
SQL

Structured Query Language
Yapılandırılmış Sorgu Dili

20 Ocak 2021
DERS 09

HAVING CLAUSE
UNION, UNION ALL,
INTERSECT, MINUS

Mehmet Bulutluoz
Elektronik muh.

Önceki Dersten Hatırladıklarımız

1- EXISTS : subquery'de EXISTS(istenenSart) şeklinde kullanılır, istenen sarta uyan değer(ler) için ilk sorgunun çalışmasını sağlar. IN komutunun kullanımına benzer ancak farklılıkları var

- IN komutundan önce bir field ismi yazmak zorundayız ancak EXISTS için field ismi yazılmaz
- IN komutundan önce hangi field ismi yazılmissa IN komutundan sonraki parantezin o field'a ait sonuçları getirmesi gerekir ancak EXISTS için böyle bir sınırlama yoktur
- IN komutu için olumsuz durum kullanımı yoktur ama NOT EXISTS kullanılır. Bunun için sorgu olumlu duruma göre yazılır ancak ilk sorgu SARTI SAGLAMAYAN değerler için çalışır

2- IS NULL : WHERE satırında kullanılır ve sorgulama sonucunda istenilen field'daki boş olan kayıt bilgilerini getirir.

IS NOT NULL : boş olmayan kayıtları getirir

Önceki Dersten Hatırladıklarımız

3- ORDER BY CLAUSE : Sorgu sonucunun istedigimiz bir veya birden fazla fielda göre sıralı olarak gelmesini sağlar.

- Eğer sıralama turunu belirtmezsek default olarak NATURAL ORDER şeklinde sıralama yapar
- İstersek ASCENDING veya DESCENDING olarak sıralama yapabiliriz . Eğer birden fazla field'a göre sıralama yapacaksa önce ilk sıralamayı yapar, sonra ilk sıralamada aynı olan değerleri 2. sıralama kriterine göre sıralar

4- ALIASES : Tablo yapısını degistirmeden sorguda gelen fieldlara farklı isimlendirme yapmamıza imkan tanır

- AS : AS kullandıktan sonra istedigimiz ismi yazabiliriz
- || : ile CONCATINATE yapabiliriz. || bir field'a sabit bir string ekleyebilir veya 2 field'ı birleştirebiliriz

5- GROUP BY: Sorgu sonuclarini belli field'lara göre gruplandirmamiza imkan tanir.

- GROUP BY kullandigimizda sorguda gruplandirma yaptigimiz field'in olmasina dikkat etmeliyiz
-

HAVING CLAUSE

HAVING, AGGREGATE FUNCTION'lar ile birlikte kullanılan FİLTRELEME komutudur.

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

```
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(456789012, 'Veli Sahin', 'Ankara', 4500, 'Ford');  
INSERT INTO personel VALUES(123456710, 'Hatice Sahin', 'Bursa', 4500, 'Honda');
```

1) Her sirketin **MIN** maaslarini eger 2000'den buyukse goster

```
SELECT sirket, MIN (maas) AS en_az_maas  
FROM personel  
GROUP BY sirket  
HAVING MIN (maas) >2000;
```

SIRKET	EN_AZ_MAAS
Honda	3500
Ford	4500
Toyota	4500
Tofas	7000

HAVING CLAUSE

2) Ayni isimdeki kisilerin aldigi toplam gelir 10000 liradan fazla ise ismi ve toplam maasi gosteren sorgu yaziniz

```
SELECT isim, SUM (maas) AS toplam_maas  
FROM personel  
GROUP BY isim  
HAVING SUM (maas) >10000;
```

ISIM	TOPLAM_MAAS
Mehmet Ozturk	16500

3) Eger bir sehirde calisan personel sayisi 1'den coksa sehir ismini ve personel sayisini veren sorgu yaziniz

```
SELECT sehir, COUNT (isim) AS toplam_personel_sayisi  
FROM personel  
GROUP BY sehir  
HAVING COUNT (isim) >1;
```

SEHIR	TOPLAM_PERSONEL_SAYISI
Istanbul	2
Ankara	3

HAVING CLAUSE

4) Eger bir sehirde alinan MAX maas 5000'den dusukse sehir ismini ve MAX maasi veren sorgu yaziniz

```
SELECT sehir, MAX (maas) AS max_maas  
FROM personel  
GROUP BY sehir  
HAVING MAX (maas) <5000;
```

SEHIR	MAX_MAAS
Bursa	4500

TECHPROED

UNION OPERATOR

İki farklı sorgulamanın sonucunu birleştiren işlemidir. Seçilen **Field SAYISI** ve **DATA TYPE**'i aynı olmalıdır.

