



## 08221 Veri Tabanı II



Elbistan Meslek Yüksek Okulu  
2015 – 2016 GÜZ Yarıyılı

## Tetikleyici :

- ⦿ Tetikleyici (trigger), SQL Server üzerinde herhangi bir işlem gerçekleştiğinde başka bir işlemin tetiklenmesi için kullanılır.
- ⦿ Tetiklenecek işlem tamamen bizim belirlediğimiz çerçevede başka bir kayıt eklemek, rapor almak veya genelde kontrol mekanizmaları oluşturmak için kullanılabilir.
- ⦿ Tetikleyici, bir çeşit özel Stored Procedure'dür.
- ⦿ Bir kaydın eklenmesi (INSERT), silinmesi (DELETE) veya değiştirilmesi (UPDATE) durumlarında otomatik olarak devreye girecektir.

# 08221 Veri Tabanı II



## Tetikleyici :

- ⦿ Tetikleyiciler, Query ekranında T-SQL kodu ile veya Management Studio ile oluşturulup yönetilebilirler.
- ⦿ Tetikleyiciler, görünümler (View) ve tablolar üzerinde tanımlanabilir.



## Tetikleyici :

- ◉ SQL Server 2008, DDL (Data Defination Language) tetikleyicileri adında yeni bir kavramı destekler.
- ◉ DDL tetikleyiciler, olay gerçekleştikten sonra değil de CREATE, ALTER ve DROP gibi DDL ifadeleri için tanımlanırlar.

# 08221 Veri Tabanı II



## Tetikleyici Ne Zaman Kullanılır?

- ◉ Değişiklikleri takip etmek,
- ◉ Birincil anahtar üretmek,
- ◉ Karmaşık iş kurallarını gerçekleştirmek,
- ◉ E-posta atmak gibi olayları otomatik olarak yapmak,
- ◉ Standart hata mesajlarının dışında bir hata mesajı elde etmek,
- ◉ Veritabanı erişimlerini takip edebilmek,
- ◉ Nesnede meydana gelebilecek değişiklikleri takip ve engellemektir.

# 08221 Veri Tabanı II



## Tetikleyiciyi Ateşleyen Olaylar ;

- Bir işlemin gerçekleşmesini veya gerçekleşmeye başlamasını tetikleyici ateşleyebilir.
- Tetikleyiciyi ateşleyen bir işlem ile karşılaşıldığında tetikleyici ile işlem bir blokta ele alınır.
- Bu blok, RAM'de bulunan geçici hafıza bloğudur.
- Tetikleyiciyi çağıran işlemi onaylamak anlamında hiçbir işlem yapmayabilir veya işlem başarılı olmadığında işlemi geçersiz kılabilir.

## Tetikleyicinin Çalışması ;

- ⦿ Tetikleyici çalıştığı zaman Inserted ve Deleted adı verilen sahte tabloları kullanır.
- ⦿ Bu tablolar tetikleyicinin ateşlendiği tabloya eşdeğer alanlara sahiptir. Bunlar da mantıksal olarak RAM'de bulunur.
- ⦿ Asıl tabloya bir kayıt eklendiğinde ve tetikleyici ateşlendiğinde bu kayıt Inserted tablosuna da eklenir.

## Tetikleyicinin Çalışması ;

- Tablodan bir kayıt silindiğinde silinen kayıt Deleted sahte tablosuna da eklenir.
- Update işlemi ise önce silme (Delete) ve ardından bir kayıt ekleme (Insert) olarak ele alınır.
- Bir kayıt güncellendiğinde asıl kayıt Deleted sahte tablosuna, değişen kayıt da Inserted sahte tablosuna yazılır.



## Tetikleyicinin Çalışması ;

- ◉ Inserted ve Deleted sahte tabloları RAM'deki hafıza bölgesinde elde edilir.
- ◉ Bundan dolayı, TRUNCATE TABLE gibi yazılmayan tablolara yansımayan değişiklikler tetikleyici tarafından yakalanamaz.
- ◉ Tetikleyiciler dışarıdan parametre almazlar. Ancak, sahte tablolar sayesinde son işlemde etkilenmekte olan kayıtlar tespit edilebilir.

# 08221 Veri Tabanı II



## Tetikleyici Türleri

SQL Server'da iki farklı tür tetikleyici vardır.

