

Veri Tabanı-SQL

23 Şubat 2023 Perşembe 02:24

ID = Örneğin kimlik numarası gibidir. Benzersiz bir değerdir. ID değeri otomatik olarak atanan bir değerdir. Genellikle 1 den başlar devam eder.

Bazı Veri Tipleri:

Bigint: büyük tam sayılar. Normal int daha büyük hali

Binary: 0 ile 1 lerden oluşan 2 lik sayı sisteminde kullanılıyor.

*****Bit**: 0 ile 1 lerden oluşur ama binary gibi değil (2 tane durumu olan ifadeler için kullanılır mesela evli mi bekar mı, askere gitmiş mi gitmemiş mi gibi..)

*****Char**: Karakterler için kullanılıyor. Sabit uzunluktaki karakterler için kullanılıyor. Herkeste boyutu aynı olan karakterler örnek tc herkeste 11 karakter mesela telefon numarası herkeste aynı formatta.. char(10) yani ilgili sütunun kaç karakterden oluşacağını gösteriyor. Mesela bunu yazıp ali dediğimizde ali 3 karakter yani 7 kere char boşluk atıyor. Bu da bellekte gereksiz yer işgal eder.

*****Decimal**: ondalıklı veri tipleri için kullanılır.

Decimal(x,y)

x: Virgülden Önce Basamak Sayısı

y: Virgülden Sonra Basamak Sayısı

örnek:

Ortalama: 78,65 düşünelim (4,2) oluyor

Ortalaması (100,00) olduğunu düşünelim decimal(5,2)

*****Float**: ondalıklı veri tipi decimal a göre daha az yer kaplar

Image: image veri tipi resim, video gibi dosyaları saklamak için kullanılan veri tipidir. Ama veri tabanlarında neredeyse hiç kullanılmaz nedeni ise veri tabanını çok fazla şişirir. Bunun yerine dosyalar sunucuya/harddisk e depolanır. Veri tabanında sadece onun doysa yolu tutulur.

*****int**: Tamsayılar için kullanılır.

*****nchar**: Yine sabit uzunluktaki karakterler için kullanılır. Başındaki n harfi latin albesi için kullanılıyor. Yani kore dilinde ya da arap dilinde bir ifade yazdığımızda eğer veri tabanına veri girişi olarak yazarsak veri tabanı bunu nchar kullanmadığımda ?? İşareti semboli atar. Nchar kullanırsak tam karşılığını alırız sorun olmaz. Char 'a göre hafızada iki kat yer kaplar.

*****Smalddatetime**: Tarih sorgularında sıkça kullanılır. Aralık olarak en uygun veri tipidir.

*****Smallint**: Küçük tam sayılar için (+ - 32650 aralığındadır)

*** **Tinyint**: 0 ile 255 aralığında küçük veriler için..

*****Varchar**: Değişken uzunluklu veriler için kullanılır yani ad için kullanılan en uygun veri tipi varchardır. Çünkü isimler değişken uzunluktaki verilerdir

DDL KOMUTLARI: (TABLO) "Veri tanımlama ifadesi"

1-Create --> oluşturma için:

```
CREATE TABLE TBIDERSLERR
(
  DERSID tinyint,
  DERSAD varchar(20)
)
```

tinyint kullanmamızın sebebi girebileceğimiz ders sayısının az olması

2-Alter --> Düzenleme için

```
--Create Table TBIDERSLERR
--(
--  DERSID tinyint,
--  DERSAD varchar(20)
--)

ALTER TABLE TBIDERSLERR
ADD KONTENJAAN smallint
```

Başka bir sutun eklemek için yorum satırına alıp alter ile kontenjaan adında sutun ekledik

3-Drop--> Silme İçin

```
--Create Table TBIDERSLERR
--(
--  DERSID tinyint,
--  DERSAD varchar(20)
--)

ALTER TABLE TBIDERSLERR
DROP COLUMN KONTENJAAN
```

Kontenjaan sütununu sildik

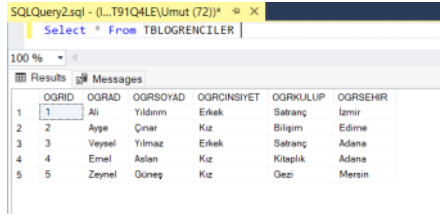
Tamamen tabloyu silmek için **drop table** dememiz gerekiyor.

