

狂神说MySQL03：DML语言

秦疆 狂神说 2020-04-29

狂神说MySQL系列连载课程，通俗易懂，基于MySQL5.7.19版本，欢迎各位狂粉转发关注学习。禁止随意转载，转载记住贴出B站视频链接及公众号链接！



上课视频同步文档



MySQL数据管理

外键

外键概念

如果公共关键字在一个关系中是主关键字，那么这个公共关键字被称为另一个关系的外键。由此可见，外键表示了两个关系之间的相关联系。以另一个关系的外键作主关键字的表被称为**主表**，具有此外键的表被称为**从表**。

在实际操作中，将一个表的值放入第二个表来表示关联，所使用的值是第一个表的主键值(在必要时可包括复合主键值)。此时，第二个表中保存这些值的属性称为**外键(foreign key)**。

外键作用

保持数据**一致性**，**完整性**，主要目的是控制存储在外键表中的数据**约束**。使两张表形成关联，外键只能引用外表中的列的值或使用空值。

创建外键

建表时指定外键约束

-- 创建外键的方式一：创建子表同时创建外键

-- 年级表 (id\年级名称)

```
CREATE TABLE `grade` (  
  `gradeid` INT(10) NOT NULL AUTO_INCREMENT COMMENT '年级ID',  
  `gradename` VARCHAR(50) NOT NULL COMMENT '年级名称',  
  PRIMARY KEY (`gradeid`)
```

```
) ENGINE=INNODB DEFAULT CHARSET=utf8

-- 学生信息表 (学号,姓名,性别,年级,手机,地址,出生日期,邮箱,身份证号)
CREATE TABLE `student` (
  `studentno` INT(4) NOT NULL COMMENT '学号',
  `studentname` VARCHAR(20) NOT NULL DEFAULT '匿名' COMMENT '姓名',
  `sex` TINYINT(1) DEFAULT '1' COMMENT '性别',
  `gradeid` INT(10) DEFAULT NULL COMMENT '年级',
  `phoneNum` VARCHAR(50) NOT NULL COMMENT '手机',
  `address` VARCHAR(255) DEFAULT NULL COMMENT '地址',
  `borndate` DATETIME DEFAULT NULL COMMENT '生日',
  `email` VARCHAR(50) DEFAULT NULL COMMENT '邮箱',
  `idCard` VARCHAR(18) DEFAULT NULL COMMENT '身份证号',
  PRIMARY KEY (`studentno`),
  KEY `FK_gradeid` (`gradeid`),
  CONSTRAINT `FK_gradeid` FOREIGN KEY (`gradeid`) REFERENCES `grade` (`gradeid`)
) ENGINE=INNODB DEFAULT CHARSET=utf8
```

建表后修改

```
-- 创建外键方式二：创建子表完毕后,修改子表添加外键
ALTER TABLE `student`
ADD CONSTRAINT `FK_gradeid` FOREIGN KEY (`gradeid`) REFERENCES `grade` (`gradeid`);
```

删除外键

操作：删除 **grade** 表，发现报错



注意：删除具有主外键关系的表时，要先删子表，后删主表

```
-- 删除外键
ALTER TABLE student DROP FOREIGN KEY FK_gradeid;
-- 发现执行完上面的,索引还在,所以还要删除索引
-- 注:这个索引是建立外键的时候默认生成的
ALTER TABLE student DROP INDEX FK_gradeid;
```

DML语言

数据库意义：数据存储、数据管理

管理数据库数据方法：

- 通过SQLyog等管理工具管理数据库数据
- 通过DML语句管理数据库数据

DML语言：数据操作语言

- 用于操作数据库对象中所包含的数据
- 包括：
 - INSERT (添加数据语句)

- UPDATE (更新数据语句)
- DELETE (删除数据语句)