- ⦿ After ve
- ⦿ Instead Of tetikleyicileridir.

## After Tetikleyiciler

- ⦿ After tetikleyicileri, kendiyle ilişkili işlem gerçekleştikten hemen sonra ateşlenir.
- ⦿ Veritabanındaki temel işlemler için (ekleme, silme ve güncelleme) After tetikleyicileri tanımlanabilir.

## After Tetikleyiciler

- Örneğin, yeni bir personel kaydı silindiğinde farklı tablolarda o personele ait diğer bilgilerin silinmesi After Delete tetikleyicisi ile sağlanabilir.
- Birden fazla tetikleyici, bir iş için tanımlanabildiği gibi, bir tetikleyici de birden fazla iş için tanımlanabilir.
- After tetikleyicileri, sadece tablolar için tanımlanabilir.

## Instead Of Tetikleyiciler

- ◉ Instead of tetikleyicileri, belirlenen işlem gerçekleşirken devreye girer ve kendi içinde tanımlanan komutları icra etmeye başlar. Yani, belirlenen işlemin yerine geçer.
- ◉ Instead of tetikleyicileri işlemlerin arasına girebildiğinden kontrol amaçlı kullanılabilirler.
- ◉ Örneğin, ekleme işlemi için tanımlanan bir tetikleyici, ekleme işleminden hemen önce araya girerek uygun koşulların sağlanıp sağlanmadığını kontrol edebilir.

## Instead Of Tetikleyiciler

- Tıpkı After tetikleyicileri gibi temel veritabanı işlemleri için Instead Of tetikleyicileri tanımlanabilir.
- Fakat After tetikleyicileri sadece tablolar için tanımlanabilirken Instead Of tetikleyicileri hem tablolar için hem de görünümler (views) için tanımlanabilirler.



## Tetikleyici Oluşturmak

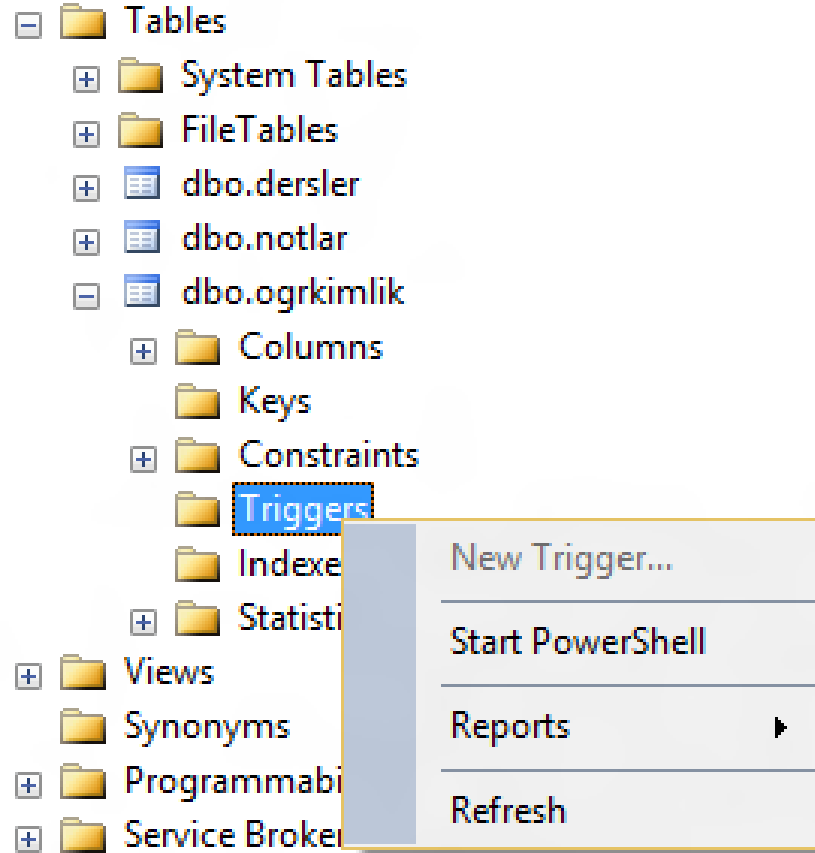
- ⦿ CREATE TRIGGER'i kullanabilmek için sysadmin, db\_owner veya db\_ddladmin rolüne sahip olmak gerekir.
- ⦿ Bir tetikleyici (trigger) başka bir tabloya erişecekse bu tablo için de tetikleyici oluşturan kullanıcının erişim izni veya güncelleme izni olması gerekir.

