触发器是一种特殊的存储过程,在插入\删除\修改特定表中的数据时触发执行的数据库事件

作用：

1. 安全性，可以基于数据库的值使用户具有操作数据库某种权利
   1. 可以基于时间限制用户的操作，如不允许下班后或节假日修改数据
   2. 可以给予数据库中的数据限制用户的操作，如不允许股票的价格的升幅一次超过10%
2. 审计，可以跟踪用户对数据库的操作
   1. 审计用户操作数据库的语句
   2. 把用户对数据库的更新写入审计表
3. 实现复杂的数据完整性规则
   1. 实现非标准的数据完整性检查和约束，触发器可产生比规则更为复杂的限制，与规则不同，触发器可以引用列或数据库对象。如可回退任何企图吃进超过自己保证金的期货
   2. 提供可变的缺省值
4. 实现复杂的非标准的数据库相关完整性规则

创建触发器语法：

Create trigger trigger\_name

Trigger\_time

Trigger\_event on table\_name

For each row

Trigger\_stmt

语法解析：

Trigger\_name：标识触发器名称，用户自行制定；

Trigger\_time：标识触发时机，取值为before和after

Trigger\_event：标识触发事件，取值为insert、update、delete

Table\_name：标识建立触发器的表名，即在哪张表上建立触发器

Trigger\_stmt：触发器程序体，可以是依据sql语句，或者用begin和end包含的多条语句，每条语句必须用分号隔开

从上面语法可知，可以创建6种类型的触发器；而且不能同时在一个表上建立两个相同类型的触发器。

在mysql中，分号是语句结束的标识符，遇到分号表示该段语句已经结束，因此解释器遇到触发器程序体中的分号就会报错，因为没有找到和begin匹配的end。

这时需要用到delimiter命令(delimiter是定界符、分隔符的意思)，是一条命令，不需要语句结束标识，语法为：

Delimiter new\_delimter\_name

其中new\_delimter\_name是指自定义的一个或多个长度的符号，默认为分号( ; ),这里可以修改为其他符号，如 $ 、//等。在这之后的语句，以分号结束的语句解析器就不会有什么反应，只有遇到$ 、//才被认为是结束语句。

注意：使用完自定义的定界符之后应该要修改回分号( ; )

例子：

delimiter $

create TRIGGER tri\_stuupdate after INSERT

on student

for EACH ROW

BEGIN

DECLARE c int;

set c=(SELECT stucount FROM class where classid = new.classid);

UPDATE class set stucount = c+1 where classid = new.classid;

END$

delimiter ;

删除触发器：drop trigger 触发器名称;

查看现存的触发器

查看：Information\_schema.triggers表

Select \* from information\_schema.triggers：该语句用于查询information\_schema数据库中的triggers表；即可看到所有的触发器

或 show triggers 查看触发器