

```
CREATE TABLE trigger_pessoa (codigo int not null auto_increment,  
nome varchar(45) not null,  
email varchar(45) not null,  
telefone varchar(45) not null,  
PRIMARY KEY (codigo));
```

```
CREATE TABLE trigger_produto (codigo int not null auto_increment,  
nome varchar(45) not null,  
estoque int not null,  
preco float not null,  
PRIMARY KEY (codigo));
```

```
CREATE TABLE trigger_compra (codigo int not null auto_increment,  
codigo_produto int not null,  
codigo_pessoa int not null,  
quantidade int not null,  
valor_unitario float not null,  
valor_total float not null,  
PRIMARY KEY(codigo),  
FOREIGN KEY (codigo_produto) REFERENCES trigger_produto(codigo),  
FOREIGN KEY (codigo_pessoa) REFERENCES trigger_pessoa(codigo));
```

```
CREATE TABLE trigger_venda (codigo int not null auto_increment,  
codigo_produto int not null,  
codigo_pessoa int not null,  
quantidade int not null,  
valor_unitario float not null,  
valor_total float not null,  
PRIMARY KEY(codigo),  
FOREIGN KEY (codigo_produto) REFERENCES trigger_produto(codigo),  
FOREIGN KEY (codigo_pessoa) REFERENCES trigger_pessoa(codigo));
```

1) Crie uma trigger, que dispare antes de um evento de inserção na tabela pessoa, para verificar se o atributo email possui a letra @. Se não possuir cancele a inserção apresentando uma mensagem de erro.

```
delimiter $$  
drop trigger if exists verifica_arroba$$  
create trigger verifica_arroba before insert on trigger_pessoa for each  
row  
begin  
declare msg varchar(255);  
if (new.email not like '%@%') then  
set msg = "Email Invalido";
```

```

signal sqlstate '45000' set message_text = msg;
end if;
end$$
delimiter ;

```

✓ 1 linha inserida.  
Inserted row id: 2 (A consulta demorou 0.0019 segundos.)

```

INSERT INTO trigger_pessoa(nome, email, telefone) values ('helena', 'kds@shdkh', '8994739')

```

Mostrar Caixa do query

```

INSERT INTO trigger_pessoa(nome, email, telefone) values ('helena', 'kdsshdkh', '8994739')

```

**Mensagens do MySQL :**

#1644 - Email Invalido

Executa comando(s) SQL na base de dados info18\_helena:

```

1 INSERT INTO trigger_pessoa(nome, email, telefone) values ('helena', 'kdsshdkh', '8994739');
2
3

```

2) Faça uma trigger que atualize automaticamente o estoque da tabela produto quando determinado produto for comprado. Também gere automaticamente o valor da coluna “valor\_total” na tabela compra.

```

delimiter $$
drop trigger if exists atualiza_valor$$
create trigger atualiza_valor before insert on trigger_compra for each
row
begin
    declare estoque int;
    declare codigoo int;
    declare precoo float;
    declare aux int;
    select codigo, estoque, preco into codigoo, estoque, precoo from trigger_produto
where new.codigo_produto=codigo;
    set aux = estoque + new.quantidade;

```

```
update trigger_produto set estoque=aux where codigo = codigoo;
set new.valor_total = new.quantidade * new.valor_unitario;
```

```
end$$
delimiter ;
```

```
INSERT INTO trigger_compra(codigo_produto, codigo_pessoa, quantidade, valor_unitario,
valor_total) values (1, 1, 2, 4, 0);
INSERT INTO trigger_produto(nome, estoque, preco) values ('guarana', 20, 4);
INSERT INTO trigger_produto(nome, estoque, preco) values ('fanta', 40, 5);
INSERT INTO trigger_produto(nome, estoque, preco) values ('coca', 12, 7);
```

	codigo	codigo_produto	codigo_pessoa	quantidade	valor_unitario	valor_total
	3	1	1	2	4	8

3) Crie uma trigger que no momento de registro de uma venda, de baixa no estoque do produto vendido, considere como valor unitário do produto o preço armazenado na tabela produto acrescido de 30%, e também faça o calculo automático do valor total da venda. Garanta que a quantidade vendida esteja disponível em estoque, se não tiver retorne uma mensagem e cancela a venda.

```
delimiter $$
drop trigger if exists registro_venda$$
create trigger registro_venda before insert on trigger_venda for each
row
begin
```

```

declare msg varchar(255);
declare quantidadee int;
declare codigoo int;
declare precoo float;
declare porcentagem float;
declare total float;
declare aux int;
select estoque, codigo, preco into quantidadee, codigoo, precoo from trigger_produto
where new.codigo_produto=codigo;
set aux = quantidadee - new.quantidade;
update trigger_produto set estoque=aux where codigo = codigoo;
set porcentagem = (precoo*30)/100;
set new.valor_unitario = porcentagem + precoo;
set new.valor_total = new.valor_unitario * new.quantidade;
if (aux<0) then
    set msg = "Quantidade Invalida";
    signal sqlstate '45000' set message_text = msg;
end if;

end$$
delimiter ;

```

#### Erro

Comando SQL:

```
INSERT INTO trigger_venda(codigo_produto, codigo_pessoa, quantidade, valor_unitario, valor_total) values (3, 1, 50, 3, 0)
```

Mensagens do MySQL: ⓘ

#1644 - Quantidade Invalida

Executa comando(s) SQL na base de dados info18\_helena: ⓘ

```
1 INSERT INTO trigger_venda(codigo_produto, codigo_pessoa, quantidade, valor_unitario, valor_total) values (3, 1, 50, 3, 0);
2
```

Clear

1 linha inserida.  
Inserted row id: 4 (A consulta demorou 0.0210 segundos.)

```
INSERT INTO trigger_venda(codigo_produto, codigo_pessoa, quantidade, valor_unitario, valor_total) values (3, 1, 5, 3, 0)
```

Mostrar Caixa do query

A mostrar registros de 0 - 0 (1 total, A consulta demorou 0.0002 segundos.)

```
SELECT * FROM `trigger_venda`
```

Número de registros: 25 Filtar registros: Pesquisar esta tabela

+ Opções

	codigo	codigo_produto	codigo_pessoa	quantidade	valor_unitario	valor_total
<input type="checkbox"/> Editar  Copiar  Apagar	4	3	1	5	6.5	32.5

↑ ☐ Todos Com os seleccionados: Muda Apagar Exportar

4) Crie uma rotina que incremente o preço de todos os produtos em 1% a cada minuto durante 5 minutos a partir de um momento específico.

delimiter \$\$

drop event if exists nome\_evento \$\$

create event nome\_evento

on schedule every 1 minute starts now( ) ends now() + interval 5 minute

on completion preserve

enable

do

begin

update trigger\_produto set preco=(preco\*0.01)+preco;

end \$\$

delimiter ;

## ESTOQUES

✓ A mostrar registros de 0 - 2 (3 total, A consulta demorou 0.0002 segundos.)




```
SELECT * FROM `trigger_produto`
```

Número de registros: 25 ▼

Filtrar registros:

Ordenar por chave:  ▼

+ Opções

				codigo	nome	estoque	preco
<input type="checkbox"/>		Edita		Copiar		Apagar	1 coca 14 7
<input type="checkbox"/>		Edita		Copiar		Apagar	2 guarana 20 4
<input type="checkbox"/>		Edita		Copiar		Apagar	3 fanta 25 5