# 08221 Veri Tabanı II



## Tetikleyici Oluşturmak

- Tetikleyiciyi oluşturmak için CREATE TRIGGER ifadesi veya Managemet Studio'da tablo altında bulunan **Triggers** klasörü üzerinde sağ tıklayarak açılan menüden **New Trigger** komutu kullanılır.





# 08221 Veri Tabanı II



## Tetikleyici Oluşturmak

### Yazılışı

```
CREATE TRIGGER tetikleyici_adi  
ON tablo_adi  
FOR veya AFTER veya INSTEAD OF (INSERT veya  
UPDATE veya DELETE)  
AS Sql ifadeleri
```

## INSERT Tetikleyicisi

- Bir tablo üzerinde yeni kayıtlar girildikten sonra devreye giren tetikleyicidir.
- Insert tetikleyici devreye girdikten sonra Inserted tablosunda yeni eklenen kayıtların bir kopyası tutulur.
- Inserted tablosu, asıl tablonun yapısal bir kopyası olup tetikleyici sonlanana kadar saklanır. Eğer, bir tabloya birden fazla kayıt girildiyse tetikleyici her kayıt için ayrı ayrı devreye girmez.
- Tek seferde işlemi gerçekleştirir. Çünkü SQL Server, satır bazında tetikleyici desteği sağlamamaktadır.

# 08221 Veri Tabanı II



## Örnek

- “notlar” tablosunda notları “geçti/kaldı” otomatik olarak yazacak bir tetikleyici yazalım.

	Kitap_Id	Kitap_Yayinevi	Kitap_ISBNNo	Kitap_BasimYili
	123	ABC Yayinevi	1234567890	2006
	128	DEF Yayıncılık	5678901234	2007
	150	XYZ Matbaası	3456789012	2007
	200	ABC Yayinevi	1234567123	2006
*	NULL	NULL	NULL	NULL

# 08221 Veri Tabanı II



Bunun için bir tetikleyici oluşturalım.

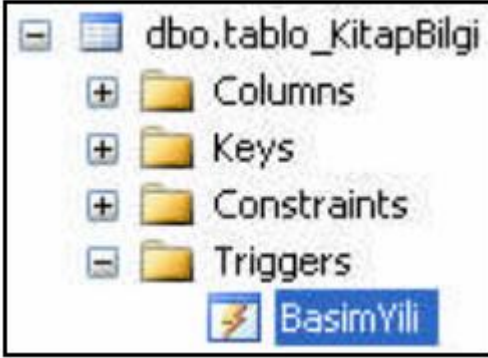
```
CREATE TRIGGER BasimYili
ON tablo_KitapBilgi
AFTER INSERT
AS
UPDATE tablo_KitapBilgi Set Kitap_BasimYili=2007
WHERE tablo_KitapBilgi.Kitap_Id=(SELECT Kitap_Id FROM INSERTED)
```

tablo\_KitapBilgi” tablosuna yeni bir kayıt girildiğinde ve kitap basım yılı boş geçildiğinde kitap basım yılı olarak 2007 değeri otomatik olarak yazılacaktır.

Tetikleyiciyi yazdıktan sonra Execute ediniz. Artık tetikleyici, Tables klasörü altında oluşturulacak ve tetiklenmeye hazır hâle gelecektir.

# 08221 Veri Tabanı II

## Tetikleyicinin konumu



“tablo\_KitapBilgi” tablosunu açarak yeni bir kayıt giriniz ve basım yılı alanına değer girmeden Enter tuşuna basınız.

# 08221 Veri Tabanı II



## Tetikleyicinin çalıştırılması

Table - dbo.tablo_KitapBilgi				
	Kitap_Id	Kitap_Yayinevi	Kitap_ISBNNo	Kitap_BasimYili
	123	ABC Yayınevi	1234567890	2006
	128	DEF Yayıncılık	5678901234	2007
	150	XYZ Matbaası	3456789012	2007
	200	ABC Yayınevi	1234567123	2006
▶ !	205	DEF Yayıncılık	9876543210	NULL
*	NULL	NULL	NULL	NULL

Basım yılı alanını boş geçip Enter tuşuna bastığınızda tetikleyici devreye girecektir.