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

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

1) Maası 4000'den çok olan işçi isimlerini ve 5000 liradan fazla maaş alan şehirleri gösteren sorguyu yazınız

```
SELECT sehir ASisci_veya_sehir_ismi ,maas
FROM personel
WHERE maas >5000
UNION
SELECT isim ASisci_veya_sehir_ismi , maas
FROM personel
WHERE maas > 4000;
```

ISCI_VEYA_SEHIR_ISMI	MAAS
Ali Yilmaz	5500
Ankara	4500
Ankara	7000
Bursa	4500
Hatice Sahin	4500
İstanbul	4500
İstanbul	5500
İzmir	6000
Mehmet Ozturk	6000
Mehmet Ozturk	7000
Veli Sahin	4500
Veli Sahin	4500

UNION OPERATOR

2) Mehmet Ozturk ismindeki kisilerin aldigi maaslari ve Istanbul'daki personelin maaslarini bir tabloda gosteren sorgu yaziniz

```
SELECT sehir AS isci_veya_sehir_ismi ,maas
FROM personel
WHERE sehir='Istanbul'
UNION
SELECT isim AS isci_veya_sehir_ismi , maas
FROM personel
WHERE isim = 'Mehmet Ozturk';
```

ISCI_VEYA_SEHIR_ISMI	MAAS
Istanbul	4500
Istanbul	5500
Mehmet Ozturk	3500
Mehmet Ozturk	6000
Mehmet Ozturk	7000

NOT : 2.sorgunun sonuna ORDER BY komutunu kullanirsaniz tum tabloyu istediginiz siralamaya gore siralar

```
ORDER BY maas;
```

ISCI_VEYA_SEHIR_ISMI	MAAS
Mehmet Ozturk	3500
Istanbul	4500
Istanbul	5500
Mehmet Ozturk	6000
Mehmet Ozturk	7000

UNION OPERATOR

3) Sehirlerden odenen ucret 3000'den fazla olanlari ve personelden ucreti 5000'den az olanlari bir tabloda maas miktarina gore sirali olarak gosteren sorguyu yaziniz

```
SELECT sehir AS isci_veya_sehir_ismi , maas
FROM personel
WHERE maas>3000
UNION
SELECT isim AS isci_veya_sehir_ismi , maas
FROM personel
WHERE maas<5000;
```

ISCI_VEYA_SEHIR_ISMI	MAAS
Ankara	3500
Ankara	4500
Ankara	7000
Bursa	4500
Hatice Sahin	4500
Istanbul	4500
Istanbul	5500
Izmir	6000
Mehmet Ozturk	3500
Veli Sahin	4500
Veli Sahin	4500

UNION OPERATOR

2 Tablodan Data Birleştirme

Personel isminde bir tablo oluşturun. İçinde id, isim, şehir, maaş ve şirket alanları olsun. Id'yi 2. yöntemle PK yapın

```
CREATE TABLE personel
```

```
(
  id number(9),
  isim varchar2(50),
  şehir varchar2(50),
  maaş number(20),
  şirket varchar2(20),
  CONSTRAINT personel_pk PRIMARY KEY (id)
);
```

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

Personel_bilgi isminde bir tablo oluşturun. İçinde id, tel ve çocuk sayısı alanları olsun. Id'yi FK yapın ve personel tablosu ile ilişki kurun

```
CREATE TABLE personel_bilgi
```

```
(
  id number(9),
  tel char(10) UNIQUE ,
  çocuk_sayısı number(2),
  CONSTRAINT personel_bilgi_fk FOREIGN KEY (id) REFERENCES personel(id)
);
```

```
INSERT INTO personel_bilgi VALUES(123456789, '5302345678', 5);
INSERT INTO personel_bilgi VALUES(234567890, '5422345678', 4);
INSERT INTO personel_bilgi VALUES(345678901, '5354561245', 3);
INSERT INTO personel_bilgi VALUES(456789012, '5411452659', 3);
INSERT INTO personel_bilgi VALUES(567890123, '5551253698', 2);
INSERT INTO personel_bilgi VALUES(456789012, '5524578574', 2);
INSERT INTO personel_bilgi VALUES(123456710, '5537488585', 1);
```