DML KOMUTLARI (Veri)

1-Select--> Seçme, Listeleme

Kullanımı: Select * From Table * işareti --> All demek (tümünü seç) , From--> den dan anlamındadır

Not: SQL de sorgunun neresini seçersek orasını çalıştırır.



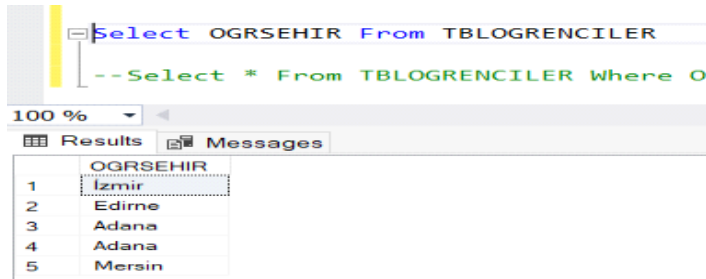
OGRID	OGRAD	OGRSOYAD	OGRCINSIYET	OGRKULUP	OGRSEHIR
1	Ali	Yıldırım	Erkek	Satranç	İzmir
2	Ayşe	Çınar	Kız	Bilgi	Edine
3	Veysel	Yılmaz	Erkek	Satranç	Adana
4	Emel	Aslan	Kız	Kitaplık	Adana
5	Zeynel	Güneş	Kız	Gezi	Mersin

Yukarıda ki görselde select*from TBLOGRENCİLER yaparak tüm öğrencileri listeledik.



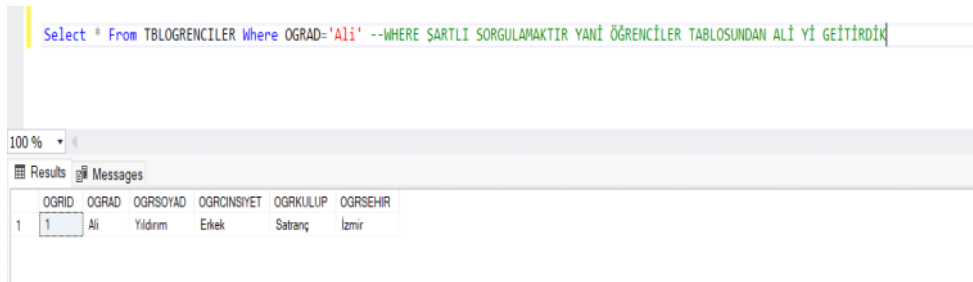
OGRAD	OGRSOYAD
Ali	Yıldırım
Ayşe	Çınar
Veysel	Yılmaz
Emel	Aslan
Zeynel	Güneş

Yukarıdaki görselde ise yıldız işareti (tümünü seç) yerine istediğimiz kısımları getirdik yani öğrenci adları ve soyadlarını listeledik.



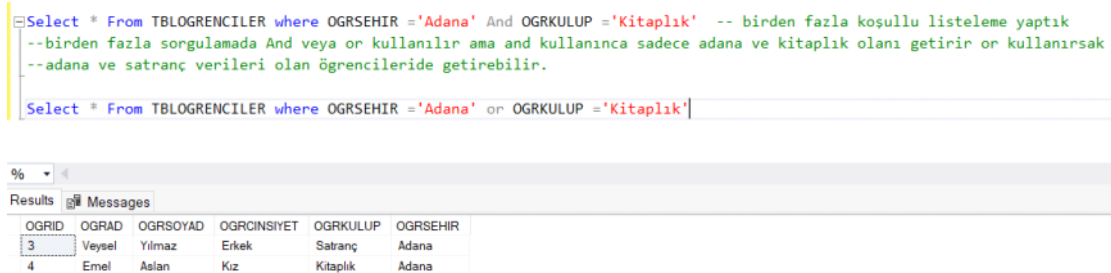
OGRSEHIR
İzmir
Edine
Adana
Adana
Mersin

Yukarıdaki görselde ise yıldız işareti (tümünü seç) yerine istediğimiz kısımları getirdik yani şehirleri listeledik.



OGRID	OGRAD	OGRSOYAD	OGRCINSIYET	OGRKULUP	OGRSEHIR
1	Ali	Yıldırım	Erkek	Satranç	İzmir

Yukarıdaki görselde ise "Where" ile şartlı listeleme yaptık. Yani öğrenciler tablosundan ismi Ali olanları getirdik.