Eklenen kaydın son durumunu görmek için tabloyu Refresh etmeniz gerekir. Bunun için, Execute SQL simgesini tıklayarak ya da kapatıp açarak tabloyu yenileyiniz. Basım yılı alanına otomatik olarak 2007 değerinin yazıldığını göreceksiniz.



## DELETE Tetikleyicisi

- Tablodan bir kayıt silindiğinde otomatik olarak yapılması istenen işlemler için DELETE tetikleyicisi kullanılır.
- DELETE tetikleyicisi çalıştıktan sonra silinen kayıt Deleted sahte tablosuna kaydedilir.
- Deleted tablosunun Inserted tablosundan farkı, asıl tablodan silinen kayıt artık Deleted tablosunda yer almaktadır.

## DELETE Tetikleyicisi

### Örnek

- “tablo\_Kitap” tablosunda “Kitap\_Id”si 150’den büyük olan kitapların silinmesini engelleyecek bir DELETE tetikleyicisi yazdığınızı varsayınız.

Kitap_Id	Kitap_Ad	Kitap_Yazar	Kitap_Sayfa
123	Visual Basic.NET	Ali CAN	550
128	C#.Net	Veli BAL	700
150	Access XP	Osman GÜR	450
200	VB 6.0 Uygulam...	Sena PINAR	330
250	C++ Uygulamaları	Yasemin ZEKİ	270
251	Delphi 7.0	Yağmur AK	310
NULL	NULL	NULL	NULL



## DELETE Tetikleyicisi

- “tablo\_Kitap” tablosu üzerinde bir tetikleyici oluşturulmaktadır. Tablodaki toplam kayıt sayısının 0’dan büyük olması durumu göz önüne alınmaktadır. Bu halde, “Kitap\_Id”si 150’ye eşit ve büyük id numaralarının silinmesi engellenecektir.

```
CREATE TRIGGER Sil
ON tablo_Kitap
FOR DELETE AS
IF (SELECT COUNT(*) FROM deleted
WHERE deleted.Kitap_Id >= 150) > 0
BEGIN
    PRINT 'Bu kitap id lerinden birini silemezsiniz.'
    PRINT 'Silme işlemi'
    PRINT 'DELETE tetikleyici tarafından engellenmiştir.'
    PRINT 'Transaction geri alınacaktır'
    ROLLBACK TRANSACTION
END
```

## DELETE Tetikleyicisi

- Tetikleyiciyi Execute ettiğinizde “tablo\_Kitap” altında “Sil” adında bir tetikleyici oluşturulacaktır.
- Yeni bir sorgu sayfası açıp tablodan kitap id’si 150’den büyük olanları silmek istediğinizde uyarı mesajı ekrana gelecektir.

```
DELETE FROM tablo_Kitap
WHERE Kitap_Id>150
```

Messages

Bu satış noktalarından birini silemezsiniz.  
Bazılarının 20 den fazla satışı mevcuttur!  
Transaction geri alınacaktır

## UPDATE Tetikleyicisi

- Tablo üzerindeki kayıt ya da kayıtlarda güncelleme olduğunda devreye girecek olan tetikleyicidir. INSERT ve DELETE tetikleyicilerden biraz farklıdır.
- Farkı ise UPDATE tetikleyici devreye girdiğinde Inserted sahte tablosu asıl tablodaki kayıtlardan, düzenlenmiş kayıtların kopyasını, Deleted sahte tablosu ise kayıtların düzenleme işleminden önceki hâllerini tutar.

## UPDATE Tetikleyicisi

### Örnek

- “tablo\_Kitap” tablosunda bulunan kitap fiyatlarından kaçının fiyatının değiştiğini bulacak bir UPDATE tetikleyici yazdığınızı varsayınız.
- Bunun için “tablo\_Kitap” tablosuna kitap fiyatlarının tutulacağı “Kitap\_Fiyat” adında ve int tipinde bir sütun oluşturarak sütun içerisine kitap fiyat bilgilerini giriniz.