UNION OPERATOR

id'si 12345678 olan personelin Personel tablosundan sehir ve maasini, personel_bilgi tablosundan da tel ve cocuk sayisini yazdirin

```
SELECT sehir AS Sehir_tel ,maas AS cocuk_sayisi_veya_maas
FROM personel
WHERE id='123456789'
```

UNION

```
SELECT tel,cocuk_sayisi
FROM personel_bilgi
WHERE id= '123456789';
```

SEHIR_TEL	COCUK_SAYISI_VEYA_TEL
5302345678	5
Istanbul	5500

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
456789152	Veli Sahin	Ankara	4500	Ford
123456710	Hatice Sahin	Bursa	4500	Honda

ID	TEL	COCUK_SAYISI
123456789	5302345678	5
234567890	5422345678	4
345678901	5354561245	3
456789012	5411452659	3
567890123	5551253698	2
456789012	5524578574	2
123456710	5537488585	1

NOT : Union islemi yaparken

- 1)Her 2 QUERY'den elde edeceginiz tablolarin sutun sayilari esit olmalı
- 2)Alt alta gelecek sutunlarin data type'leri ayni olmalı

UNION ALL OPERATOR

1) Personel tablosundada maasi 5000'den az olan tum isimleri ve maaslari bulunuz

```
SELECT isim,maas  
FROM personel  
WHERE maas<5000;
```

ISIM	MAAS
Veli Sahin	4500
Mehmet Ozturk	3500
Veli Sahin	4500
Hatice Sahin	4500

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

TECHPROED

UNION ALL OPERATOR

2) Ayni sorguyu UNION ile iki kere yazarak calistirin

```
SELECT sehir,maas  
FROM personel  
WHERE maas<5000
```

UNION

```
SELECT sehir,maas  
FROM personel  
WHERE maas<5000;
```

SEHIR	MAAS
Ankara	3500
Ankara	4500
Bursa	4500
Istanbul	4500

3) Ayni sorguyu UNION ALL ile iki kere yazarak calistirin

```
SELECT sehir,maas  
FROM personel  
WHERE maas<5000;
```

UNION ALL

```
SELECT sehir,maas  
FROM personel  
WHERE maas<5000;
```

SEHIR	MAAS
Istanbul	4500
Ankara	3500
Ankara	4500
Bursa	4500
Istanbul	4500
Ankara	3500
Ankara	4500
Bursa	4500

UNION ALL OPERATOR

UNION islemi 2 veya daha çok **SELECT** isleminin sonuc **KUMELERINI** birleştirmek için kullanılır, Aynı kayıt birden fazla olursa, sadece bir tanesini alır.

UNION ALL ise tekrarlı elemanları, tekrar sayısınca yazar.

NOT : UNION ALL ile birleştirmelerde de

- 1) Her 2 QUERY'den elde edeceğiniz tabloların sütun sayıları eşit olmalı
- 2) Alt alta gelecek sütunların data type'leri aynı olmalı

TECHPROED

UNION ALL OPERATOR

1) Tabloda personel maasi 4000'den cok olan tum sehirleri ve maaslari yazdirin

SEHIR	MAAS
Istanbul	5500
Istanbul	4500
Izmir	6000
Ankara	7000
Ankara	4500
Bursa	4500

SELECT sehir,maas
FROM personel
WHERE maas>4000;

2) Tabloda personel maasi 5000'den az olan tum isimleri ve maaslari yazdirin

ISIM	MAAS
Veli Sahin	4500
Mehmet Ozturk	3500
Veli Sahin	4500
Hatice Sahin	4500

SELECT isim,maas
FROM personel
WHERE maas<5000;

