
SQL

Structured Query Language
Yapılandırılmış Sorgu Dili

22 Ocak 2021
DERS 11

REGEXP_LIKE,
OFFSET, FETCH NEXT
PIVOT CLAUSES, ALTER

Mehmet Bulutluoz
Elektronik muh.

Önceki Dersten Hatırladıklarımız

1- JOIN : Bir tablonun kendisinden veya iki farklı tablodan istediğimiz field'ları kullanarak yeni tablo elde etmek için kullanılır.

- join işlemi yeni tablo oluşturmaz fakat yazdığımız sorgu sonuçlarını istediğimiz şekilde bir tabloda görmemizi sağlar
 - Daha önce gördüğümüz union,intersect,minus komutları tabloları birleştirmiyor, ayrı ayrı sorgular yapıp sorgu sonuçlarını birleştiriyordu. Hatta farklı tablolardan gelen ism ve şehir gibi değerleri tablo olarak yansıtamadığından alt alta yazmamız gerekiyordu.
 - INNER : her iki tabloda ortak olan recordlara ait her iki tablodaki bilgileri gösterir
 - LEFT JOIN: Birinci tablodaki tüm recordları ve ortak recordlara ait ikinci tablodaki bilgileri gösterir
 - RIGHT JOIN : İkinci tablodaki tüm recordları ve ortak recordlara ait birinci tablodaki bilgileri gösterir
 - FULL JOIN : Her iki tablodaki tüm record'ları gösterir, bir tabloda olup diğeri tabloda karşılığı olmayan bilgiler için null yazar
-

Önceki Dersten Hatırladıklarımız

- SELF JOIN : Bir tablonun kendi içerisinde farklı bir kurala göre INNER JOIN yapılmasıdır.

Aynı tablo iki kere kullanılacağı için geçici olarak tabloya iki farklı isim verilir ve bu genelde tablo adının ilk harfi ile 1 ve 2 rakamlarının kullanılmasıyla olur (p1,p2 gibi)

FROM satırında tablo adı ile kısaltmalar birlikte yazılarak tanımlanır

FROM personel p1 INNER JOIN personel p2

ON komutundan sonra oluşturulan iki sanal tablo arasındaki kural tanımlanır

- LIKE : WHERE komutundan sonra kullanılır, wildcards kullanımina izin vererek PATTERN MATCHING (şekil benzetme) özelliğini kullanır
 - (%) : yazdığımız yerden sonra 0 veya daha fazla karakter olabilir
 - (_) : sadece 1 tane karakteri gösterir

LIKE Condition

3) [] REGEXP_LIKE => sadece bir karakteri gösterir.

```
CREATE TABLE kelimeler  
(  
  id number(10) UNIQUE,  
  kelime varchar2(50) NOT NULL,  
  Harf_sayisi number(6)  
);
```

```
INSERT INTO kelimeler VALUES (1001, 'hot', 3);  
INSERT INTO kelimeler VALUES (1002, 'hat', 3);  
INSERT INTO kelimeler VALUES (1003, 'hit', 3);  
INSERT INTO kelimeler VALUES (1004, 'hbt', 3);  
INSERT INTO kelimeler VALUES (1008, 'hct', 3);  
INSERT INTO kelimeler VALUES (1005, 'adem', 4);  
INSERT INTO kelimeler VALUES (1006, 'selim', 5);  
INSERT INTO kelimeler VALUES (1007, 'yusuf', 5);
```

SORU : İlk harfi h, son harfi t olup 2. harfi a veya i olan 3 harfli kelimelerin tüm bilgilerini yazdıran QUERY yazın

```
SELECT *  
FROM kelimeler  
WHERE REGEXP_LIKE (kelime, 'h[ai]t');
```

LIKE Condition

SORU : İlk harfi h,son harfi t olup 2.harfi a ile k arasinda olan 3 harfli kelimelerin tum bilgilerini yazdiran QUERY yazin

```
SELECT *  
FROM kelimeler  
WHERE REGEXP_LIKE (kelime, 'h[a-k]t');
```

ID	KELIME	HARF_SAYISI
1002	hat	3
1003	hit	3
1004	hbt	3
1008	hct	3

SORU : Icinde m veya i olan kelimelerin tum bilgilerini yazdiran QUERY yazin

```
SELECT *  
FROM kelimeler  
WHERE REGEXP_LIKE (kelime, '[mi](*) '); [a|n] de olur
```

