 计算机科学与技术 2016 级

王凯祺 16337233

2018 年 10 月 22 日

1 实验 3.1 实体完整性实验

实体完整性，是指使用主键来唯一地标识一个实体。

在关系数据库中，一条记录代表一个实体。而实体是可以相互区分、识别的，也即它们应具有某种唯一性标识（该标识不能取相同的值，也不能为空）。

实体完整性强制表的标识符列或主键的完整性（通过索引、UNIQUE 约束、PRIMARY KEY 约束或 IDENTITY 属性）。

1.1 创建表时定义实体完整性（列级实体完整性）

定义供应商表的实体完整性。

```
1 create table supplier (  
2     suppkey int primary key,  
3     name char(25),  
4     address varchar(40),  
5     nationkey int,  
6     phone char(15),  
7     acctbal real,  
8     comment varchar(101)  
9 );
```

1.2 创建表时定义实体完整性（表级实体完整性）

定义供应商表的实体完整性。

```
1 create table supplier (  
2     suppkey int,  
3     name char(25),  
4     address varchar(40),  
5     nationkey int,  
6     phone char(15),  
7     acctbal real,  
8     comment varchar(101),  
9     primary key(suppkey)  
10 );
```

1.3 创建表后定义实体完整性

定义供应商表。

```
1 create table supplier (  
2     supkey int,  
3     name char(25),  
4     address varchar(40),  
5     nationkey int,  
6     phone char(15),  
7     acctbal real,  
8     comment varchar(101)  
9 );  
10  
11 alter table supplier add primary key (supkey);
```

1.4 定义实体完整性（主码由多个属性组成）

```
1 create table supplier (  
2     partkey int,  
3     supkey int,  
4     name char(25),  
5     address varchar(40),  
6     nationkey int,  
7     phone char(15),  
8     acctbal real,  
9     comment varchar(101),  
10    primary key (partkey, supkey)  
11    # 主码由多个属性组成，实体完整性必须定义在表级  
12 );
```

1.5 有多个候选码时定义实体完整性

定义国家表的实体完整性，其中 nationkey 和 name 都是候选码，选择 nationkey 作为主码，name 上定义唯一性约束。

```
1 create table nation (  
2     nationkey int primary key,  
3     name char(25) unique,  
4     regionkey int,  
5     comment varchar(152)  
6 );
```

1.6 删除实体完整性

指定一张表，让它删主键就好了（因为主键是唯一的）。

```
1 alter table nation drop primary key;
```

1.7 增加两条相同的记录，验证实体完整性是否起作用

插入两条主码相同的记录就会违反实体完整性约束。

```
1 insert into supplier values (11,'test1','test1',101,'1234',0.0,'test1');
```

1 row(s) affected

```
1 insert into supplier values (11,'test2','test2',102,'2345',0.1,'test2');
```

Error Code: 1062. Duplicate entry '11' for key 'PRIMARY'

1.8 实验总结

定义实体完整性实际上就是创建主键的过程。主键是唯一的，可以由一列组成，也可以由多列组合而成。当插入的行中主键与已有的行相同时，会拒绝其插入。主键还作为索引，提升查询的速度。

2 实验 3.2 参照完整性实验

参照完整性，就是定义在外键与主键之间的引用规则。当一个表需要用到别的表的主键时，可以使用外键进行引用。

2.1 创建表时定义参照完整性

先定义地区表的实体完整性，再定义国家表上的参照完整性。

```
1 create table region (  
2     regionkey int primary key,  
3     name char(25),  
4     comment varchar(152)  
5 );  
6  
7 create table nation (  
8     nationkey int primary key,  
9     name char(25),  
10    regionkey int,  
11    comment varchar(152),  
12    foreign key (regionkey) references region(regionkey)  
13 );
```

