

PMI 1413  
4(2/2)



# Trigger

trigger merupakan kode prosedural yang akan secara otomatis dieksekusi akibat dipicu oleh sebuah event / kejadian pada suatu Tabel.

Trigger dapat difungsikan :

- Untuk mencegah akses terhadap data yang spesifik,
- Sebagaisebagai logging atau melakukan audit perubahan data.

# Trigger

Terdapat tiga kejadian (event) umum yang dapat menyebabkan trigger dipicu :

- INSERT, berarti trigger dijalankan ketika terdapat sejumlah data baru yang diinput
- UPDATE, berarti trigger dijalankan ketika terdapat data / record yang diubah
- DELETE, berarti trigger dijalankan ketika terdapat data / record yang dihapus

# Manfaat Trigger

- Trigger merupakan salah satu mekanisme untuk menjalankan aturan bisnis dan menjaga integritas data dalam sebuah basis data. Mekanisme yang lainnya adalah penggunaan CONSTRAINT
- Trigger dapat mengevaluasi status dari sebuah tabel sebelum dan sesudah terjadi perubahan data dan melakukan aksinya sesuai perbedaan yang terjadi
- Trigger dapat melakukan perubahan *cascade* melalui relasi tabel pada database, akan tetapi perubahan ini akan lebih efisien jika dieksekusi menggunakan *referential integrity constraints*.
- Trigger dapat melakukan pencegahan akses atas data yang kompleks dibanding dengan menggunakan CHECK CONSTRAINT
- Tidak seperti pada Check Constraint, Trigger dapat mereferensikan kolom / field dalam tabel yang berbeda.

# Sintak Perintah Trigger

- Membuat Trigger

```
CREATE TRIGGER {Nama_Trigger}  
ON {Nama_Table}  
FOR {DELETE | INSERT | UPDATE}  
AS  
{SQL Statement}
```

# Sintak Perintah Trigger

- Menghapus Trigger

**DROP TRIGGER** *{Nama\_Trigger}*

# Sintak Perintah Trigger

- Mengubah perintah dalam Trigger

```
ALTER TRIGGER {Nama_Trigger}  
ON {Nama_Table}  
FOR {DELETE | INSERT | UPDATE}  
AS  
{SQL Statement}
```

# Sintak Perintah Trigger

- Mengubah nama Trigger

```
sp_rename 'Old_Nama_Trigger', 'New_Nama_Trigger';
```



# Sintak Perintah Trigger

- Mendapat informasi tentang trigger

**sp\_helptext** {*Nama\_Trigger*}

**sp\_help** {*Nama\_Trigger*}

**sp\_columns** {*Nama\_Trigger*}

# Sintak Perintah Trigger

- mendapatkan daftar trigger yang telah dibuat

```
SELECT nama_trigger, data_trigger  
FROM sys.triggers  
WHERE type = 'TR';  
GO
```