

# VERİTABANI TASARIMI

Öğretim Görevlisi A. Berika VAROL MALKOÇOĞLU

# İçindekiler

- Tetikleyiciler
- Union ifadeleri

# Tetikleyiciler (Triggers)

- Bir tablo üzerinde
  - Ekleme,
  - Silme ve
  - Güncellemeye bağlı olarak tetiklenip çalışan SQL kodlarıdır.
- Bir tetikleyici olayı meydana geldiğinde, trigger fonksiyonu uygun bir zamanda veriyi işlemek üzere çağrılır.
- Saklı yordamların özel bir türüdür.

# Tetikleyiciler X Saklı Yordamlar

- Tetikleyiciler veriler üzerinde değişiklik yapıldığında belirlenen talimatlara göre otomatik çalışır.
- Saklı yordamları el ile çalıştırmak gerekir.

# Tetikleyiciler (Triggers)

- Genel amacı;
  - Veri bütünlüğünü korur.
  - Veriler üzerinde bir işlem yapıldığında, iş kurallarına uygun gerçekleşip gerçekleşmediği kontrol edilerek hatalı girişlerin yapılması önler.
  - Veri güvenliğini sağlar.
  - Bir kullanıcı silindiğinde onunla ilgili satış tablosundaki siparişleri silmek istediğimizde tanımlanır.
  - Tablolarda veya veritabanında mevcut olmayan ek bilgiyi türetmek için kullanılır.

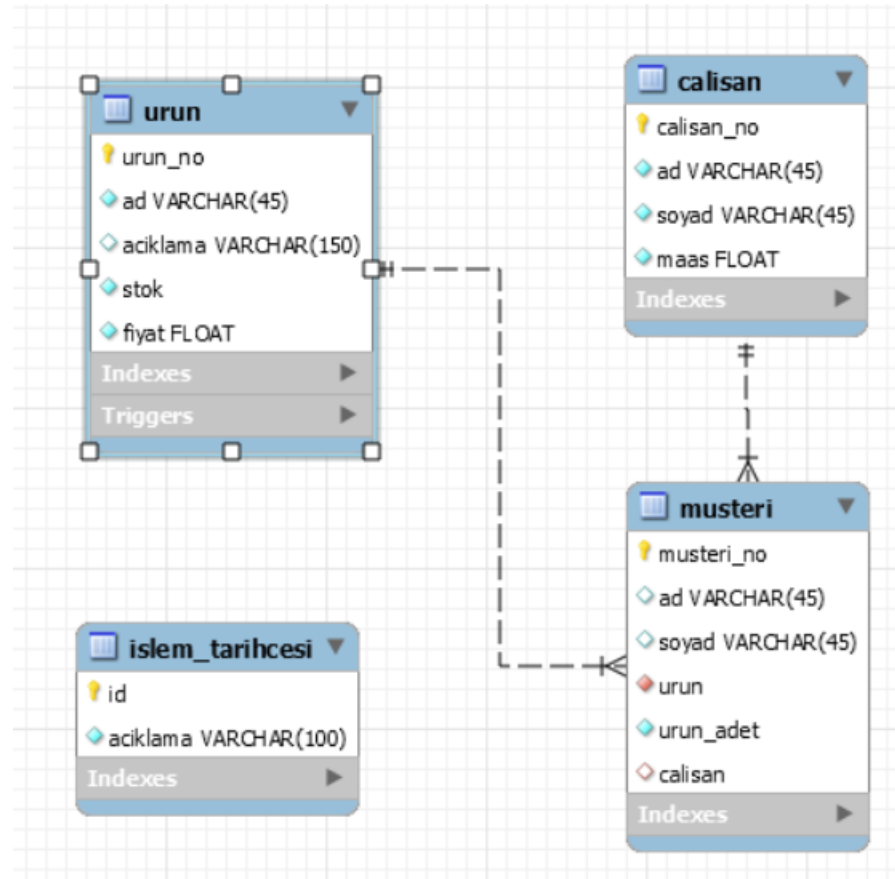
# Tetikleyiciler (Triggers)

- DML tetikleyicileri iki tip olarak sınıflandırılır;
  - **AFTER** tetikleyicileri: Bir tabloda INSERT, UPDATE, DELETE işlemleri yapıldıktan sonra belli işlemlerin yapılması için kullanılmaktadır.
  - **BEFORE** tetikleyicileri: Bir tablo INSERT, UPDATE veya DELETE işlemlerini yapmadan önce belli işlemlerin yapılması için kullanılmaktadır.
- 6 adet tetikleyiciyi vardır:
  - Before Insert
  - After Insert
  - Before Update
  - After Update
  - Before Delete
  - After Delete

