
创建触发器

1. 创建 BEFORE 类型触发器

BEFORE 类型触发操作，是在执行了监视动作前，会执行触发事件。

当触发器涉及到对触发器自身的表更新操作时，只能使用 BEFORE 类型触发器，不可使用 AFTER 类型触发器。另外，如果更新的是同一行数据，须直接更新新行的变量，即使用 new 关键字。

【例 8-27】在 borrow 表中创建触发器，当读者每次还书时，在登记还书时间前，须先将该次借书行为的 re_turn（是否归还）设为是，在命令行窗口中输入如下命令：

```
delimiter $$
create trigger tr_borrow_returnData before update
  on borrow for each row
begin
  if new.returnDate is not null then
    set new.re_turn='是';
  end if;
end$$
delimiter ;
```

触发器创建好后，每当读者归还一本书，登记还书日期，则该次借书行为的 re_turn（是否归还）将被先设为是。

```
mysql> delimiter $$
mysql> create trigger tr_borrow_returnData before update
->   on borrow for each row
-> begin
->   if new.returnDate is not null then
->     set new.re_turn='是';
->   end if;
-> end$$
Query OK, 0 rows affected (0.10 sec)

mysql> delimiter ;
```

现在验证一下，先查看 borrowNo（借书号）为 10 的书是否归还。

```
select * from borrow where borrowNo=10;
```

```
mysql> select * from borrow where borrowNo=10;
```

borrowNo	bookNo	readerNo	borrowDate	returnDate	re_new	re_turn
10	b196	r029	2017-03-07 00:00:00	NULL	否	NULL

```
1 row in set (0.01 sec)
```

接下来，设置该次借书完成，还书时间为 2017 年 6 月 7 日，再次查看 re_turn 记录。

```
update borrow set returnDate='2017-06-07' where borrowNo=10;
```

```
select * from borrow where borrowNo=10;
```

```
mysql> update borrow set returnDate='2017-06-07' where borrowNo=10;
Query OK, 1 row affected (0.03 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> select * from borrow where borrowNo=10;
```

borrowNo	bookNo	readerNo	borrowDate	returnDate	re_new	re_turn
10	b196	r029	2017-03-07 00:00:00	2017-06-07 00:00:00	否	是

```
1 row in set (0.00 sec)
```

可见，该次借书行为的 re_turn（是否归还）被自动设为“是”。

如果该触发器使用 after 触发器，该触发器将不能建立，系统提示无法创建 after trigger。

```
mysql> drop trigger tr_borrow_returnData;
Query OK, 0 rows affected (0.03 sec)

mysql> delimiter $$
mysql> create trigger tr_borrow_returnData after update
-> on borrow for each row
-> begin
-> if new.returnDate is not null then
-> set new.re_turn='是';
-> end if;
-> end$$
ERROR 1362 (HY000): Updating of NEW row is not allowed in after trigger
```

2. 创建 AFTER 类型触发器

AFTER 类型触发操作，是在执行了监视动作后，才会执行触发事件。

【例 8-28】在 borrow 表中创建一个 after 类型触发器，当读者还书时，需登记该书的还书时间，将该书的借书行为 re_turn（是否归还）设为是，然后将该书籍剩余册数加 1，在命令行窗口中输入如下命令：

```
delimiter $$
create trigger tr_borrow_return after update
on borrow for each row
begin
if new.re_turn='是' then
update book set number=number+1 where bookNo=new.bookNo;
end if;
end$$
delimiter ;
```

触发器 tr_borrow_return 创建之后，每当 borrow 因为还书新修改一条记录，就会产生一次还书行为，该书剩余数量应该增加了一本。

现在验证一下该触发器，如果 readerNo 为 ‘r003’ 的同学，将借的 bookNo 为 ‘b003’ 的书归还。

首先在查看 ‘b003’ 书的剩余册数：

```
select bookNo,bookName,number from book where bookNo='b003';
```

```
mysql> select bookNo,bookName,number from book where bookNo='b003';
```

bookNo	bookName	number
b003	红楼梦	10

```
1 row in set (0.00 sec)
```

现在完成还书动作，在 borrow 表中修改该书的还书日期为 2021 年 1 月 10 日：

```
update borrow set returnDate=20210110, re_turn='是'
where bookNo='b003' and readerNo='r003' and re_turn='否';
然后再次查看 'b003' 书的剩余册数：
select bookNo,bookName,number from book where bookNo='b003';
```

```
mysql> update borrow set returnDate=20210110, re_turn='是'
-> where bookNo='b003' and readerNo='r003' and re_turn='否';
Query OK, 1 row affected (0.05 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> select bookNo,bookName,number from book where bookNo='b003';
```

bookNo	bookName	number
b003	红楼梦	11

```
1 row in set (0.00 sec)
```

【例 8-29】在 borrow 表中创建一个触发器，当读者借书时，需登记该书的图书编号、读者编号、借书时间，然后将该书籍剩余册数减 1，在命令行窗口中输入如下命令：

```
delimiter $$
create trigger tr_borrow_new after insert
on borrow for each row
begin
update book set number=number-1 where bookNo=new.bookNo;
end$$
delimiter ;
```

