

**T.C.
ONDOKUZ MAYIS ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**



Veri Tabanı Yönetimi Laboratuvarı

Föy 7

15060007 Kübra Nur Saruhan

1-tOgrenci: ogrenciID (Primary Key), ad, soyad, dogumtarih, adres, telefon, bolumID, vizenotu, finalnotu bilgilerini içermektedir. Oluşturduğunuz öğrenci tablosu üzerine bir silme tetikleyicisi yazınız. Bu tablodan bir kayıt silindiğinde togrenciYedek isimli ve aynı özelliklere sahip başka bir tabloya silinen kaydın aynısının eklenmesini sağlayınız

Vtys7hafta isminde veri tabanı oluşturuldu. Soruda istenilen tOgrenci tablosu ddl ile oluşturuldu.

```
SQLQuery2.sql - LA...ELKG\saruhan (56))* X SQLQuery1.sql - LA...ELKG\saruhan (56)* X
```

```
use vtys7hafta;  
  
CREATE TABLE tOgrenci (  
    ogrenciID INT PRIMARY KEY IDENTITY(1,1) NOT NULL,  
    ad NVARCHAR(10),  
    soyad NVARCHAR(10),  
    dogumTarihi DATE,  
    adres NVARCHAR(15),  
    telefon NVARCHAR(11),  
    bolumID INT,  
    vizeNotu INT,  
    finalNotu INT  
);
```

100 %

Messages

Commands completed successfully.

Veriler insert into ile eklendi

```
SQLQuery9.sql - LA...ELKG\saruhan (56)    SQLQuery8.sql - LA...ELKG\saruhan (51))* -p X
use vtys7hafta;

INSERT INTO toGrenci (ad, soyad, dogumTarihi, adres, telefon, bolumID, vizeNotu, finalNotu)
VALUES ('Kübra Nur', 'Saruhan', '2018-12-13', 'Samsun', '1111111111', 1, 70, 90 ),
       ('Sümeýra', 'Aydın', '2018-12-15', 'Ankara', '2222222222', 2, 80, 95);

100 %
Messages

(2 rows affected)
```

[illegible]

tOgrenciYedek tablosu oluşturuldu. SilineniEkle adında trigger oluşturuldu. 'AFTER DELETE' tOgrenci tablosunda delete işlemi yapıldıktan sonra çalışır anlamına gelir. Çalıştırılacak kod begin end arasına yazılır. Silmek istediğimiz data, ilk olarak arka plandaki sanal olarak oluşan 'deleted' tablosuna gider. Daha sonra silinir ve 'deleted' tablosu SQL belleğinden düşer. Insert Into ile deleted tablosuna giden veri tOgrenciYedek tablosuna eklenir.

```
SQLQuery5.sql - LA...ELKG\saruhan (52))* SQLQuery4.sql - LA...ELKG\saruhan (51))*
USE vtys7hafta;

CREATE TABLE tOgrenciYedek (
    ogrenciID INT PRIMARY KEY NOT NULL,
    ad NVARCHAR(10),
    soyad NVARCHAR(10),
    dogumTarihi DATE,
    adres NVARCHAR(15),
    telefon NVARCHAR(11),
    bolumID INT,
    vizeNotu INT,
    finalNotu INT
);
GO
CREATE TRIGGER SilineniEkle ON tOgrenci
AFTER DELETE
AS BEGIN
    INSERT INTO tOgrenciYedek SELECT * FROM deleted
END
```

100 %

Messages

Commands completed successfully.

Oluşturulan trigger test edilir. Silinen 4. id tOgrenciYedek tablosunada eklenmiştir.

```
LAPTOP-UK1FELKG.v...fta - dbo.tOgrenci cOgrenci trigger...1FELKG\saruhan (56))* SQLQuery11.sql - L...FELKG\saruhan (51))
USE vtys7hafta;

DELETE FROM tOgrenci where ogrenciID = 4;
SELECT * FROM tOgrenciYedek;
```

100 %

Results Messages

	ogrenciID	ad	soyad	dogumTarihi	adres	telefon	bolumID	vizeNotu	finalNotu
1	4	Sümeyra	Aydın	2018-12-15	Ankara	22222222222	2	80	95