ID	KELIME	HARF_SAYISI
1003	hit	3
1005	adem	4
1006	selim	5

SORU : a veya s ile baslayan kelimelerin tum bilgilerini yazdiran QUERY yazin

```
SELECT *  
FROM kelimeler  
WHERE REGEXP_LIKE (kelime, '^[as] ');
```

ID	KELIME	HARF_SAYISI
1005	adem	4
1006	selim	5

LIKE Condition

SORU : m veya f ile biten kelimelerin tum bilgilerini yazdiran QUERY yazin

SELECT *

FROM kelimeler

WHERE REGEXP_LIKE (kelime, '[ea]\$');

ID	KELIME	HARF_SAYISI
1005	adem	4
1006	selim	5
1007	yusuf	5

TECHPROED

NOT LIKE Condition

SORU 1 : ilk harfi h olmayan kelimelerin tum bilgilerini yazdiran QUERY yazin

```
SELECT *  
FROM kelimeler  
WHERE kelime NOT LIKE 'h%';
```

ID	KELIME	HARF_SAYISI
1005	adem	4
1006	selim	5
1007	yusuf	5

SORU 2 : a harfi icermeyen kelimelerin tum bilgilerini yazdiran QUERY yazin

```
SELECT *  
FROM kelimeler  
WHERE kelime NOT LIKE '%a%';
```

ID	KELIME	HARF_SAYISI
1001	hot	3
1003	hit	3
1004	hbt	3
1008	hct	3
1006	selim	5
1007	yusuf	5

NOT LIKE Condition

SORU 3 : ikinci ve ucuncu harfi 'de' olmayan kelimelerin tum bilgilerini yazdiran QUERY yazin

```
SELECT *  
FROM kelimeler  
WHERE kelime NOT LIKE '_de%';
```

ID	KELIME	HARF_SAYISI
1001	hot	3
1002	hat	3
1003	hit	3
1004	hbt	3
1008	hct	3
1006	selim	5
1007	yusuf	5

SORU 4 : 2. harfi e,i veya o olmayan kelimelerin tum bilgilerini yazdiran QUERY yazin

```
SELECT *  
FROM kelimeler  
WHERE NOT REGEXP_LIKE (kelime, '[_eio]');
```

ID	KELIME	HARF_SAYISI
1002	hat	3
1004	hbt	3
1008	hct	3
1007	yusuf	5

UPPER – LOWER - INITCAP

Tabloları yazdırırken büyük harf, küçük harf veya ilk harfleri büyük diğerleri küçük harf yazdırmak için kullanırız

```
SELECT UPPER(kelime)
FROM kelimeler;
```

UPPER(KELIME)
HOT
HAT
HIT
HBT
HCT
ADEM
SELIM
YUSUF

```
SELECT LOWER(kelime)
FROM kelimeler;
```

LOWER(KELIME)
hot
hat
hit
hbt
hct
adem
selim
yusuf

```
SELECT INITCAP(kelime)
FROM kelimeler;
```

INITCAP(KELIME)
Hot
Hat
Hit
Hbt
Hct
Adem
Selim
Yusuf

DISTINCT

URUN_ID	MUSTERI_ISIM	URUN_ISIM
10	Ali	Portakal
10	Ali	Portakal
20	Veli	Elma
30	Ayşe	Armut
20	Ali	Elma
10	Adem	Portakal
40	Veli	Kaysi
20	Elif	Elma

```
SELECT DISTINCT urun_isim  
FROM muster_iurun;
```

URUN_ISIM
Elma
Portakal
Kaysi
Armut

```
SELECT DISTINCT muster_iisim  
FROM muster_iurun;
```

MUSTERI_ISIM
Veli
Ayşe
Elif
Adem
Ali

Tabloda kaç farklı meyve vardır ?

```
SELECT COUNT(DISTINCT urun_isim) AS urun_cesit_sayisi  
FROM muster_iurun;
```

URUN_CESIT_SAYISI
4

FETCH NEXT (SAYI) ROW ONLY- OFFSET

1) Tabloyu urun_id ye gore siralayiniz

URUN_ID	MUSTERI_ISIM	URUN_ISIM
10	Ali	Portakal
10	Ali	Portakal
10	Adem	Portakal
20	Veli	Elma
20	Elif	Elma
20	Ali	Elma
30	Ayşe	Armut
40	Veli	Kaysi