3) Iki sorguyu UNION ve UNION ALL ile birlestirin

ID	ISIM	SEHIR	MAAS	SIRKET
123456789	Ali Yilmaz	Istanbul	5000	Honda
234567890	Veli Sahin	Istanbul	5000	Toyota
345678901	Mehmet Ozturk	Ankara	4500	Honda
456789012	Mehmet Ozturk	Izmir	6000	Ford
567890123	Mehmet Ozturk	Ankara	6000	Tofas
456715012	Veli Sahin	Ankara	4500	Ford
123456710	Hatice Sahin	Bursa	4500	Honda

SEHIR	MAAS
Ankara	4500
Ankara	7000
Bursa	4500
Hatice Sahin	4500
Istanbul	4500
Istanbul	5500
Izmir	6000
Mehmet Ozturk	3500
Veli Sahin	4500

SEHIR	MAAS
Istanbul	5500
Istanbul	4500
Izmir	6000
Ankara	7000
Ankara	4500
Bursa	4500
Veli Sahin	4500
Mehmet Ozturk	3500
Veli Sahin	4500
Hatice Sahin	4500

INTERSECT OPERATOR

1) Personel tablosundan Istanbul veya Ankara'da calisanlarin id'lerini yazdir

ID
123456789
234567890
345678901
567890123
456715012

```
SELECT id
FROM personel
WHERE sehir IN ('Istanbul','Ankara');
```

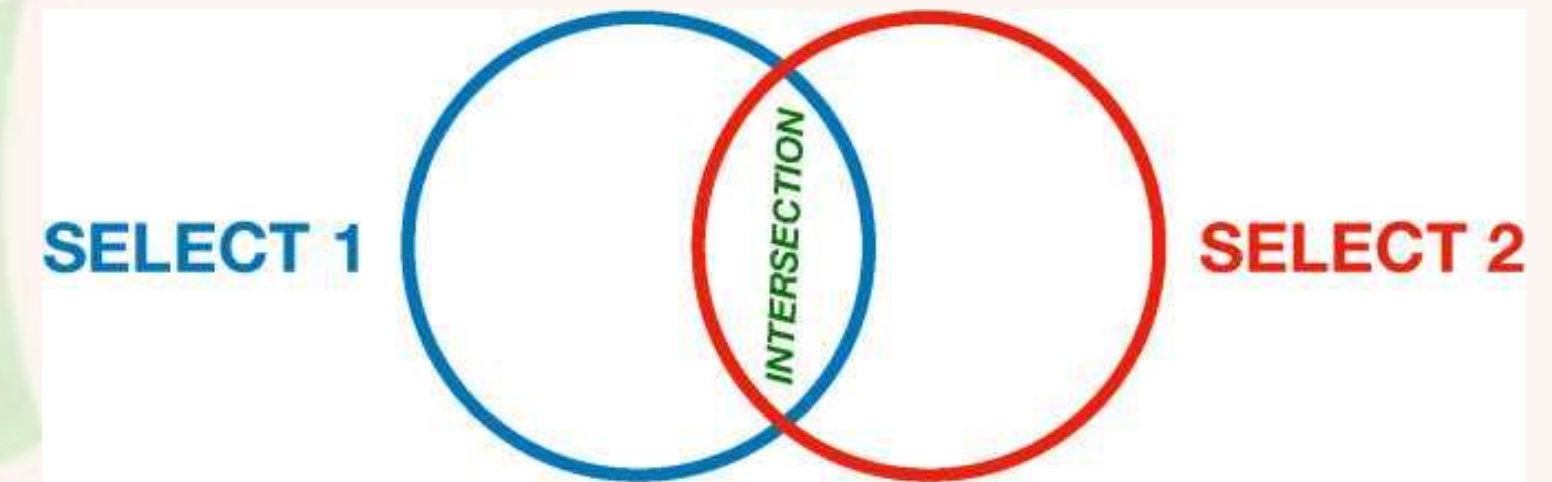
2) Personel_bilgi tablosundan 2 veya 3 cocugu olanlarin id lerini yazdirin

ID
345678901
456789012
567890123
456789012

```
SELECT id
FROM personel_bilgi
WHERE cocuk_sayisi IN (2,3);
```

3) Iki sorguyu INTERSECT ile birlestirin

ID
345678901
567890123



INTERSECT OPERATOR

1) Maasi 4800'den az olanlar veya 5000'den cok olanlarin id'lerini listeleyin

ID
234567890
345678901
456789012
567890123
456715012
123456710

```
SELECT id
FROM personel
WHERE maas NOT BETWEEN 4800 AND 5500;
```