# Tetikleyiciler (Triggers)

```
CREATE TRIGGER trigger_adi [BEFORE/AFTER] [INSERT/UPDATE/DELETE]  
ON tablo_adi  
FOR EACH ROW  
BEGIN  
    -- SQL komutları  
END;
```

# Eczane Veritabanı





# Before Insert

- Eczane veritabanında ürün tablosuna bir ilaç eklendiğinde işlem tablosuna ilacın adının eklendiği bilgisinin eklenmesi;

```
16 • use eczane;
17
18 DELIMITER //
19 • CREATE TRIGGER bir_urun_eklendi BEFORE INSERT
20 ON urun
21 FOR EACH ROW
22 BEGIN
23     INSERT INTO islem_tarihcesi(aciklama) VALUES(CONCAT(NEW.ad, 'eklendi.'));
24 END//
25 DELIMITER ;
26
```

```
--
27 • insert into eczane.urun values (5,'Bepanthen','krem',20,17.49);
28
29 • select * from islem_tarihcesi;
```

Result Grid | Filter Rows:  | Edit: | Export/Import: | V

	id	aciklama
▶	2	Bepanthen eklendi.
•	NULL	NULL

```
13 • select * from eczane.urun;
14
```

Result Grid | Filter Rows:  | Edit:

	urun_no	ad	aciklama	stok	fiyat
▶	1	Aspirin	NULL	15	3.98
	2	Paorl	ağrı kesici	10	8.18
	3	Aferin	grip ilacı	5	13.7
	4	Katarin	grip ilacı	23	7.99
	5	Bepanthen	krem	20	17.49

# After Insert

- Eczane veritabanında müşteri bir ilaç aldığıında stok değerinin güncellenmesi;

```
1  DELIMITER //
```

```
2  • CREATE TRIGGER stok_guncellendi AFTER INSERT
```

```
3  ON musteri
```

```
4  FOR EACH ROW
```

```
5  BEGIN
```

```
6      DECLARE satis INT;
```

```
7      DECLARE urun_id INT;
```

```
8      SELECT urun_adet INTO satis FROM musteri where musteri_no=new.musteri_no;
```

```
9      SELECT urun INTO urun_id FROM musteri where musteri_no=new.musteri_no;
```

```
10     UPDATE eczane.urun SET stok=stok-satis WHERE urun_no=urun_id;
```

```
11 END//
```

```
12 DELIMITER ;
```

```
13
```

```
13 • select * from eczane.urun;
```

```
14
```

Result Grid					
Filter Rows: <input type="text"/>					
	urun_no	ad	aciklama	stok	fiyat
▶	1	Aspirin	NULL	15	3.98
	2	Paorl	ağrı kesici	10	8.18
	3	Aferin	grip ilacı	5	13.7
	4	Katarin	grip ilacı	23	7.99
	5	Bepanthen	krem	20	17.49
*	NULL	NULL	NULL	NULL	NULL

1

```
43 • insert into eczane.musteri values (100,'Aslı','Can',1,2,10);
```

```
44
```

```
45 • select * from eczane.musteri;
```

Result Grid						
Filter Rows: <input type="text"/>						
	musteri_no	ad	soyad	urun	urun_adet	calisan
▶	100	Aslı	Can	1	2	10
*	NULL	NULL	NULL	NULL	NULL	NULL

2

```
65 • select * from eczane.urun;
```

```
66
```

Result Grid					
Filter Rows: <input type="text"/>					
	urun_no	ad	aciklama	stok	fiyat
▶	1	Aspirin	NULL	13	3.98
	2	Paorl	ağrı kesici	9	8.18
	3	Aferin	grip ilacı	5	13.7
	4	Katarin	grip ilacı	23	7.99
	5	Bepanthen	krem	20	17.49
*	NULL	NULL	NULL	NULL	NULL

3

# After Update

- Eczane veritabanında ürün tablosunda bir ilaç güncellendiğinde işlem tablosuna bilginin eklenmesi;

```
35 DELIMITER //
36 • CREATE TRIGGER bir_urun_guncellendi AFTER UPDATE
37 ON urun
38 FOR EACH ROW
39 BEGIN
40     INSERT INTO islem_tarihcesi(aciklama) VALUES(CONCAT('ürün tablosu güncellendi.'));
41 END//
42 DELIMITER ;
43
```

# After Update

```
44 • update urun set ad="Parol" where urun_no=2;  
45  
46 • select * from urun;
```

	urun_no	ad	aciklama	stok	fiyat
▶	1	Aspirin	NULL	13	3.98
	2	Parol	ağrı kesici	9	8.18
	4	Katarin	grip ilacı	23	7.99
	5	Bepanthen	krem	20	17.49
*	NULL	NULL	NULL	NULL	NULL

48

```
49 • select * from islem_tarihcesi;
```

	id	aciklama
▶	2	Bepanthen eklendi.
	3	Aferin silindi.
	4	ürün tablosu güncellendi.
*	NULL	NULL

