

# SQL 通配符

2018-09-04 16:38

通配符可以用于替代字符串中的任何其他字符。

SQL中，通配符经常与SQL like操作符一起使用。

通配符	描述
%	替代 0 个或多个字符
_	替代一个字符
[charlist]	字符列中的任何单一字符
[^charlist]或[!charlist]	不在字符列中的任何单一字符

举例：

% 通配符：

```
Select * from websites where url like 'http%';
```

\_ 通配符：

```
Select * from websites where name like 'G_o_gle';
```

[charlist]通配符：

```
Select * from websites where name REGEXP '^[GFs]';
```

备注：mySQL只支持%和\_通配符，要完成[charlist]和[^charlist]通配符的查询效果，要使用regexp或not regexp运算符（或RLIKE和 not RLIKE）来操作正则表达式。

# Drop、Delete、truncate三者之间的区别

2018-09-04 15:48



SQL关于删除的三个语句：DROP、TRUNCATE、DELETE 的区别。

## DROP:

```
DROP test;
```

删除表test，并释放空间，将test删除的一干二净。

## TRUNCATE:

```
TRUNCATE test;
```

删除表test里的内容，并释放空间，但不删除表的定义，表的结构还在。

## DELETE:

### 1、删除指定数据

删除表test中年龄等于30的且国家为US的数据

```
DELETE FROM test WHERE age=30 AND country='US';
```

### 2、删除整个表

仅删除表test内的所有内容，保留表的定义，不释放空间。

```
DELETE FROM test 或者 DELETE FROM test;  
DELETE * FROM test 或者 DELETE * FROM test;
```

zt 8个月前 [01-12]



`truncate table` 命令将快速删除数据表中的所有记录，但保留数据表结构。这种快速删除与 `delete from` 数据表的删除全部数据表记录不一样，`delete` 命令删除的数据将存储在系统回滚段中，需要的时候，数据可以回滚恢复，而 `truncate` 命令删除的数据是不可以恢复的。

## 相同点

`truncate` 和不带 `where` 子句的 `delete`，以及 `drop` 都会删除表内的数据。

## 相同点

truncate 和不带 where 子句的 delete, 以及 drop 都会删除表内的数据。

## 不同点:

1. truncate 和 delete 只删除数据不删除表的结构(定义), drop 语句将删除表的结构被依赖的约束(constrain), 触发器(trigger), 索引(index); 依赖于该表的存储过程/函数将保留, 但是变为 invalid 状态。
2. delete 语句是 dml, 这个操作会放到 rollback segment 中, 事务提交之后才生效; 如果有相应的 trigger, 执行的时候将被触发。 truncate, drop 是 ddl, 操作立即生效, 原数据不放到 rollback segment 中, 不能回滚。 操作不触发 trigger。
3. delete 语句不影响表所占用的 extent, 高水线(high watermark)保持原位置不动。 显然 drop 语句将表所占用的空间全部释放。 truncate 语句缺省情况下将空间释放到 minextents 个 extent, 除非使用 reuse storage; truncate 会将高水线复位(回到最开始)。
4. 速度: 一般来说: drop > truncate > delete。
5. 安全性: 小心使用 drop 和 truncate, 尤其没有备份的时候。否则哭都来不及。

使用上, 想删除部分数据用 delete, 注意带上 where 子句。 回滚段要足够大。

想删除表, 当然用 drop。

想保留表而将所有数据删除。如果和事务无关, 用 truncate 即可。如果和事务有关, 或者想触发 trigger, 还是用 delete。

# SQL注意事项

2018-09-04 14:06

文本字段vs数值字段

SQL使用单引号来环绕文本值（大部分数据库系统也接受双引号），文本值得内容区分大小写，例如：

```
Select * from websites where country="CN";
```

如果是数值字段，不须使用引号，例如：

```
Select * from Websites where id=1;
```

# NULL值

2018-09-05 15:50

## NULL

NULL值代表遗漏的位置数据。

默认地，表的列可以存放NULL值，除非特定说明该列not null。

如果表中的某个列是可选的，那么我们可以在不向该列添加值的情况下插入新记录或更新已有的记录。这意味着该字段将以 NULL 值保存。

## IS NULL & IS NOT NULL

在测试NULL值时，不能通过比较运算符来进行测试，需要使用IS NULL和IS NOT NULL来进行测试

举例：

```
SELECT LastName,FirstName,Address FROM Persons  
WHERE Address IS (NOT) NULL
```