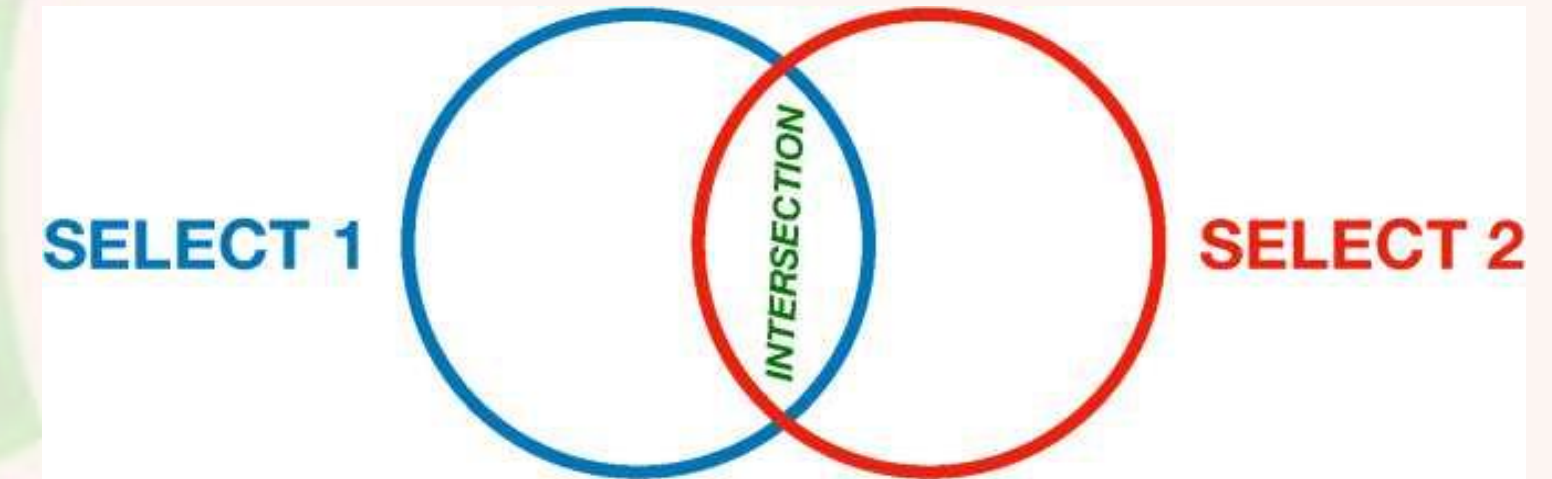
2) Personel_bilgi tablosundan 2 veya 3 cocugu olanlarin id lerini yazdirin

ID
345678901
456789012
567890123
456789012

```
SELECT id
FROM personel_bilgi
WHERE cocuk_sayisi IN (2,3);
```

3) Iki sorguyu INTERSECT ile birlestirin

ID
345678901
456789012
567890123



TECHNOLOGY

INTERSECT OPERATOR

3) Honda, Ford ve Tofas'ta çalışan ortak isimde personel varsa listeleyin

```
SELECT isim  
FROM personel  
WHERE sirket='Honda'
```