2) Sirali tablodan ilk 3 kaydi listeleyin

```
SELECT *  
FROM musteri_urun  
ORDER BY urun_id  
FETCH NEXT 3 ROW ONLY;
```

URUN_ID	MUSTERI_ISIM	URUN_ISIM
10	Ali	Portakal
10	Adem	Portakal
10	Ali	Portakal

3) Sirali tablodan 4. kayittan 7.kayida kadar olan kayitlari listeleyin

```
SELECT *  
FROM musteri_urun  
ORDER BY urun_id  
OFFSET 3 ROW  
FETCH NEXT 4 ROW ONLY;
```

URUN_ID	MUSTERI_ISIM	URUN_ISIM
20	Veli	Elma
20	Elif	Elma
20	Ali	Elma
30	Ayşe	Armut

PIVOT CLAUSES

```
CREATE TABLE muster_i_urun
(
  urun_id number(10),
  muster_i_isim varchar2(50),
  urun_isim varchar2(50)
);
```

```
INSERT INTO muster_i_urun VALUES (10, 'Ali', 'Portakal');
INSERT INTO muster_i_urun VALUES (10, 'Ali', 'Portakal');
INSERT INTO muster_i_urun VALUES (20, 'Veli', 'Elma');
INSERT INTO muster_i_urun VALUES (30, 'Ayse', 'Armut');
INSERT INTO muster_i_urun VALUES (20, 'Ali', 'Elma');
INSERT INTO muster_i_urun VALUES (10, 'Adem', 'Portakal');
INSERT INTO muster_i_urun VALUES (40, 'Veli', 'Kaysi');
INSERT INTO muster_i_urun VALUES (20, 'Elif', 'Elma');
```

```
SELECT *
FROM (SELECT urun_isim, muster_i_isim FROM muster_i_urun)
PIVOT (COUNT(urun_isim) FOR urun_isim IN ('Portakal', 'Elma', 'Kaysi', 'Armut'));
```

MUSTERI_ISIM	'Portakal'	'Elma'	'Kaysi'	'Armut'
Veli	0	1	1	0
Ayse	0	0	0	1
Elif	0	1	0	0
Adem	1	0	0	0
Ali	2	1	0	0

```
SELECT *
FROM (SELECT urun_isim, muster_i_isim FROM muster_i_urun)
PIVOT (COUNT(muster_i_isim) FOR muster_i_isim IN ('Ali', 'Veli', 'Ayse', 'Adem', 'Elif'));
```

URUN_ISIM	'Ali'	'Veli'	'Ayse'	'Adem'	'Elif'
Elma	1	1	0	0	1
Portakal	2	0	0	1	0
Kaysi	0	1	0	0	0
Armut	0	0	1	0	0

ALTER TABLE STATEMENT

ALTER TABLE statement tabloda **add**, **modify**, veya **drop/delete columns** islemleri için kullanılır.

ALTER TABLE statement tabloları yeniden isimlendirmek için de kullanılır.

```
CREATE TABLE personel
(  
  id number(9),  
  isim varchar2(50),  
  sehir varchar2(50),  
  maas number(20),  
  sirket varchar2(20),  
  CONSTRAINT personel_pk PRIMARY KEY (id)  
);
```

```
INSERT INTO personel VALUES(123456789, 'Ali Yilmaz', 'Istanbul', 5500, 'Honda');  
INSERT INTO personel VALUES(234567890, 'Veli Sahin', 'Istanbul', 4500, 'Toyota');  
INSERT INTO personel VALUES(345678901, 'Mehmet Ozturk', 'Ankara', 3500, 'Honda');  
INSERT INTO personel VALUES(456789012, 'Mehmet Ozturk', 'Izmir', 6000, 'Ford');  
INSERT INTO personel VALUES(567890123, 'Mehmet Ozturk', 'Ankara', 7000, 'Tofas');  
INSERT INTO personel VALUES(456715012, 'Veli Sahin', 'Ankara', 4500, 'Ford');  
INSERT INTO personel VALUES(123456710, 'Hatice Sahin', 'Bursa', 4500, 'Honda');
```

ID	ISIM	SEHIR	MAAS	SIRKET
123456789	Ali Yilmaz	Istanbul	5500	Honda
234567890	Veli Sahin	Istanbul	4500	Toyota
345678901	Mehmet Ozturk	Ankara	3500	Honda
456789012	Mehmet Ozturk	Izmir	6000	Ford
567890123	Mehmet Ozturk	Ankara	7000	Tofas
456715012	Veli Sahin	Ankara	4500	Ford
123456710	Hatice Sahin	Bursa	4500	Honda