OGRID	OGRAD	OGRSOYAD	OGRCINSIYET	OGRKULUP	OGRSEHIR
3	Veysel	Yılmaz	Erkek	Satranç	Adana
4	Emel	Aslan	Kız	Kitaplık	Adana

2-Insert--> Ekleme demek

Kullanımı:

Insert Into Table (Column1,Column2,Column3,...) Values (V1,V2,V3,)

```
Insert Into TBLOGRENCILER(OGRAD,OGRSOYAD,OGRSEHIR) Values ('Baran','Yücedağ','Elazığ')  
--Select * From TBLOGRENCILER
```

OGRID	OGRAD	OGRSOYAD	OGRCINSIYET	OGRKULUP	OGRSEHIR
1	Ali	Yıldırım	Erkek	Satranç	İzmir
2	Ayşe	Çınar	Kız	Bilgim	Edirne
3	Veysel	Yılmaz	Erkek	Satranç	Adana
4	Emel	Aslan	Kız	Kitaplık	Adana
5	Zeynel	Güneş	Kız	Gezi	Mersin
6	Baran	Yücedağ	NULL	NULL	Elazığ

Yukarıda insert sayesinde yeni bir öğrenci ekledik öğrencinin cinsiyet ve kulübünü sonra ekleyeceğiz

```
Insert Into TBLNOTLAR (OGRENCI,DERS,SINAV1,SINAV3) values (6,2,88,67)
```

Yukarıda ise 6 numaralı yani ID olan öğrenci için bazı sınav notlarının girişini yaptık

Oğrenci= 2 numaralı öğrenci

Ders= 2 numaralı ders

1.Sınav 88 2. Sınav 67 e denk geliyor

3-Delete --> Silme Kullanımı:

```
Delete From TBLDERSLER --Tüm Dersleri Siler  
Delete From TBLDERSLER where DERSID =6 --Ders id si 6 olanı siler  
Delete From TBLDERSLER where DERSAD = 'Algoritma' -- Dersini Siler
```

4-Truncate: Tabloyu sıfırlar

5-Update --> Güncelleme

```
Update TBLOGRENCILER Set OGRCINSIYET='Erkek' -- Tüm öğrencilerin cinsiyetini erkek yapar!!! Sakın umut böyle bi mallık yapmamalısın  
Update TBLOGRENCILER Set OGRCINSIYET='Kız' Where OGRID= 4 -- 4 numaralı öğrencinin cinsiyetini kız yaptık  
Update TBLOGRENCILER Set OGRCINSIYET='Erkek' Where OGRID= 6 -- 6 numaralı öğrencinin cinsiyetini kız yaptık
```

```
Update TBLOGRENCILER Set OGRKULUP='Kitaplık' Where OGRID= 6 --6 id li öğrenciye kulup atadık
```

```
-- Update TBLNOTLAR set ORTALAMA=(SINAV1+SINAV2+SINAV3)/3 --SINAV NOTLARINI TOPLATIP ORTALAMASINI ALDIRDIK
```

```
-- Update TBLNOTLAR set ORTALAMA=(SINAV1+SINAV2+SINAV3)/3 --SINAV NOTLARINI TOPLATIP ORTALAMASINI ALDIRDIK
UPDATE TBLNOTLAR set DURUM=1 where ORTALAMA>=50 -- ortalaması 50 üzeri olanlara true değeri yazdırdık
UPDATE TBLNOTLAR set DURUM=0 where ORTALAMA<50 -- ortalaması 50 den aşağı gelenlere false değeri yazdırdık
```

10 %

Messages

(1 row affected)

Hazır Fonksiyonlar: Bazı aritmetik işlemler için uzun uzun formülü yazmak yerine sadece fonksiyonu çağırarak işlemleri kolayca yapabiliriz

```
Select COUNT(*) From TBLOGRENCILER --ogrenciler tablosundaki kaç tane ogrenci varsa getirdik
Select COUNT(*) As 'Toplam Kayıt' From TBLOGRENCILER -- tablo başlığının adını "toplam kayıt" yaptık
Select COUNT(*) As 'Toplam Kişi' From TBLOGRENCILER where OGRSEHIR='Adana' -- adanada olan toplam kişi sayısı
```

1 %

Results Messages

(No column name)