触发器 tr_borrow_new 创建之后，每当 borrow 因为新增加一条记录，就会产生一次借书行为，该书剩余数量应该减少了一本。

现在验证一下，如果 readerNo 为 'r003' 的同学，于当前日期借了一本 bookNo 为 'b003' 的书。

首先在【例 8-2】中已经查看到 'b003' 书的剩余册数为 11 本。

现在完成借书动作，在 borrow 表增加一行借书的数据：

```
insert borrow(bookNo,readerNo,borrowDate) values('b003','r003',curdate());
然后查看这次借书行为并再次查看 'b003' 书的剩余册数：
select * from borrow where bookNo='b003' and readerNo='r003';
select bookNo,bookName,number from book where bookNo='b003';
```

```
mysql> insert borrow(bookNo,readerNo,borrowDate) values('b003','r003',curdate());
Query OK, 1 row affected (0.11 sec)

mysql> select * from borrow where bookNo='b003' and readerNo='r003';
```

borrowNo	bookNo	readerNo	borrowDate	returnDate	re_new	re_turn
502	b003	r003	2021-01-01 00:00:00	2021-01-10 00:00:00	否	是
503	b003	r003	2021-03-18 00:00:00	NULL	否	否

```
4 rows in set (0.03 sec)

mysql> select bookNo,bookName,number from book where bookNo='b003';
```

bookNo	bookName	number
b003	红楼梦	10

```
1 row in set (0.00 sec)
```

【例 8-30】按图书馆借书规则，任何图书在只剩 1 本书的情况下称为孤本，禁止借出。需将图书表的 onShelf(是否在架)字段设置为否，禁止借出。反之，当有读者还了这本书，剩余数量超过 1 本时，需将图书表的 onShelf(是否在架)字段设置为是，允许借出。请在 book 表中创建一个触发器。

分析：该触发器要求当 book 表中修改 number 字段数据时触发修改本表 onShelf 字段，因此只能用 before 触发器。在命令行窗口中输入如下命令：

```
delimiter $$
create trigger tr_book_onShelf before update
on book for each row
begin
    if new.number<=1 then set new.onShelf='否';
    else set new.onShelf='是';
end if;
end$$
delimiter ;
```

现在验证一下该触发器，首先查看一下 bookNo 为 'b120' 这本书的在架情况和剩余数目，以及 bookNo 为 'b120' 这本书的借书情况。

```
select bookNo,bookName,onShelf,number from book where bookNo='b120';
select * from borrow where bookNo='b120';
```

```
mysql> select bookNo,bookName,onShelf,number from book where bookNo='b120';
```

bookNo	bookName	onShelf	number
b120	杨红樱（魔法老师）	否	1

```
1 row in set (0.04 sec)

mysql> select * from borrow where bookNo='b120';
```

borrowNo	bookNo	readerNo	borrowDate	returnDate	re_new	re_turn
30	b120	r086	2017-07-25 00:00:00	NULL	否	NULL
80	b120	r084	2018-07-10 00:00:00	NULL	否	NULL
274	b120	r054	2017-05-05 00:00:00	2017-08-03 00:00:00	是	是

```
3 rows in set (0.03 sec)
```

上图中看到有两位读者借书还未还，现在设定 readerNo 为 'r086' 的读者还书，先查看该次借书还书情况：

```
update borrow set returnDate='2018-10-01' where borrowNo=80;
```

```
select * from borrow where borrowNo=80;
```

```
mysql> update borrow set returnDate='2018-10-01' where borrowNo=80;  
Query OK, 1 row affected (0.04 sec)  
Rows matched: 1 Changed: 1 Warnings: 0
```

```
mysql> select * from borrow where borrowNo=80;
```

borrowNo	bookNo	readerNo	borrowDate	returnDate	re_new	re_turn
80	b120	r084	2018-07-10 00:00:00	2018-10-01 00:00:00	否	是

```
1 row in set (0.00 sec)
```

上图可见，readerNo 为 'r086' 的读者还书后，re_turn 字段自动设置为 '是'。然后再查看 bookNo 为 'b120' 这本书的在架情况和剩余数目。

```
select bookNo,bookName,onShelf,number from book where bookNo='b120';
```

```
mysql> select bookNo,bookName,onShelf,number from book where bookNo='b120';
```

bookNo	bookName	onShelf	number
b120	杨红樱（魔法老师）	是	2

```
1 row in set (0.00 sec)
```

上图可见，readerNo 为 'r086' 的读者还书后，book 表中 number 字段自动加 1，onShelf 字段自动设置为 '是'。

当再次有读者借阅这本书时，book 表中 number 字段自动减 1，onShelf 字段自动设置为 '否'：

```
insert borrow(bookNo,readerNo,borrowDate) values('b120','r003',curdate());
```

```
select bookNo,bookName,onShelf,number from book where bookNo='b120';
```

```
mysql> insert borrow(bookNo,readerNo,borrowDate) values('b120','r003',curdate());  
Query OK, 1 row affected (0.04 sec)
```

```
mysql> select bookNo,bookName,onShelf,number from book where bookNo='b120';
```

bookNo	bookName	onShelf	number
b120	杨红樱（魔法老师）	否	1

```
1 row in set (0.00 sec)
```