2-Personel(Id,Ad,Maas) ve ZamListesi(Id,Ad,EskiMaas,YeniMaas,Tarih) adında iki tablomuz olsun. Bir personelin maaşına zam yapıldığında ZamListesi tablosuna ilgili personel bilgilerini, eski maaşını ve yeni maaşını ekleyin. (trigger kullanın)

Soruda istenilen personel ve zam listesi oluşturuldu. Insert into ile tabloya veriler eklendi.

```
USE vtys7hafta
GO
CREATE TABLE Personel(
id INT PRIMARY KEY IDENTITY(1,1),
ad NVARCHAR(25),
maas MONEY
)
CREATE TABLE ZamListesi(
id INT,
ad NVARCHAR(25),
eskiMaas MONEY,
yeniMaas MONEY,
tarih DATETIME
)
100 %
Messages
Commands completed successfully.
LAPTOP-UK1FELKG.v...fta - dbo.Personel SQLQuery14.sql - L...FELK
USE vtys7hafta
GO
INSERT INTO Personel(ad, maas)
VALUES('Kübra', 3000),
('Beyza', 5000),
('Sümeyra', 4000);
100 %
Messages
(3 rows affected)
```

ZamListesiEkleme adında trigger oluşturuldu. Update etmek istediğimiz datanın eski hali 'deleted' tablosuna gider, yeni hali de 'inserted' tablosuna kaydedilir. 'ON Personel AFTER UPDATE' Trigger Personel tablosunda güncellemeden sonra gerçekleşir anlamına gelir. 'deleted' ve 'inserted' tablosundaki veriler @ ile tanımlanan global değişkenlere atanır. Tanımlanan değişkenler Insert Into ile ZamListesine eklenir

```
SQLQuery17.sql - L...FELKG\saruhan (55))* SQLQuery15.sql - L...FELKG\saruhan (56))* SQLQuery16.sql - L...FELK
USE vtys7hafta
GO
CREATE TRIGGER ZamListesiEkleme
ON Personel
AFTER UPDATE
AS
DECLARE @id INT, @ad NVARCHAR(50), @eskiMaas MONEY, @yeniMaas MONEY
BEGIN
SELECT @id=inserted.id, @ad=inserted.ad, @eskiMaas=deleted.maas, @yeniMaas=inserted.maas
FROM inserted JOIN deleted ON inserted.id=deleted.id
INSERT INTO ZamListesi VALUES(@id, @ad, @eskiMaas, @yeniMaas, GETDATE())
END
100 %
Messages
Commands completed successfully.
```

Yazılan trigger test edilir. İd'si 1 olan personelin maaşı güncellendiğinde zam listesine de eski ve yeni maaşı eklenir.

```
USE vtys7hafta;

UPDATE Personel SET maas=20000 WHERE id=1

SELECT * FROM Personel
SELECT * FROM ZamListesi
```

id	ad	maas
1	Kübra	20000,00
2	Beyza	5000,00
3	SümeYra	4000,00

id	ad	eskiMaas	yeniMaas	tarih
1	Kübra	3000,00	20000,00	2018-12-13 19:26:06.220

3- Oyuncu(Id,Ad,Takım,GüncellemeTarihi) adında bir tablomuz olsun. Bu tablodaki herhangi bir kayıt güncellendiğinde GüncellemeTarihi otomatik olarak anlık tarihe update olsun. Bu işlemi gerçekleştiren bir trigger yazınız

Soruda belirtilen Oyuncu tablosu ddl ile oluşturulup Insert Into ile veriler eklendi.

```
SQLQuery18.sql - L...FELKG\saruhan (52))* X
USE vtys7hafta
GO
CREATE TABLE Oyuncu(
  Id INT PRIMARY KEY IDENTITY(1,1),
  Ad NVARCHAR(50),
  Takim NVARCHAR(50),
  GuncellemeTarihi DATETIME DEFAULT GETDATE()
)

Messages
Commands completed successfully.
```

```
LAPTOP-UK1FELKG....afta - dbo.Oyuncu SQLQuery20.sql
USE vtys7hafta
GO
INSERT INTO Oyuncu(Ad, Takim)
VALUES ('Ali', 'Fenerbahçe'),
      ('Ahmet', 'Galatasaray'),
      ('Mehmet', 'Besiktas');
```

