代码执行结构

在 MySQL 编程中，代码的执行结构有三种，分别为：

顺序结构；

分支结构；

循环结构。

顺序结构，自不必多说，在本文中，我们着重了解分支结构和循环结构。

分支结构

分支结构：事先准备多个代码块，通过判断条件是否满足，执行对应的代码。

在 MySQL 中，只有if分支结构，其基本语法为：

if 条件判断 then

-- 满足条件时，要执行的代码

else -- 可以没有 else 语句

-- 不满足条件时，要执行的代码

end if;

接下来，我们利用触发器和if分支，完成这样的需求：

在生成订单前，判断商品的库存是否满足，如果满足，则插入订单；否则，插入失败。

依次执行如下 SQL 语句，进行测试：

-- 查看商品表

select \* from goods;

-- 查看订单表

select \* from orders;

-- 修改语句结束符

delimiter $$

-- 创建触发器

create trigger before\_order before insert on orders for each row

begin

-- 判断商品库存是否满足订单

-- 通过商品表获取商品库存量

select inventory from goods where id = NEW.goods\_id into @inventory;

-- 比较库存

if @inventory < NEW.goods\_number then

-- 库存不够，由于触发器不能阻止事件发生，因此我们只能暴力报错

insert into XXX values(XXX);

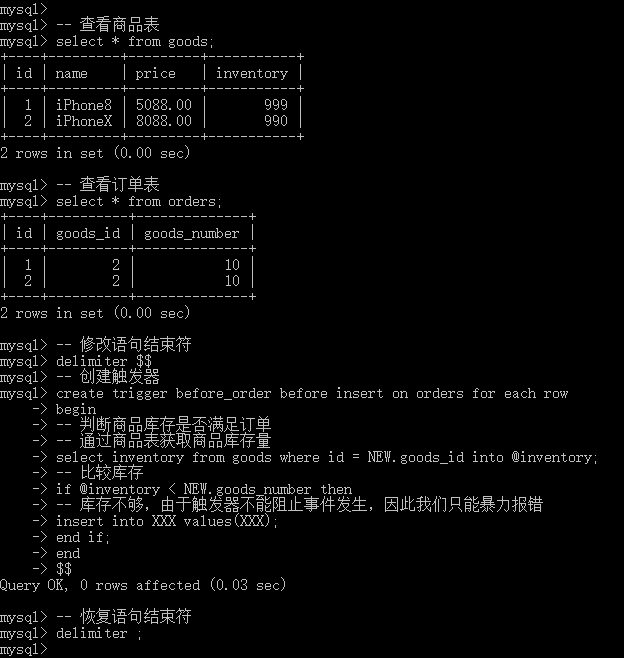
end if;

end

$$

-- 恢复语句结束符

delimiter ;

[](https://camo.githubusercontent.com/f6f2e3c04ef21952475a3bf0dac82ea8557129c8/687474703a2f2f696d672e626c6f672e6373646e2e6e65742f3230313830313033323230313430393939)

-- 插入订单

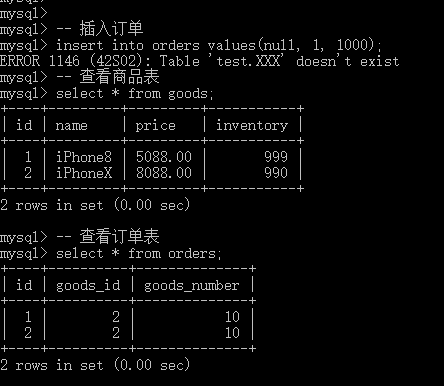
insert into orders values(null, 1, 1000);

-- 查看商品表

select \* from goods;

-- 查看订单表

select \* from orders;

[](https://camo.githubusercontent.com/6d223ed2992e6b76147f225d321113746566b46d/687474703a2f2f696d672e626c6f672e6373646e2e6e65742f3230313830313033323230323136383738)

-- 插入订单

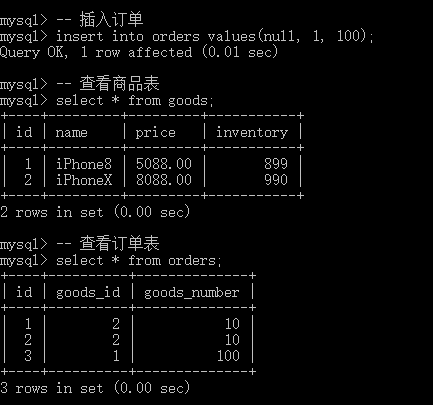
insert into orders values(null, 1, 100);

-- 查看商品表

select \* from goods;

-- 查看订单表

select \* from orders;

[](https://camo.githubusercontent.com/c9d7992621378d891bcf4c0f75f16cd193beb3a9/687474703a2f2f696d672e626c6f672e6373646e2e6e65742f3230313830313033323230343031363435)

如上图所示，虽然在报错的时候，没有给出友好的提示信息，但我们已经实现了该需求。

循环结构

循环结构：表示某段代码在指定条件下进行重复执行动作。

在 MySQL 中，没有for循环，仅有while循环、loop循环和repeat循环，呃，还有一种非标准的goto循环，在此我们仅介绍while循环，其基本语法为：

while 条件判断 do

-- 满足条件时要执行的代码

-- 变更循环条件

end while;

在使用循环结构的时候，我们经常需要对循环进行控制，即在循环结构内部进行判断和控制。虽然在 MySQL 中没有continue和break，但是有其替代关键字：

iterate：迭代，类似于continue，表示结束本次循环，不执行后续步骤，直接开始下一次循环；

leave：离开，类似于break，直接结束整个循环。

上述两个关键字的使用方法为，

基本语法：iterate/leave + 循环名称;

因此，在我们定义循环结构的时候，就需要进行略微的修改了，具体形式如下：

循环名称: while 条件判断 do

-- 满足条件时要执行的代码

-- 变更循环条件

iterate/leave 循环名称; -- 控制循环语句

end while;

由于触发器只能执行简单的一次触发动作，因此不适合演示循环结构。循环结构需要结合函数进行使用，所以在介绍函数的时候，我们再来体验循环结构的魅力。

温馨提示：符号[]括起来的内容，表示可选项；符号+，则表示连接的意思。