

MySQL避免重复插入记录方法 (ignore, Replace, ON DUPLICATE KEY UPDATE)

本文章来给大家提供三种在mysql中避免重复插入记录方法，主要是讲到了ignore, Replace, ON DUPLICATE KEY UPDATE三种方法，各位同学可尝试参考。

1.使用ignore关键字

如果是用主键primary或者唯一索引unique区分了记录的唯一性,避免重复插入记录可以使用：

代码如下

```
INSERT IGNORE INTO table_name (email, phone, user_id) VALUES ('test9@163.com', '99999', '9999');
```

这样当有重复记录就会忽略,执行后返回数字0

2.使用Replace

语法格式：

代码如下

```
REPLACE INTO table_name (col_name, ...) VALUES (...);
```

```
REPLACE INTO table_name (col_name, ...) SELECT ...;
```

```
REPLACE INTO table_name SET col_name='value';
```

...算法说明：

REPLACE的运行与INSERT很相像,但是如果旧记录与新记录有相同的值，则在新记录被插入之前，旧记录被删除，即：

尝试把新行插入到表中

当因为对于主键或唯一关键字出现重复关键字错误而造成插入失败时：

从表中删除含有重复关键字值的冲突行

再次尝试把新行插入到表中

旧记录与新记录有相同的值的判断标准就是：

表有一个PRIMARY KEY或UNIQUE索引，否则，使用一个REPLACE语句没有意义。该语句会与INSERT相同，因为没有索引被用于确定是否新行复制了其它的行。

返回值：

REPLACE语句会返回一个数，来指示受影响的行的数目。该数是被删除和被插入的行数的和

受影响的行数可以容易地确定是否REPLACE只添加了一行，或者是否REPLACE也替换了其它行：检查该数是否为1（添加）或更大（替换）。

示例:

eg:(phone字段为唯一索引)

代码如下

```
REPLACE INTO table_name (email, phone, user_id) VALUES ('test569', '99999', '123');
```

3.ON DUPLICATE KEY UPDATE

ON DUPLICATE KEY UPDATE作用

先声明一点，ON DUPLICATE KEY UPDATE为Mysql特有语法，这是个坑语句的作用，当insert已经存在的记录时，执行Update

用法

举个例子：

user_admin_t表中有一条数据如下

_id	password	create_time	modify_time	data_status	role	admin_status	reserved_1	reserved_2	username	name
2	21232F297A57A5A743894A	2017-10-14 01:37:49	2017-10-14 01:37:51	1	普通管理员	2	(Null)	(Null)	普通管理员	普通管理员

表中的主键为id，现要插入一条数据，id为'1'，password为'第一次插入的密码'，正常写法为：

```
INSERT INTO user_admin_t (_id,password) VALUES ('1','第一次插入的密码')
```

执行后刷新表数据，我们来看表中内容

_id	password	create_time	modify_time	data_status	role	admin_status	reserved_1	reserved_2	username	name
1	第一次插入的密码	(Null)	(Null)	1	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)
2	21232F297A57A5A743894A	2017-10-14 01:37:49	2017-10-14 01:37:51	1	普通管理员	2	(Null)	(Null)	普通管理员	普通管理员

此时表中数据增加了一条主键'_id'为'1'，'password'为'第一次插入的密码'的记录，当我们再次执行插入语句时，会发生什么呢？

-- 执行

```
INSERT INTO user_admin_t (id,password) VALUES ('1','第一次插入的密码')
```

[Err] 1062 - Duplicate entry '1' for key 'PRIMARY'

Mysql告诉我们，我们的主键冲突了，看到这里我们是不是可以改变一下思路，当插入已存在主键的记录时，将插入操作变为修改：

-- 在原sql后面增加 ON DUPLICATE KEY UPDATE

```
INSERT INTO user_admin_t (_id,password) VALUES ('1','第一次插入的密码')
ON DUPLICATE KEY UPDATE
_id = 'upId',password = 'upPassword';
```

我们再一次执行：

[SQL]

```
INSERT INTO user_admin_t (_id,password) VALUES ('1','第一次插入的密码')
ON DUPLICATE KEY UPDATE
_id = 'upId',password = 'upPassword';
```

受影响的行: 2
时间: 0.131s

可以看到 受影响的行为2，这是因为将原有的记录修改了，而不是执行插入，看一下表中数据：

2	21232F297A57A5A743894A0	2017-10-14 01:37:49	2017-10-14 01:37:51	1	普通管理员	2	(Null)	(Null)	普通管理员	普通管理员
Upld	upPassword	(Null)	(Null)	1	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)

原本'id'为'1'的记录，改为了'Upld'，'password'也变为了'upPassword'，很好的解决了重复插入问题