

MySQL之DDL语言知多少

01 MySQL语言分类

MySQL作为最流行的关系型数据库之一，有着和其他主流数据库几乎一致的SQL语法。相较于其他编程语言来说，虽然SQL语法比较简单，关键字也比较少，但实际也是麻雀虽小五脏俱全的。

一般来说，SQL语言主要分为三类：

- DML (Data Manipulation Language) ，即数据操纵语言，最常用的增删改查就属于DML，操作对象是数据表中的记录；
- DDL (Data Definition Language) ，即数据定义语言，例如建数据库、建表等，都属于数据定义语言，也是今天本文的主角；
- DCL (Data Control Language) ，数据控制语言，如Grant、Rollback等等，常见于数据库安全管理，多数人一般很少用。

如果DML再进一步细分的话，那么增删改可算是纯粹的DML，而查叫做DQL (Data Query Language) ，足见查询SQL地位之重要。

本文主要介绍DDL中的"增删改查"，对应关键字：

- 增：Create
- 删：Drop / Truncate
- 改：Alter + add/drop/modify/change
- 查：Show / Desc

严格的讲，DDL中的增删改查对象不只是database和table，还有Procedure、Function、Trigger、Index、View等等，但今天本文仅聚焦于database和table。

02 Create

在MySQL中创建数据库和创建表中，一定会用到Create语法。

- 创建数据库

创建数据库语法简单，最简洁的建库语句如下：

```
CREATE DATABASE <数据库名>;
```

一般来说，Create语法都可以在对象之后增加If not exists，用于处理创建表已存在的情况，此时仅会发出警告而不会报错。

同时，还可以为新创建的数据库指定字符集 charset 和 校对规则 collate：

```
CREATE DATABASE [IF NOT EXISTS] <数据库名>
[[DEFAULT] CHARACTER SET <字符集名>]
[[DEFAULT] COLLATE <校对规则>];
```

如果不指定字符集和校对规则，所建数据库默认为：utf8和utf8_general_ci。当然，需注意的是MySQL中的utf8字符集实际上是一个伪utf-8，真正意义上的utf-8在MySQL中是utf8mb4(utf-8 more bytes 4)。具体可查阅相关文档介绍。

- 创建表

相较于建库而言，建表的可选项较多，当然这里的可选项是指针对定义表中列字段而言。与建库类似，create建表也支持if not exists语法，用来处理表重复的情形。

常见的建表语句例如：

```
CREATE Table [IF NOT EXISTS] <数据表名>
(字段名1 类型 (长度) [DEFAULT] [NOT NULL] ,
..... ,
[PRIMARY KEY],
[FOREIGN KEY ])
```

还可以在列字段后指定引擎信息，例如不想使用默认引擎innodb，而想使用MyISAM引擎，则可在建表语句后增加如下语法：

```
CREATE Table [IF NOT EXISTS] <数据表名> (.....) ENGINE = MyISAM;
```

当想从一个表中筛选若干数据来建立一个新表时，可以在表名后接as语句。如下语句将使用表tb中数据创建一个一样表结构和表记录的新数据表：

```
CREATE Table [IF NOT EXISTS] <数据表名> AS SELECT * FROM tb;
```

03 DROP

与Create对应的关键字是Drop，且其操作对象也几乎一致：不仅可以Drop数据库和数据表，也可以drop函数、过程、索引等等。当然，这里还是主要介绍drop数据库和数据表。

- 删数据库

与create类似、又比create更为简单，drop语句用法有限，没有太多可定义的操作选项，仅需增加drop对象的关键字即可。所以在drop数据库时，即

```
DROP Database <数据库名> ;
```

再一次与create类似，为了防止数据库不存在时删除引发错误，可增加存在性判断。当然，创建时是要判断是否不存在，而drop时则判断是否存在：

```
DROP Database [If Exists] <数据库名> ;
```

- 删数据表

常规的删数据表与删数据库几乎完全一致，仅需更改关键字和对应表名即可：

```
DROP Table [If Exists] <数据表名> ;
```

除了Drop关键字删数据表外，还有另外一个关键字可以删数据表，即Truncate，英文截断的意思。如其名字描述的那样，Truncate与Drop(完全丢弃)不同，用Truncate删除数据表时仅是"截断"记录数据，而保留数据表的结构信息。

```
Truncate Table <数据表名> ;
```

某种意义上，Truncate的效果与Delete 不加限定条件时的效果一致，但其速度更快；且由于truncate是数据定义语言，其操作对象不是记录，所以不支持事务和触发器等。

04 Alter

对于已定义的数据库和数据表来说，如果想修改其中的某些选项和信息，此时就要用到alter关键字。Alter英文即是更改，键盘的alt即为其缩写。

因为广义上的更改含义有多种，例如增加或删除个字段叫更改，修改一些选项信息也叫更改，所以Alter其实又常常配套以下几个附属关键字：

- set

alter与set配套使用常用于修改字段默认值等信息，例如：

```
ALTER TABLE <已有数据表> ALTER <字段> SET DEFAULT <默认值>;
```

- add

对数据表增加字段：

```
ALTER TABLE <数据表名> ADD <新字段名> <数据类型> [.....];
```

- drop

有增加即有删减，对已有数据表删减字段的语法为：

```
ALTER TABLE <数据表名> DROP <已有字段名>;
```

- modify

对已有表进行修改，例如修改其数据类型等：

```
ALTER TABLE <数据表名> MODIFY <已有字段名> <数据类型>;
```

- change

修改表信息的另一个关键字是应用change，例如仍然修改某个字段的数据类型，则应用change语法为：

```
ALTER TABLE <数据表名> CHANGE <已有字段名> <已有字段名> <数据类型>;
```

注意到其与modify的一个重要不同是需要写字段名2次，所以实际上change还可用于更改字段名，即将新字段名放于已有字段名之后即可，若字段名相同意味着不修改，也不会报错。

- rename

modify和change都是用于更改列字段信息，rename则是用于更改表名，其语法为：

```
ALTER TABLE <已有数据表名> RENAME TO <新数据表名>;
```

05 Show

与DML中查询最为复杂不同，DDL中的查询用法其实最为简单。主要是应用Show关键字进行显示查询：

例如，显示数据库创建信息：

```
Show Create Database <数据库名>;
```

类似的，可用如下语句查询数据表创建信息：

```
Show Create Table <数据表名>;
```

如果应用Desc关键字，则可用于显示数据表结构信息：

```
Desc <数据表名>;
```

注意，desc查询的数据表信息，包括各字段名、数据类型、默认值等等；而show查询返回的则是创建该数据表时的SQL语句。

另外，还有一个可算是查询的用法，即为了查询当前应用的是哪个数据库时，可用如下语句：

```
Select Database();
```

返回的当前应用数据库名。值得注意的是这里的database()算作内置函数，与select version()查询MySQL版本信息用法类似。

06 总结

本文对MySQL中面向数据库和数据表的DDL常用增删改查进行了简单介绍，包括Create、Drop、Alter、Show等关键字的基本语法。

当然，仅限于介绍对数据库和数据表的操作，这些关键字同样适用于索引、存储过程、函数等对象。