# Before Delete

- Eczane veritabanında ürün tablosundan bir ilaç silindiğinde işlem tablosuna ilacın adının silindiği bilgisinin eklenmesi;

```
22     DELIMITER //
```

```
23 •   CREATE TRIGGER bir_urun_silindi BEFORE DELETE
```

```
24     ON urun
```

```
25     FOR EACH ROW
```

```
26     BEGIN
```

```
27         INSERT INTO islem_tarihcesi(aciklama) VALUES(CONCAT(OLD.ad, ' silindi.'));
```



```
28     END//
```

```
29     DELIMITER ;
```

# Before Delete

```
32 • DELETE FROM urun WHERE urun_no=3;
33
34 • select * from islem_tarihcesi;
35
36
```



<

Result Grid   Filter Rows:  Edit:

	id	aciklama
▶	2	Bepanthen eklendi.
	3	Aferin silindi.
•	NULL	NULL

```
36 • select * from urun;
```

<

Result Grid   Filter Rows:  Edit:

	urun_no	ad	aciklama	stok	fiyat
▶	1	Aspirin	NULL	13	3.98
	2	Paorl	ağrı kesici	9	8.18
	4	Katarin	grip ilacı	23	7.99
	5	Bepanthen	krem	20	17.49
•	NULL	NULL	NULL	NULL	NULL



# Tetikleyiciler (Triggers)

Bağlam	New.field	Old.field
Before / After Insert	Yeni değerleri içerir	Hiç bir değer içermez
Before / After Update	Yeni değerleri içerir	Eski değerleri içerir
Before / After Delete	Hiç bir değer içermez	Eski değerleri içerir

# Tetikleyicileri Görüntülemek

- Oluşturulan tetikleyicileri görmek için;

**SHOW TRIGGERS;**

51 • **SHOW TRIGGERS;**

<

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	Trigger	Event	Table	Statement	Timing	Created	sql_mode	Definer	character_set_client	collatio
▶	stok_guncellendi	INSERT	musteri	BEGIN DECLARE satis INT; DECLARE urun_jd...	AFTER	2020-10-25 21:19:52.10	STRICT_TRANS_TABLES,NO_ENGINE_SUBSTIT...	root@localhost	utf8mb4	utf8mb
	bir_urun_eklendi	INSERT	urun	BEGIN INSERT INTO islem_tarihcesi(aciklama) ...	BEFORE	2020-10-25 20:05:40.54	STRICT_TRANS_TABLES,NO_ENGINE_SUBSTIT...	root@localhost	utf8mb4	utf8mb
	bir_urun_guncellendi	UPDATE	urun	BEGIN INSERT INTO islem_tarihcesi(aciklama) ...	AFTER	2020-10-25 21:41:22.70	STRICT_TRANS_TABLES,NO_ENGINE_SUBSTIT...	root@localhost	utf8mb4	utf8mb
	bir_urun_silindi	DELETE	urun	BEGIN INSERT INTO islem_tarihcesi(aciklama) ...	BEFORE	2020-10-25 21:33:09.34	STRICT_TRANS_TABLES,NO_ENGINE_SUBSTIT...	root@localhost	utf8mb4	utf8mb

# Tetikleyicileri Silmek

- Oluşturulan tetikleyicileri silmek için;

**DROP TRIGGER** calisan\_son\_guncelleme;

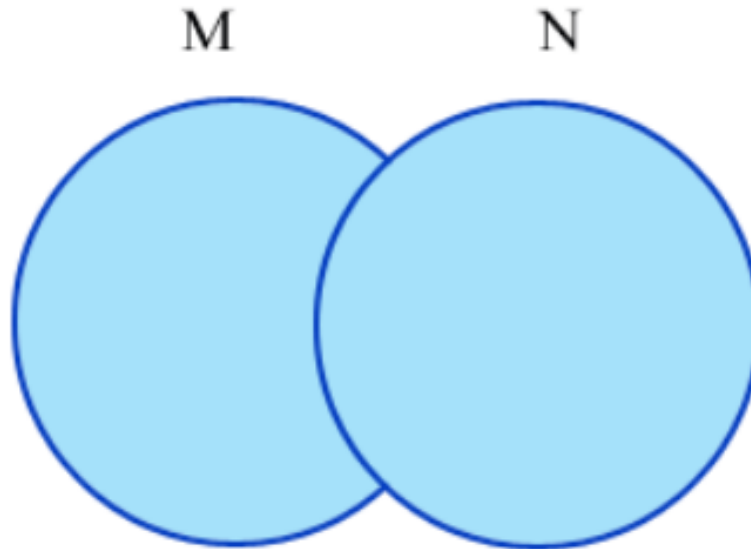
# Union ve Union ALL

- İki veya daha fazla sorgunun sonuç kümesini, UNION ve UNION ALL operatörlerini kullanarak tek bir sonuç kümesinde birleştirir.

```
SELECT column_name(s) FROM table1  
UNION / UNION ALL  
SELECT column_name(s) FROM table2;
```