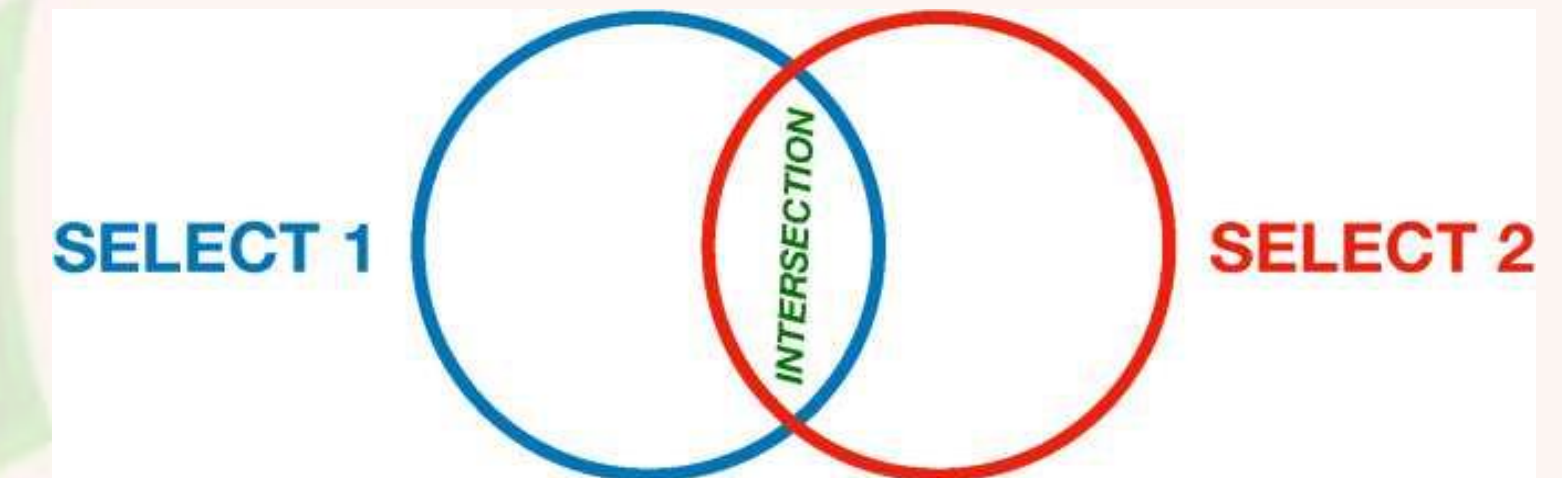
INTERSECT

```
SELECT isim  
FROM personel  
WHERE sirket='Ford'
```

INTERSECT

```
SELECT isim  
FROM personel  
WHERE sirket='Tofas';
```

ISIM
Mehmet Ozturk



TECH PRO

MINUS OPERATOR

1) 5000'den az maas alip Honda'da calismayanlari yazdirin

```
SELECT isim,sirket  
FROM personel  
WHERE maas<5000
```

MINUS

```
SELECT isim,sirket  
FROM personel  
WHERE sirket='Honda'
```

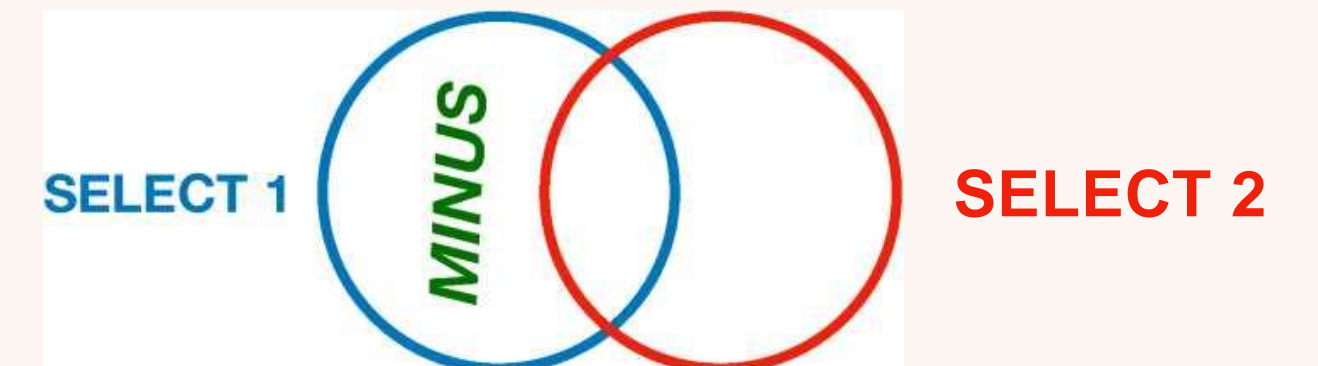
2) Ismi Mehmet Ozturk olup Istanbul'da calismayanlarin isimlerini ve sehirlerini listeleyin

```
SELECT isim,sehir  
FROM personel  
WHERE isim='Mehmet Ozturk'
```

MINUS

```
SELECT isim,sirket  
FROM personel  
WHERE sehir='Istanbul';
```

ISIM	SIRKET
Veli Sahin	Ford
Veli Sahin	Toyota



ISIM	SEHIR
Mehmet Ozturk	Ankara
Mehmet Ozturk	Izmir