

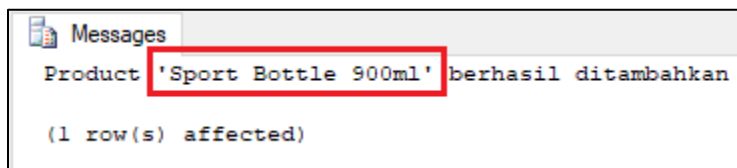
Trigger

Untuk dapat mencapai kompetensi mata kuliah, mahasiswa harus mencapai kompetensi-kompetensi dasar sebagai berikut:

1. Memahami dasar-dasar pembuatan trigger seperti format penulisan.
2. Memahami penggunaan tabel virtual inserted dan deleted.
3. Memahami cara melakukan pengtesan terhadap trigger.

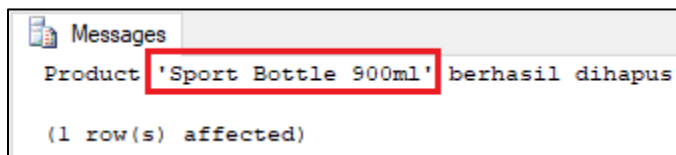
1. Buatlah sebuah trigger dengan nama *tg_add_product*, trigger ini otomatis dijalankan pada saat proses *penambahan* data product baru. Jika perintah berikut dijalankan maka akan menampilkan pesan seperti pada gambar.

```
insert into products (product_id, name, stok, purchasing_price,
selling_price, category_id, supplier_id)
values ('CG-006-IT000040', 'Sport Bottle 900ml', 15, 75000, 780000, 'CG-
006', 'SUP-000017')
```



2. Buatlah sebuah trigger dengan nama *tg_delete_product*, trigger ini otomatis dijalankan pada saat proses *penghapusan* data product. Jika perintah berikut dijalankan maka akan menampilkan pesan seperti pada gambar.

```
delete from products where product_id = 'CG-006-IT000040'
```



3. Buatlah sebuah trigger dengan nama *tg_kurangi_stok*, trigger ini otomatis dijalankan pada saat proses penjualan product. Sebagai contoh kita akan menjual data product dengan nama *Roberto Cavalli Sunglass*, berikut data product sebelum proses penjualan.

Results		Messages	
	product_id	name	stok
1	CG-001-IT000002	Roberto Cavalli Sunglass	100

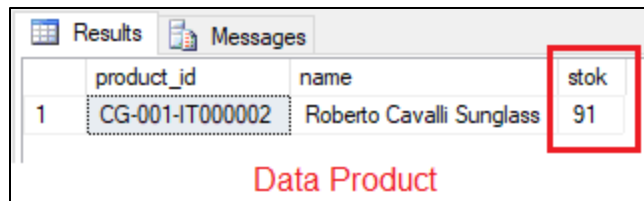
Setelah menjalankan perintah berikut, stok barang seharusnya menjadi **91**.

```
insert into orderitems (order_id, product_id, amount, purchasing_price,
selling_price)
values ('20160305-00001', 'CG-001-IT000002', 9, 160000, 165000)
```



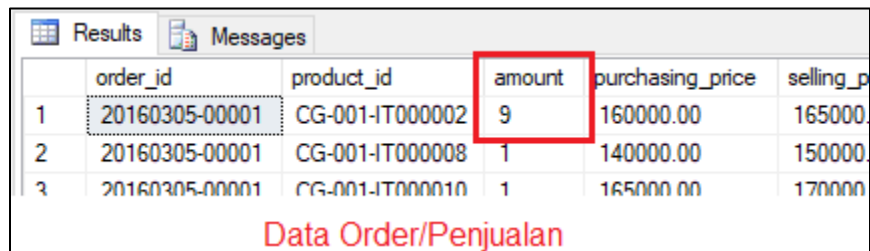
	product_id	name	stok
1	CG-001-IT000002	Roberto Cavalli Sunglass	91

4. Buatlah sebuah trigger dengan nama *tg_update_stok*, trigger ini otomatis dijalankan pada saat proses *update* data penjualan product. Sebagai contoh ada kesalahan input jumlah product yang di jual, sehingga kita akan mengupdate jumlah (amount) penjualan data product dengan nama *Roberto Cavalli Sunglass*, yang awalnya berjumlah **9** menjadi **15**. Berikut data product dan penjualan sebelum proses update.



	product_id	name	stok
1	CG-001-IT000002	Roberto Cavalli Sunglass	91

Data Product

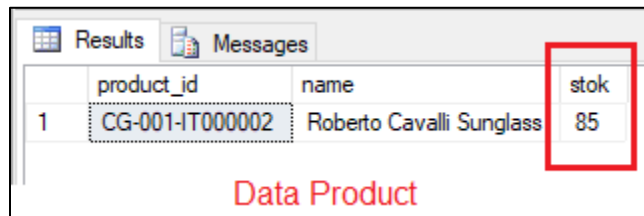


	order_id	product_id	amount	purchasing_price	selling_p
1	20160305-00001	CG-001-IT000002	9	160000.00	165000
2	20160305-00001	CG-001-IT000008	1	140000.00	150000
3	20160305-00001	CG-001-IT000010	1	165000.00	170000

Data Order/Penjualan

Setelah menjalankan perintah berikut, stok product seharusnya menjadi **85**.

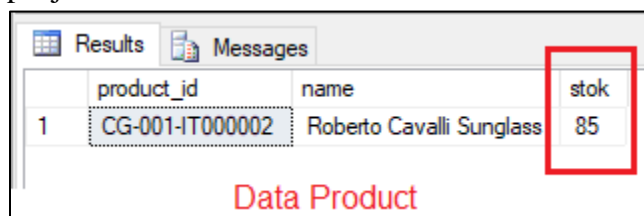
```
update orderitems set amount = 15  
where order_id = '20160305-00001' AND product_id = 'CG-001-IT000002'
```



	product_id	name	stok
1	CG-001-IT000002	Roberto Cavalli Sunglass	85

Data Product

5. Buatlah sebuah trigger dengan nama *tg_tambah_stok*, trigger ini otomatis dijalankan pada saat proses *pembatalan* penjualan product. Sebagai contoh kita akan membatalkan penjualan data product dengan nama *Roberto Cavalli Sunglass*, berikut data product sebelum proses pembatalan penjualan.

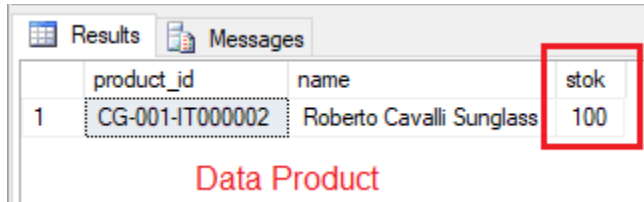


	product_id	name	stok
1	CG-001-IT000002	Roberto Cavalli Sunglass	85

Data Product

Setelah menjalankan perintah berikut, stok product seharusnya kembali menjadi **100**.

```
delete from orderitems  
where order_id = '20160305-00001' AND product_id = 'CG-001-IT000002'
```



	product_id	name	stok
1	CG-001-IT000002	Roberto Cavalli Sunglass	100

Data Product

Kamarudin, M.Kom
<http://coding4ever.net/>
<https://github.com/rudi-krsoftware/open-retail>