## UPDATE Tetikleyicisi

```
CREATE TRIGGER indirim
ON tablo_Kitap
FOR UPDATE
AS
raiserror ('%d kayıt üzerinde değişiklik yapılmıştır',0,1,@@rowcount)
RETURN
```

Bu tetikleyicide kullanılan RAISERROR fonksiyonu, SQL Server'da hata meydana geldiğinde hata mesajı vermek için kullanılır. Burada ise sadece mesaj verme işlemi için kullanılmıştır.

@@rowcount ise SQL Serverde en son yapılan işlemde kaç kaydın etkilendiğini verir.

## UPDATE Tetikleyicisi

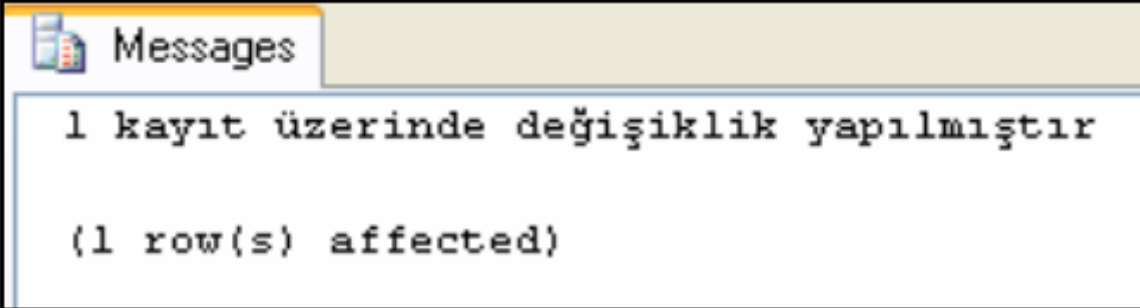
Bu tetikleyicinin otomatik olarak devreye girebilmesi için bu tetikleyiciyi ateşleyen bir SQL kodunun olması gerekir.

Örnek olması açısından “tablo\_Kitap” tablosunda “Kitap\_Id” numarası 250’den büyük kayıtların fiyatlarında 5 YTL’lik bir indirim yapılması istensin.

```
UPDATE tablo_Kitap  
SET Kitap_Fiyat=Kitap_Fiyat-5  
WHERE Kitap_Id>250
```

## UPDATE Tetikleyicisi

Execute işleminden sonra Messages penceresinde sorgu sonucu kaç kaydın bu işlemde etkilendiği görüntülenecektir.

A screenshot of the SQL Server Messages window. The window title is 'Messages'. The text inside the window reads: '1 kayıt üzerinde değişiklik yapılmıştır' followed by '(1 row(s) affected)' on the next line.

```
Messages  
1 kayıt üzerinde değişiklik yapılmıştır  
(1 row(s) affected)
```

## INSTEAD OF Tetikleyicisi

- ◉ INSTEAD OF tetikleyicisi, belirlenen işlem gerçekleşirken devreye girer ve kendi içinde tanımlanan komutları icra etmeye başlar. Yani, belirlenen işlemin yerine geçer.
- ◉ Bu tetikleyiciden önceki tetikleyiciler, veriler uygun değilse ROLLBACK ile işlemleri geri alırlar.
- ◉ INSTEAD OF, işlem gerçekleşirken verilerin uygunluğunu denetleyecektir.



## INSTEAD OF Tetikleyicisi

### Örnek

- Uygulaması yapılacak olan bu tabloda, kayıtlı olan müşteri silinmeye çalışıldığı anda silme işleminin gerçekleştiği tarih ve saat otomatik olarak “SilmeGirisimi” adlı alana yazılacaktır.
- “Musteriler” adında bir tablo oluşturarak tabloda kullanılacak alanları T-SQL kodu ile tanımlayalım.

## INSTEAD OF Tetikleyicisi

### Örnek

```
CREATE TABLE Musteriler
(
    Musteri_No int IDENTITY (1, 1) NOT NULL ,
    Ad_Soyad VARCHAR(30),
    SilmeGirisimi DATETIME
)
```

EXECUTE ederek tablonun oluşumunu tamamlayınız.

Oluşturulan tabloya kayıt girişini de yine T-SQL kodları ile gerçekleştiriniz.