ALTER TABLE STATEMENT

1) ADD default deger ile tabloya bir sutun ekleme

```
ALTER TABLE personel  
ADD ulke_isim varchar2(20) DEFAULT 'Turkiye';
```

ID	ISIM	SEHIR	MAAS	SIRKET	ULKE_ISIM
123456789	Ali Yilmaz	Istanbul	5500	Honda	Turkiye
234567890	Veli Sahin	Istanbul	4500	Toyota	Turkiye
345678901	Mehmet Ozturk	Ankara	3500	Honda	Turkiye
456789012	Mehmet Ozturk	Izmir	6000	Ford	Turkiye
567890123	Mehmet Ozturk	Ankara	7000	Tofas	Turkiye
456715012	Veli Sahin	Ankara	4500	Ford	Turkiye
123456710	Hatice Sahin	Bursa	4500	Honda	Turkiye

2) Tabloya birden fazla sutun ekleme

```
ALTER TABLE personel  
ADD (cinsiyet varchar2(20) , yas number(3));
```

ID	ISIM	SEHIR	MAAS	SIRKET	ULKE_ISIM	CINSIYET	YAS
123456789	Ali Yilmaz	Istanbul	5500	Honda	Turkiye	-	-
234567890	Veli Sahin	Istanbul	4500	Toyota	Turkiye	-	-
345678901	Mehmet Ozturk	Ankara	3500	Honda	Turkiye	-	-
456789012	Mehmet Ozturk	Izmir	6000	Ford	Turkiye	-	-
567890123	Mehmet Ozturk	Ankara	7000	Tofas	Turkiye	-	-
456715012	Veli Sahin	Ankara	4500	Ford	Turkiye	-	-
123456710	Hatice Sahin	Bursa	4500	Honda	Turkiye	-	-

ALTER TABLE STATEMENT

3) DROP tablodan sutun silme

ALTER TABLE personel
DROP COLUMN yas;

ID	ISIM	SEHIR	MAAS	SIRKET	ULKE_ISIM	CINSIYET
123456789	Ali Yilmaz	Istanbul	5500	Honda	Turkiye	-
234567890	Veli Sahin	Istanbul	4500	Toyota	Turkiye	-
345678901	Mehmet Ozturk	Ankara	3500	Honda	Turkiye	-
456789012	Mehmet Ozturk	Izmir	6000	Ford	Turkiye	-
567890123	Mehmet Ozturk	Ankara	7000	Tofas	Turkiye	-
456715012	Veli Sahin	Ankara	4500	Ford	Turkiye	-
123456710	Hatice Sahin	Bursa	4500	Honda	Turkiye	-

4) RENAME COLUMN sutun adi degistirme

ALTER TABLE personel
RENAME COLUMN ulke_isim TO ulke_adi;

ID	ISIM	SEHIR	MAAS	SIRKET	ULKE_ADI	CINSIYET
123456789	Ali Yilmaz	Istanbul	5500	Honda	Turkiye	-
234567890	Veli Sahin	Istanbul	4500	Toyota	Turkiye	-
345678901	Mehmet Ozturk	Ankara	3500	Honda	Turkiye	-
456789012	Mehmet Ozturk	Izmir	6000	Ford	Turkiye	-
567890123	Mehmet Ozturk	Ankara	7000	Tofas	Turkiye	-
456715012	Veli Sahin	Ankara	4500	Ford	Turkiye	-
123456710	Hatice Sahin	Bursa	4500	Honda	Turkiye	-

ALTER TABLE STATEMENT

5) RENAME tablonun ismini degistirme

ALTER TABLE personel
RENAME TO isciler;

ISCILER
Table Status: Valid Created 17 minutes ago

6) MODIFY sutunlarin ozelliklerini degistirme

ALTER TABLE isciler
MODIFY ulke_adi varchar2(30) NOT NULL;

Constraints		
Constraint	Type	Condition
SYS_C0040949308	Check	"ULKE_ADI" IS NOT NULL
PERSONEL_PK	Primary Key	-

Columns			
#	Column	Type	Length
1	ID	NUMBER	22
2	ISIM	VARCHAR2	50
3	SEHIR	VARCHAR2	50
4	MAAS	NUMBER	22
5	SIRKET	VARCHAR2	20
6	ULKE_ADI	VARCHAR2	30
7	CINSIYET	VARCHAR2	20