6

Toplam Kayıt

6

Toplam Kişi

2

- İşareti tümünü getirir.

```
Select SUM(Sinav1),SUM(Sinav2),SUM(Sinav3) From TBLNOTLAR -- 6 tane ogrencilerimizin 1,2 ve 3. sınav notlarının toplamı
Select AVG(Sinav1), AVG(Sinav2), AVG(Sinav3) From TBLNOTLAR -- AVG ile 6 öğrencinin 1,2 ve 3. sınavlarının ortalamasını gösterdik
Select MAX(Sinav1),MAX(Sinav2),MAX(Sinav3) From TBLNOTLAR -- 6 öğrencinin 1.2. ve 3. sınavlarının en yüksek notunu gösterdik
Select MIN(Sinav1),MIN(Sinav2),MIN(Sinav3) From TBLNOTLAR -- 6 öğrencinin 1.2. ve 3. sınavlarının en düşük notunu gösterdik
```

100 %

Results Messages

	(No column name)	(No column name)	(No column name)
1	456	380	370
1	76	63	61
1	88	90	88
1	45	35	15

Gruplandırma: (group by ile yapılır.)Bir tabloda bulunan ilgi alanların kaçar tane olduğu belirtir. Mesela öğrencilerin kullüplere göre dağılımları veya kız erkek öğrenci sayıları, adana oturan öğrenci sayıları (verileri grupluyor iste)

```
Select OGRSEHIR, COUNT(*) As 'Toplam' From TBLOGRENCILER GROUP BY OGRSEHIR -- her bir şehirden kaç ogrenci olduğunu gördük
Select OGRCINSIYET, COUNT(*) As 'Toplam' From TBLOGRENCILER GROUP BY OGRCINSIYET
Select Durum, COUNT(*) From TBLNOTLAR GROUP BY Durum -- Derten geçen ve kalan ogrenci sayıları 0 false 1 true
```

100 %

Results Messages

	OGRSEHIR	Toplam
1	Adana	2
2	Edirne	1
3	Elazığ	1
4	Izmir	1
5	Mersin	1

	OGRCINSIYET	Toplam
1	Erkek	3
2	Kız	3

	Durum	(No column name)
1	0	1
2	1	5

Query executed successfully

\\pcsl\SQL\EXPRESS (16.0.BTM) DESKTOP-T817M

```
Select OGRKULUP, COUNT(*) From TBLOGRENCILER GROUP BY OGRKULUP -- ogrenci kuluplerinde kaç kişi var oldugunu gösterdik
```

%

Results Messages

OGRKULUP	(No column name)
Bilgim	3
Gezi	2
Kitaplık	2
Satranç	3

HAVING:

```
Select OGRSEHIR, COUNT(*) AS 'SAYI' FROM TBLOGRENCILER GROUP BY OGRSEHIR HAVING COUNT(*) =2 --GRUBLAMA YAPARKEN WHERE DEĞİL HAVING KULLANILIR  
--BURADA ŞEHİRLERDE 2 KİŞİ OLANLARI ÇAĞIRDIK 1 YAZSAYDIK 1 KİŞİ OLAN ŞEHİRLERİ GETİRİRDİ
```

```
Select OGRKULUP, COUNT(*) From TBLOGRENCILER GROUP BY OGRKULUP HAVING OGRKULUP='Kitaplık' --ogrenci kulupleri içerisinde sadece kitaplık olanları çağırdık
```

```
Select Durum, COUNT(*) AS 'GEÇENLER' From TBLNOTLAR GROUP BY Durum HAVING DURUM='true' -- GEÇEN OGRENÇİ SAYISI
```

```
Select Durum, COUNT(*) AS 'KALANLAR' From TBLNOTLAR GROUP BY Durum HAVING DURUM='false' -- KALAN OGRENÇİ SAYISI
```

%

Results Messages

OGRSEHIR	SAYI
Adana	2
Edirne	2
Elazığ	2
Mersin	2

OGRKULUP	(No column name)
Kitaplık	2

Durum	GEÇENLER
1	8

Durum	KALANLAR
0	2

Query executed successfully.

(local)\SQLEXPRESS (16.0 RTM) | DESKTOP-T91Q4LE\Umut (74) | DbOgrenciNot | 00:00:00 | 7 rows

Ln 64 Col 1 Ch 1 INS