## INSTEAD OF Tetikleyicisi

### Örnek

```
INSERT INTO Musteriler (Ad_Soyad) VALUES ('Kemal ORTA')
INSERT INTO Musteriler (Ad_Soyad) VALUES ('Orkun TAN')
INSERT INTO Musteriler (Ad_Soyad) VALUES ('Naciye BÜYÜK')
INSERT INTO Musteriler (Ad_Soyad) VALUES ('Naci BÜYÜK')
INSERT INTO Musteriler (Ad_Soyad) VALUES ('Metin KÜÇÜK')
```

Sorguyu çalıştırarak kayıtların tabloya eklenmesini sağlayınız.

## INSTEAD OF Tetikleyicisi

### Örnek

“Musteri\_Silme” adında bir tetikleyici oluşturarak bu tetikleyicinin tablodan kayıt silinmesinin başlaması durumunda sistem tarih ve saatini “SilmeGirisimi” sütununa yazmasını ve sorguyu EXECUTE ederek tetikleyicinin oluşmasını sağlayınız.

```
CREATE TRIGGER Musteri_Silme
ON Musteriler
INSTEAD OF DELETE
AS
UPDATE Musteriler SET SilmeGirisimi = GETDATE()
WHERE Musteri_No IN (SELECT Musteri_No FROM DELETED)
```

## Tetikleyici Yönetmek

### Tetikleyiciyi Düzenlemek (ALTER)

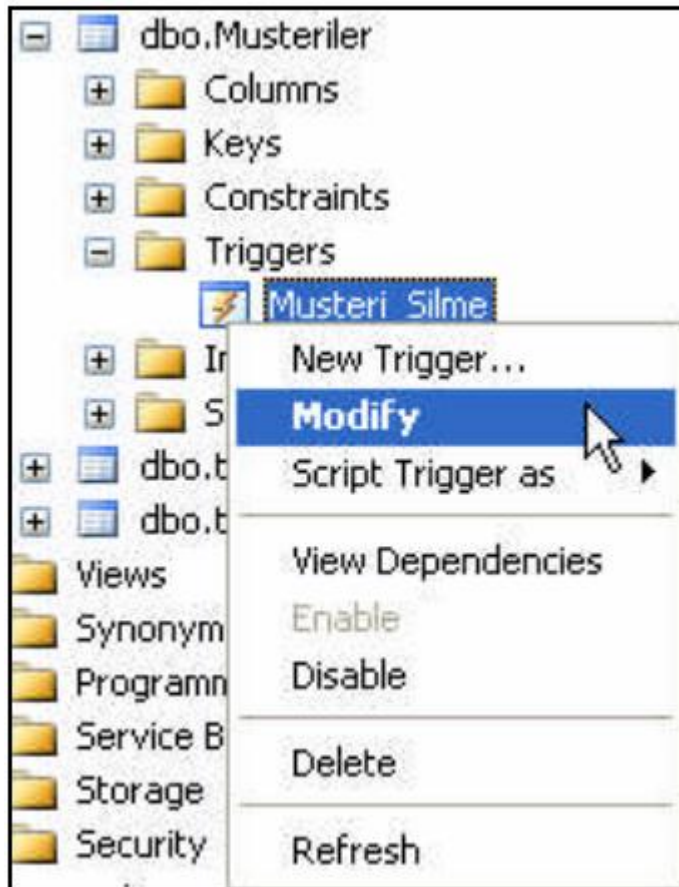
- ⦿ CREATE ile oluşturulmuş bir tetikleyiciyi daha sonra üzerinde değişiklikler yapmak üzere tekrar açmak isteyebilirsiniz.
- ⦿ Bunun için, tablounuzda oluşturulmuş bir tetikleyici üzerindeyken sağ tıklayarak açılan menüden Modify komutunu vermeniz gerekir.