(3 rows affected)

LAPTOP-UK1FELKG....afta - dbo.Oyuncu 3 insert.sql - LAPT...FELKG\saruhan (52))				
	Id	Ad	Takim	GuncellemeTa...
▶	3	Ali	Fenerbahçe	2018-12-13 19:5...
	4	Ahmet	Galatasaray	2018-12-13 19:5...
	5	Mehmet	Besiktas	2018-12-13 19:5...
*	NULL	NULL	NULL	NULL

OyuncuGuncelle adında trigger oluşturuldu. Update etmek istediğimiz verinin yeni hali de 'inserted' tablosuna kaydedilir. 'ON Oyuncu AFTER UPDATE' Trigger Oyuncu tablosunda güncellemeden sonra gerçekleşir anlamına gelir. 'inserted' tablosundaki id ile Oyuncu tablosundaki güncellenen id ile eşit ise GuncellemeTarihi GetDate() ile sistemin güncel zamanı ile güncellenir.

```

USE vtys7hafta
GO
CREATE TRIGGER OyuncuGuncelle
ON Oyuncu
AFTER UPDATE
AS
DECLARE @Id INT
BEGIN
SELECT @Id=Id FROM inserted
UPDATE Oyuncu SET GuncellemeTarihi=GETDATE() WHERE Id=@Id
END

```

100 %

Messages

Commands completed successfully.

Id'si 4 olan oyuncunun takımı güncellendiğinde Güncelleme tarihide değişiklik yapıldığı saati yani 20:22:04 olarak güncelledi.

SQLQuery23.sql - L...FELKG\saruhan (57))* LAPTOP-UK1FELKG....afta - d				
<pre> USE vtys7hafta GO UPDATE Oyuncu SET Takim='Fenerbahçe' WHERE Id=4 SELECT * FROM Oyuncu </pre>				
100 %				
Results Messages				
	Id	Ad	Takim	GuncellemeTarihi
1	3	Ali	Fenerbahçe	2018-12-13 19:57:49.633
2	4	Ahmet	Fenerbahçe	2018-12-13 20:22:04.833
3	5	Mehmet	Besiktas	2018-12-13 19:57:49.633

4-Kitap(Id,Ad,Yazar) ve KitapArsiv(Id,Ad,Yazar,ArsivTarihi) adında iki tablomuz olsun. Kitap tablosundan bir kayıt silindiğinde otomatik olarak arşiv tablosuna bir kayıt eklensin. Bu işlemi gerçekleştiren bir trigger yazınız

Soruda istenilen Kitap ve Kitap Arsiv tabloları oluşturuldu. Insert Into ile Kitap tablosuna veriler eklendi.

```
SQLQuery24.sql - L...FELKG\saruhan (55))* -> X L
USE vtys7hafta
GO
CREATE TABLE Kitap(
  Id INT PRIMARY KEY IDENTITY(1,1),
  Ad NVARCHAR(50),
  Yazar NVARCHAR(50)
)
CREATE TABLE KitapArsiv(
  Id INT,
  Ad NVARCHAR(50),
  Yazar NVARCHAR(50),
  ArsivTarihi DATETIME
)
100 %
Messages
Commands completed successfully.
```

```
SQLQuery25.sql - L...FELKG\saruhan (55))* -> X
USE vtys7hafta
GO
INSERT INTO Kitap
VALUES('Küçük Ağa','Tarık Buğra'),
      ('Kürk Mantolu Madonna','Sabahattin Ali'),
      ('İnsan Ne ile Yaşar','Tolstoy')
100 %
Messages
(3 rows affected)
```

	Id	Ad	Yazar
▶	1	Küçük Ağa	Tarık Buğra
	2	Kürk Mantolu ...	Sabahattin Ali
	3	İnsan Ne ile Yaşar	Tolstoy
*	NULL	NULL	NULL