添加数据

INSERT命令

语法：

```
INSERT INTO 表名 [(字段1, 字段2, 字段3, ...)] VALUES ('值1', '值2', '值3')
```

注意：

- 字段或值之间用英文逗号隔开。
- '字段1, 字段2...' 该部分可省略，但添加的值必与表结构, 数据列, 顺序相对应, 且数量一致。
- 可同时插入多条数据，values 后用英文逗号隔开。

```
-- 使用语句如何增加语句？
-- 语法：INSERT INTO 表名 [(字段1, 字段2, 字段3, ...)] VALUES ('值1', '值2', '值3')
INSERT INTO grade (gradename) VALUES ('大一');

-- 主键自增, 那能否省略呢？
INSERT INTO grade VALUES ('大二');

-- 查询: INSERT INTO grade VALUE ('大二') 错误代码: 1136
Column count doesn't match value count at row 1

-- 结论: '字段1, 字段2...' 该部分可省略，但添加的值必与表结构, 数据列, 顺序相对应, 且数量一致。

-- 一次插入多条数据
INSERT INTO grade (gradename) VALUES ('大三'), ('大四');
```

练习题目

自己使用INSERT语句为课程表subject添加数据。使用到外键。

修改数据

update命令

语法：

```
UPDATE 表名 SET column_name=value [, column_name2=value2, ...] [WHERE condition];
```

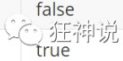
注意：

- column_name 为要更改的数据列
- value 为修改后的数据，可以为变量，具体指，表达式或者嵌套的SELECT结果
- condition 为筛选条件，如不指定则修改该表的所有列数据

where条件子句

可以简单的理解为：有条件地从表中筛选数据

运算符	含义	范围	结果
=	等于	5=6	false
<> 或 !=	不等于	5!=6	true
>	大于	5>6	false
<	小于	5<6	true
>=	大于等于	5>=6	false
<=	小于等于	5<=6	true
BETWEEN	在某个范围之间	BETWEEN 5 AND 10	
AND	并且	5 > 1 AND 1 > 2	false
OR	或	5 > 1 OR 1 > 2	true



测试：

```
-- 修改年级信息
UPDATE grade SET gradename = '高中' WHERE gradeid = 1;
```

删除数据

DELETE命令

语法：

```
DELETE FROM 表名 [WHERE condition];
```

注意：condition为筛选条件，如不指定则删除该表的所有列数据

```
-- 删除最后一个数据
DELETE FROM grade WHERE gradeid = 5
```

TRUNCATE命令

作用：用于完全清空表数据，但表结构，索引，约束等不变；

语法:

```
TRUNCATE [TABLE] table_name;

-- 清空年级表
TRUNCATE grade
```

注意：区别于**DELETE**命令

- 相同：都能删除数据，不删除表结构，但TRUNCATE速度更快
- 不同：
 - 使用TRUNCATE TABLE 重新设置AUTO_INCREMENT计数器
 - 使用TRUNCATE TABLE不会对事务有影响（事务后面会说）

测试:

```
-- 创建一个测试表
CREATE TABLE `test` (
  `id` INT(4) NOT NULL AUTO_INCREMENT,
  `coll` VARCHAR(20) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=INNODB DEFAULT CHARSET=utf8

-- 插入几个测试数据
INSERT INTO test(coll) VALUES('row1'),('row2'),('row3');

-- 删除表数据(不带where条件的delete)
DELETE FROM test;
-- 结论:如不指定where则删除该表的所有列数据,自增当前值依然从原来基础上进行,会记录日志.

-- 删除表数据(truncate)
TRUNCATE TABLE test;
-- 结论:truncate删除数据,自增当前值会恢复到初始值重新开始;不会记录日志.

-- 同样使用DELETE清空不同引擎的数据库表数据.重启数据库服务后
-- InnoDB：自增列从初始值重新开始（因为是存储在内存中,断电即失）
-- MyISAM：自增列依然从上一个自增数据基础上开始（存在文件中,不会丢失）
```



视频同步更新

如果觉得帮助到了您，不妨赞赏支持一下吧！



“赠人玫瑰，手有余香”

狂神说的赞赏码

 狂神说