# 08221 Veri Tabanı II



## Tetikleyici Yönetmek

### Tetikleyiciyi Düzenlemek (ALTER)



## Tetikleyici Yönetmek

### Tetikleyiciyi Düzenlemek (ALTER)

- ◉ Modify komutunu verdiğinizde daha önceden oluşturduğunuz tetikleyici açılacaktır.

```
set ANSI_NULLS ON
set QUOTED_IDENTIFIER ON
go

ALTER TRIGGER [Musteri_Silme]
ON [dbo].[Musteriler]
INSTEAD OF DELETE
AS
UPDATE Musteriler SET SilmeGirisimi = GETDATE()
WHERE Musteri_No IN (SELECT Musteri_No FROM DELETED)
```

## Tetikleyici Yönetmek

### Tetikleyiciyi Düzenlemek (ALTER)

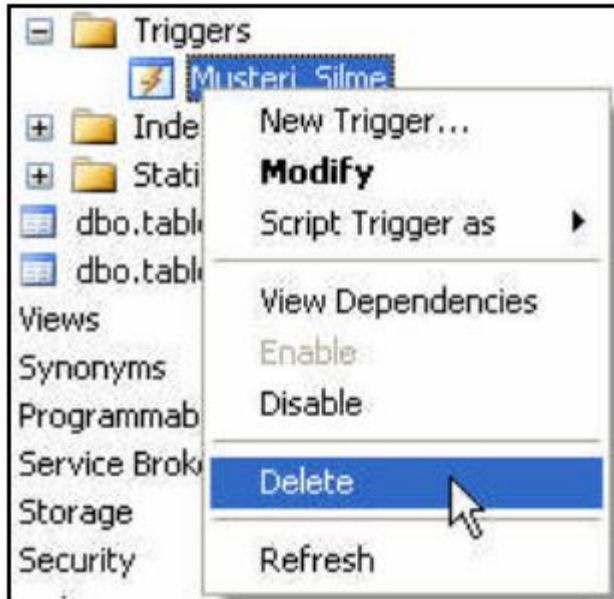
- ◉ Tetikleyici oluşturulurken CREATE komutu ile oluşturulmuştu. Modify (düzenle) komutunu verdiğinizde ise tetikleyici, ALTER komutu ile açılacaktır.
- ◉ Daha önceden oluşturulmuş tetikleyiciyi düzenlerken yeni bir sorgu sayfası açıp Modify komutunu kullanmadan da ALTER komutu ile tetikleyiciyi yeniden yazarak düzenleyebilirsiniz. Ancak bu önerilmez. Sonraki versiyolarda “DESIGN” ifadesi ile bu komut kullanılabilir.



## Tetikleyici Yönetmek

### Tetikleyiciyi Silmek (DROP)

- Oluşturulan bir tetikleyiciyi DROP komutuyla ya da tetikleyici üzerinde sağ tıklayarak açılan menüden Delete komutunu vererek silebilirsiniz.



T-SQL kodu olarak,

DROP TRIGGER Musteri\_Silme  
şeklinde yazabilirsiniz.



## Tetikleyici Yönetmek

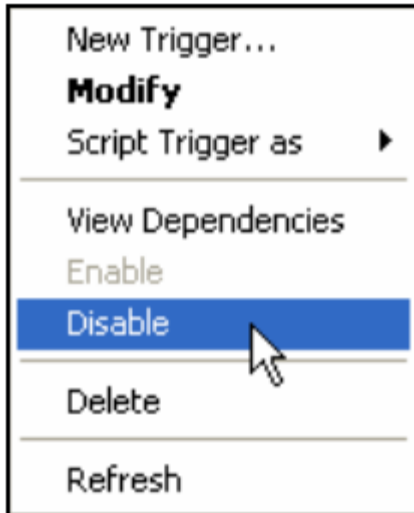
### Tetikleyiciyi Aktif ya da Pasif Yapma

- Oluşturulan bir tetikleyici, her zaman aktif durumdadır. Tetikleyiciyle ilgili bir durum oluştuğunda otomatik olarak devreye girer.
- İstenildiğinde oluşturulan bu tetikleyiciler pasif hâle getirilebilir.

## Tetikleyici Yönetmek

### Tetikleyiciyi Aktif ya da Pasif Yapma

- Tetikleyici üzerinde sağ tıklayarak açılan menüde Disable komutunu verdiğinizde tetikleyici pasif hâle gelir.



Tetikleyiciyi pasif hâle getirmenin bir diğer yolu da kod yazmaktır.

```
ALTER TABLE tablo_adi  
DISABLE TRIGGER Musteri_Silme
```

Tetikleyiciyi tekrar aktif hâle getirebilmek için menüden Enable komutunu veya `ENABLE TRIGGER` komutunu kullanmanız gerekir.

# 08221 Veri Tabanı II



HAFTAYA : Import-Export