

## 一. 触发器基础

1. 触发器的意义：有时，我们希望某些语句能再事件发生后自动执行，比如：增加一个客户的州名到列表。始终保持其州名是大写的。
2. 触发器的种类：在mysql中，触发器一共有三种：INSERT, DELETE, UPDATE
3. 如何创建触发器：首先，需要了解创建触发器时，要给出四条信息。  
①触发器名。 ②关联的表。 ③相应的活动 (INSERT, DELETE, UPDATE) ④何时执行 (AFTER 或者BEFORE)
4. 例子：CREATE TRIGGER newproduct AFTER INSERT ON products  
FOR EACH ROW SELECT 'Product added'
5. 注意事项：每个表每个事件只允许创建一个触发器，因此，每个表**至多支持6个触发器**。

## 二. INSERT 触发器

1. NEW表：NEW是一个虚拟的表，可以在INSERT触发器中使用。里面的内容是被插入的行。在BEFORE INSERT中NEW表的值可以被更新。（允许改变被插入的值）
2. 使用示例：CREATE TRIGGER neworder AFTER INSERT ON orders FOR EACH ROW SELECT NEW.order\_num     -- 这个语句弥补了INSERT没有返回值的遗憾。

## 三. DELETE触发器

1. OLD表：OLD表也是一个虚拟的表，可以用来访问被删除的行。另外，OLD表中的值是只读的。不能更新。
2. 使用示例：由于DELETE的操作无法被撤销。所以，我们可以用触发器，把DELETE的行保存在一个表格中。（关于这点，如果订单不能存档，DELETE也会被放弃。）

## 四. UPDATE触发器

1. 同时具有NEW和OLD表：在UPDATE触发器中，你可以用OLD表访问修改之前的数据，也可以用NEW表，访问之前的数据。
2. 使用示例：CREATE TRIGGER updatevendor BEFORE UPDATE ON vendors  
FOR EACH ROW SET NEW.vend\_state = Upper(NEW.vend\_state);  
-- （这个专门用来保持一致的大写格式）

## 五. 总结

触发器的作用：

1. 返回相应不返回具体值的操作。
2. 用来保存删除的行。
3. 使数据保持一致的格式。