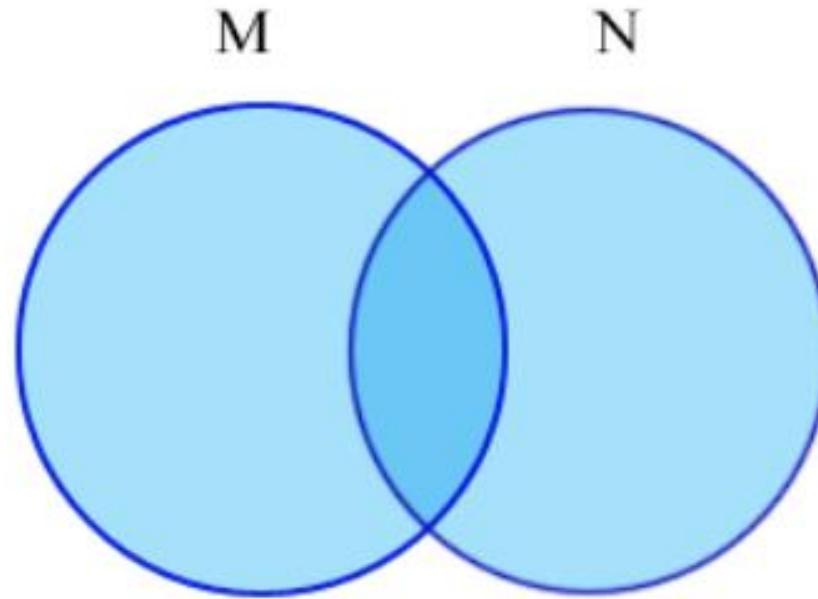
# Union

- UNION operatörü, yinelenen satırları eledikten sonra her iki sorgudan da sonucu döndürür.



# Union All

- UNION ALL operatörü, her iki sorudan tüm satırları döndürür ve hiçbir yinleme eliminasyonu olmaz.



# Union Örneği

## Employees tablosu

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID
100	Steven	King	SKING	515.123.4567	17-Jun-87	AD_PRES
101	Neena	Kochhar	NKOCHHAR	515.123.4568	21-Sep-89	AD_VP
102	Lex	De Haan	LDEHAAN	515.123.4569	13-Jan-93	AD_VP
103	Alexander	Hunold	AHUNOLD	590.423.4567	3-Jan-90	IT_PROG
104	Bruce	Ernst	BERNST	590.423.4568	21-May-91	IT_PROG
105	David	Austin	DAUSTIN	590.423.4569	25-Jun-97	IT_PROG

## Job History tablosu

EMPLOYEE_ID	START_DATE	END_DATE	JOB_ID	DEPARTMENT_ID
102	13-Jan-93	24-Jul-98	IT_PROG	60
101	21-Sep-89	27-Oct-93	AC_ACCOUNT	110
101	28-Oct-93	15-Mar-97	AC_MGR	110
201	17-Feb-96	19-Dec-99	MK_REP	20
114	24-Mar-98	31-Dec-99	ST_CLERK	50
122	1-Jan-99	31-Dec-99	ST_CLERK	50

# Union

1	Select employee_id, job_id
2	FROM employees
3	UNION
4	Select employee_id, job_id
5	FROM job_history;

employee_id	job_id
100	AD_PRES
101	AD_VP
102	AD_VP
103	IT_PROG
104	IT_PROG
105	IT_PROG
106	IT_PROG
107	IT_PROG
108	FI_MGR
109	FI_ACCOUNT
110	FI_ACCOUNT
111	FI_ACCOUNT
112	FI_ACCOUNT
113	FI_ACCOUNT
114	PU_MAN
115	PU_CLERK
116	PU_CLERK
117	PU_CLERK



# Union All

1	Select employee_id, job_id, department_id
2	FROM employees
3	UNION ALL
4	Select employee_id, job_id, department_id
5	FROM job_history;

employee_id	job_id	department_id
100	AD_PRES	90
101	AD_VP	90
102	AD_VP	90
103	IT_PROG	60
104	IT_PROG	60
105	IT_PROG	60
106	IT_PROG	60
107	IT_PROG	60
108	FI_MGR	100
109	FI_ACCOUNT	100
110	FI_ACCOUNT	100
111	FI_ACCOUNT	100
112	FI_ACCOUNT	100
113	FI_ACCOUNT	100
114	PU_MAN	30
115	PU_CLERK	30
116	PU_CLERK	30
117	PU_CLERK	30
118	PU_CLERK	30
119	PU_CLERK	30
120	ST_MAN	50
121	ST MAN	50