

## 外键约束(参照完整性)

外键约束用于建立主键所在的表（主表）和外键所在的表（从表）之间的数据联系。可以将表中主键值的一列或多列添加到另一个表中创建两个表之间的连接。

注意：从表的外键列名必须是主表的主键列名，且在删除主表时必须删除外键约束或者先删除从表的数据（记录），再删除主表的数据（记录）。

MySQL 外键约束（FOREIGN KEY）是表的一个特殊字段，经常与主键约束一起使用。对于两个具有关联关系的表而言，相关联字段中主键所在的表就是主表（父表），外键所在的表就是从表（子表）。

外键用来建立主表与从表的关联关系，为两个表的数据建立连接，约束两个表中数据的一致性和完整性。主表删除某条记录时，从表中与之对应的记录也必须有相应的改变。一个表可以有一个或多个外键，外键可以为空值，若不为空值，则每一个外键的值必须等于主表中主键的某个值。

定义外键时，需要遵守下列规则：

- 主表必须已经存在于数据库中，或者是当前正在创建的表。如果是后一种情况，则主表与从表是同一个表，这样的表称为自参照表，这种结构称为自参照完整性。
- 必须为主表定义主键。
- 主键不能包含空值，但允许在外键中出现空值。也就是说，只要外键的每个非空值出现在指定的主键中，这个外键的内容就是正确的。
- 在主表的表名后面指定列名或列名的组合。这个列或列的组合必须是主表的主键或候选键。
- 外键中列的数目必须和主表的主键中列的数目相同。
- 外键中列的数据类型必须和主表主键中对应列的数据类型相同。

### 1、在创建表时设置外键约束

在 CREATE TABLE 语句中，通过 FOREIGN KEY 关键字来指定外键，具体的语法格式如下：

```
[CONSTRAINT <外键名>] FOREIGN KEY 字段名 [, 字段名 2, ...]  
REFERENCES <主表名> 主键列 1 [, 主键列 2, ...]  
[ON DELETE {RESTRICT | CASCADE | SET NULL | NO ACTION}]  
[ON UPDATE {RESTRICT | CASCADE | SET NULL | NO ACTION}]
```

“ON DELETE”和“ON UPDATE”可以为每个外键定义指定动作。指定动作包含两部分，第一部分指定这个指定动作由什么触发，是 UPDATE 或是 DELETE 语句；第二部分指定接下来采取什么行动，例如 RESTRICT、CASCADE、SET NULL、NO ACTION 和 SET DEFAULT。

- RESTRICT：当要删除或更新父表中被参照列上在外键出现的时候，拒绝对父表的删除或更新操作。
- CASCADE：从父表删除或更新行时，自动删除或更新子表中匹配的行。
- SET NULL：当从父表删除或更新时，设置子表中与之对应的外键列为 NULL。如果外键列没有指定 NOT NULL 限定，这就是合法的。
- NO ACTION：意味着不采取动作，即如果有一个相关的外键值在被参考的表里，删除或更新父表中该值将不被允许，应用和 RESTRICT 一样。
- SET DEFAULT：作用和 SET NULL 一样，只不过 SET DEFAULT 是指定子表中的外键列为默认值。

如果没有指定动作，两个参照动作就会默认地使用 RESTRICT 命令。

MySQL 参照完整性约束目前只可以用在那些使用 InnoDB 存储引擎创建的表中，对于其他类型的表。MySQL 服务器能够解析 CREATE TABLE 语句中的 FOREIGN KEY 语法，但不能使

用或保存它。

要修改表的存储引擎，可以采用 ALTER TABLE 语句修改表的存储引擎为 InnoDB。

注意：从表的外键关联的必须是主表的主键，且主键和外键的数据类型必须一致。例如，两者都是 INT 类型，或者都是 CHAR 类型。如果不满足这样的要求，在创建从表时，就会出现“ERROR 1005(HY000): Can't create table”错误。

【例 5-5】为 Lib 数据库创建 reader\_bk 表，表结构参照 reader 表。其中 schoolNo 参照 school 表的 schoolNo。在 MySQL 的命令行格式中，输入以下命令：

```
CREATE TABLE reader_bk (
    readerNo char(4) NOT NULL,
    readerName varchar(10) NOT NULL,
    gender char(2) NOT NULL,
    phoneNumber varchar(12) DEFAULT NULL,
    schoolNo char(4) NOT NULL,
    effectDate datetime DEFAULT NULL,
    lostEffectDate datetime DEFAULT NULL,
    breakRules char(2) DEFAULT NULL,
    borrowAdd int(11) DEFAULT NULL,
    PRIMARY KEY (readerNo),
    KEY FK_Reader_bk_school (schoolNo),
    CONSTRAINT FK_Reader_bk_school FOREIGN KEY (schoolNo) REFERENCES school
(schoolNo)
    ON DELETE CASCADE
    ON UPDATE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

这样设置 reader\_bk 表的外键 schoolNo 参照 school 表的 schoolNo，当 school 表的 schoolNo 发生删除或修改时，对应 reader\_bk 表中的 schoolNo 的值也会发生删除或修改。

## 2、在修改表时添加外键约束

外键约束也可以在修改表时添加，但是添加外键约束的前提是：从表中外键列中的数据必须与主表中主键列中的数据一致或者是没有数据。

在修改数据表时添加外键约束的语法格式如下：

```
ALTER TABLE <数据表名> ADD CONSTRAINT <外键名>
FOREIGN KEY(<列名>) REFERENCES <主表名> (<列名>);
```

注意：在为已经创建好的数据表添加外键约束时，要确保添加外键约束的列的值全部来源于主键列，并且外键列不能为空。

【例 5-6】假设当前的 book 表并没有将“classNo”列设为外键参照与 bookClass 表的“classNo”，请添加名为“FK\_Book\_BookClass”外键。在 MySQL 的命令行格式中，输入以下命令：

```
ALTER TABLE book ADD CONSTRAINT FK_Book_BookClass
FOREIGN KEY (classNo) REFERENCES bookClass(classNo);
```

## 3、删除外键约束

当一个表中不需要外键约束时，就需要从表中将其删除。外键一旦删除，就会解除主表和从表间的关联关系。

删除外键约束的语法格式如下所示：

```
ALTER TABLE <表名> DROP FOREIGN KEY <外键约束名>;
```

MySQL 删除外键时，还必须删除之前因为外键存在系统创建的同名索引，请参阅下例。

**【例 5-7】**删除 book 表的外键。在 MySQL 的命令行格式中，输入以下命令：

```
ALTER TABLE book DROP FOREIGN KEY FK_Book_BookClass;
```

```
ALTER TABLE book DROP INDEX FK_Book_BookClass;
```