KitapArsivEkle adında trigger oluşturuldu. 'ON Kitap AFTER DELETE' Trigger Kitap tablosundan veri silindikten sonra gerçekleşir anlamına gelir. Delete etmek istediğimiz verinin eski hali 'deleted' tablosuna kaydedilir. Deleted tablosundaki veriler değişkenlere atanır. Insert Into ile KitapArsiv' değişkenler eklenir. GetDate sistemin o anki zamanını alır.

```
CREATE TRIGGER KitapArsiveEkle
ON Kitap
AFTER DELETE
AS
DECLARE @KitapId INT, @KitapAd NVARCHAR(50), @KitapYazar NVARCHAR(50)
BEGIN
SELECT @KitapId=Id, @KitapAd=Ad, @KitapYazar=Yazar FROM deleted
INSERT INTO KitapArsiv VALUES(@KitapId, @KitapAd, @KitapYazar, GETDATE())
END
```

100 %

Messages

Commands completed successfully.

Id'si 1 olan kitap tablodan silini KitapArsive eklenmiştir.

```
USE vtys7hafta
GO
DELETE Kitap WHERE Id=1;
SELECT * FROM Kitap;
SELECT * FROM KitapArsiv;
```

100 %

Results Messages

Id	Ad	Yazar
1	Kürk Mantolu Madonna	Sabahattin Ali
2	İnsan Ne ile Yaşar	Tolstoy

Id	Ad	Yazar	ArsivTarihi
1	Küçük Ağa	Tank Buğra	2018-12-13 20:35:08.237

5-siparis(id,ürünId,miktar,tarih ve ürün(id,ad,fiyat,stok) adında iki tablo yaratalım. sipariş tablosuna bir insert yapıldığında, ilgili ürünün stok miktarını da güncelleyin.

Soruda istenilen Urun Ve Sipariş tabloları oluşturuldu.

SQLQuery28.sql - L...FELKG\saruhan (56))* -> X 4 sorgu arşive ekl

```
USE vtys7hafta
GO
CREATE TABLE Urun(
Id INT PRIMARY KEY IDENTITY(1,1),
Ad NVARCHAR(50),
Fiyat INT,
Stok INT
)
CREATE TABLE Siparis(
Id INT PRIMARY KEY IDENTITY(1,1),
UrunId INT,
Miktar INT,
Tarih DATETIME
FOREIGN KEY (UrunId) REFERENCES Urun(Id)
)
```

100 %

Messages

Commands completed successfully.

Urun tablosuna veriler eklendi.

```
USE vtys7hafta
GO
INSERT INTO Urun
VALUES('Çikolata',1, 500),
      ('Şeker',6, 300),
      ('Un',10, 200)
```

100 %

Messages

(3 rows affected)

UrunGuncelle adında trigger oluşturuldu. 'ON Siparis AFTER INSERT' Trigger Siparis tablosunda eklemeyen sonra gerçekleşir anlamına gelir. 'inserted' tablosundaki veriler @ ile tanımlanan global değişkenlere atanır. Tanımlanan değişkenlerde Miktar Stok bilgisinden çıkartılarak Urun tablosu güncellenir.

```
SQLQuery30.sql - L...FELKG\saruhan (58))* SQLQuery31.sql - L...FELKG\saruhan (59)*
USE vtys7hafta
GO
CREATE TRIGGER UrunGuncelle
ON Siparis
AFTER INSERT
AS
DECLARE @UrunId INT,@Miktar INT
BEGIN
SELECT @UrunId=UrunId, @Miktar=Miktar FROM inserted
UPDATE Urun SET Stok = Stok - @Miktar WHERE Id = @UrunId
END
```

100 %

Messages

Commands completed successfully.

Id'si 1, Miktarı 20 ve Şimdiki zamanı bilgisi eklendiğinde Urun tablosundaki stok güncellenir, Sipariş tablosuna eklenir.

```
SQLQuery30.sql - L...FELKG\saruhan (58))* SQLQuery31.sql - L...FELKG\saruhan (59)*
USE vtys7hafta
GO
INSERT INTO Siparis VALUES(1,20,GETDATE())
SELECT * FROM Urun
SELECT * FROM Siparis
```

100 %

Results Messages

	Id	Ad	Fiyat	Stok
1	1	Çikolata	1	480
2	2	Şeker	6	300
3	3	Un	10	200

	Id	UrunId	Miktar	Tarih
1	1	1	20	2018-12-13 21:01:41.903