2.2 创建表后定义参照完整性

定义国家表的参照完整性。

```
1 create table nation (  
2     nationkey int primary key,  
3     name char(25),  
4     regionkey int references 'region'('regionkey'),  
5     comment varchar(152)  
6 );
```

```

7
8 alter table nation
9 add foreign key (regionkey) references region(regionkey);

```

2.3 定义参照完整性的违约处理

定义国家表的参照完整性，当删除或修改被参照表记录时，设置参照表中相应记录的值为空值。

```

1 create table nation (
2     nationkey int primary key,
3     name char(25),
4     regionkey int,
5     comment varchar(152),
6     foreign key (regionkey) references region(regionkey)
7         on delete set null on update set null
8 );

```

2.4 删除参照完整性

删除国家表的外码。

先使用 show create table 查看外键名称。

```

1 show create table nation;

```

```

1 CREATE TABLE `nation` (
2     `nationkey` int(11) NOT NULL,
3     `name` char(25) DEFAULT NULL,
4     `regionkey` int(11) DEFAULT NULL,
5     `comment` varchar(152) DEFAULT NULL,
6     PRIMARY KEY (`nationkey`),
7     KEY `regionkey` (`regionkey`),
8     CONSTRAINT `nation_ibfk_1` FOREIGN KEY (`regionkey`) REFERENCES `region` (`
9         regionkey`)
10 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci

```

可以看到，外键名称是 nation_ibfk_1。

```

1 alter table nation drop foreign key nation_ibfk_1;

```

2.5 插入一条国家记录，验证参照完整性是否起作用

```

1 insert into nation values (1001, 'nation1', 1001, 'comment1');

```

Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails ('lab2':nation', CONSTRAINT 'nation_ibfk_1' FOREIGN KEY ('regionkey') REFERENCES 'region' ('region-key'))

2.6 实验总结

参照完整性是建立在别的表的实体完整性之上的。在插入数据时，会验证别的表上是否存在这个 primary key，若存在才允许插入。删除实体完整性时，数据库会根据建表时定义的违约情况进行设置空值或级联删除的处理。

3 实验 3.3 用户自定义完整性实验

3.1 定义属性 NULL / NOT NULL 约束

定义地区表各属性的 NULL / NOT NULL 属性。

```
1 create table region (  
2     regionkey int not null primary key,  
3     name char(25) not null,  
4     comment varchar(152) null  
5 );
```

3.2 定义属性 DEFAULT 约束

定义国家表的 regionkey 的缺省属性值为 0 值，表示其他地区。

```
1 create table nation (  
2     nationkey int primary key,  
3     name char(25),  
4     regionkey int default 0,  
5     comment varchar(152),  
6     foreign key (regionkey) references region(regionkey)  
7 );
```

3.3 定义属性 UNIQUE 约束

定义国家表的名称属性必须唯一的完整性约束。

```
1 create table nation (  
2     nationkey int primary key,  
3     name char(25) unique,  
4     regionkey int default 0,  
5     comment varchar(152)  
6 );
```

3.4 使用 check

使用 check 定义 regionkey 所要满足的约束。

```
1 create table region (  
2     regionkey integer primary key,  
3     name char(25) not null,  
4     check (regionkey >= 100),
```

```
5      check (regionkey <= 200)
6  );
```

3.5 添加一条记录，验证是否违反 check 约束

```
1 insert into region values (99, '123');
```

1 row(s) affected

结果是插入成功了。查阅资料知，*MYSQL* 不支持 *check* 约束，但可以使用 *check* 约束，而没有任何效果。所以插入数据时 *check* 约束没有起作用。

3.6 实验总结

通过实验，我知道了 *NULL* 约束允许空值，*NOT NULL* 约束禁止空值；*DEFAULT* 约束在未指定值时默认填入 *DEFAULT* 值；*UNIQUE* 约束保证属性必须唯一；*CHECK* 约束是某些属性必须满足的约束。而在 *Mysql* 中，*check* 约束未实现，其他约束都可以正常使用。