

İÇİNDEKİLER

SQLQuery1

- 3.2_VTOlusturma
- 3.3_Bolumsonu_VTIslemleri
- 4.1_TemelSQLKomutlariGiris
- 4.2_SelectKomutu
- 4.3_InsertKomutu
- 4.4_UpdateKomutu
- 4.5_DeleteKomutu
- 4.6_TruncateKomutu

SQLQuery2

- 4.7_WhereSartiKavrami
- 4.8_WhereSartiUygulama

SQLQuery3

- 4.9_AMD_OR_Operatorleri
- 4.10_AND_OPERATORU_UYGULAMA

SQLQuery4

- 4.11_OR_OPERATORU_UYGULAMA
- 4.12_DISTINCT_KOMUTU

SQLQuery5

- 4.13_ORDERBY_KOMUTU

SQLQuery6

SQLQuery7

SQLQuery8

- 5.6_GROUPBY_KULLANIMI3_BIRMAZANIN_GUNLERE_GORE_SATISLARI_SQL
- 5.7_GROUPBY_KULLANIMI4_BirGuneGoreMagazalarinSatisRaporu

SQLQuery9

- 5.8_GROUPBY_KULLANIMI5_SehirlerinAylaraGoreSatisRaporları

SQLQuery10

- 5.9_GROUPBY_KULLANIMI6_UrunKategorilerineGoreSatisRaporları

SQLQuery1:

--4.2_SelectKomutu

-- Tabloyu getirmek: SELECT

USE ETRADE

SELECT

[ID], [CUSTOMERNAME], [CITY], [BIRTHDATE],

[DISTRICT], [GENDER]

FROM

CUSTOMERS

--4.3_InsertKomutu

-- Tabloya yeni satır eklemek: INSERT

INSERT INTO CUSTOMERS --Tabloya yeni satırlar eklemek için kullanız.

(CUSTOMERNAME,CITY,BIRTHDATE,DISTRICT,GENDER) --Customer tablosuna ekleyeceğimiz stünları yazdık.

VALUES

('MEHMET ÖZAL','ANTALYA','1990-12-21','ALANYA','E')

--4.4_UpdateKomutu

-- Veriyi Güncelleme/Değiştirme: UPDATE

SELECT * FROM CUSTOMERS --sadece tabloyu görmek için yazdık.

--*NULL hiçbirsey yok manasındadır. Boşluk bir karakterdir ve aslında Null olarak nitelendirilmez.

UPDATE CUSTOMERS

SET NATION = '' --Nation stünunu boşluk ile değiştirdik

UPDATE CUSTOMERS

SET NATION = 'TR',AGE=35 --Tüm kişilerin Nation stünunu TR ile ve Age stünunu 35 ile değiştirdik.

SELECT DATEDIFF(YEAR,'1994-03-20','2020-01-01') --Yaşı hesaplamak için doğum tarihi ile rastgele bir tarih arasındaki yıl farkı

SELECT DATEDIFF(YEAR,'1994-03-20',GETDATE()) -- GETDATE() ile şimdi ki tarihi çağırılmış olduk ve güncel yaşı hesapladık.

--*Burda ise tüm kişiler için tek bir fonksiyonla yaşları hesaplayıp tabloya ekleyeceğiz:

UPDATE CUSTOMERS

SET NATION = 'TR',AGE=DATEDIFF(YEAR,BIRTHDATE,GETDATE()) --Nation stünunu TR ile ve Age stünunu 35 ile değiştirdik.

--*kişinin BIRTHDATE ile GETDATE yani doğum yaşı ile şimdiki yılı çıkartarak yaşıını herbir kişi için bulmuş olduk.

--4.5_DeleteKomutu

-- Veriyi Silme: DELETE

DELETE FROM CUSTOMERS

--* DELETE KOMUTU indexleri silmez bu sebeple önceden 1000 tane verimiz olduğunu varsayılahım.

--*1000 veriyi DELETE ile sildikten sonra yeni bir veri eklediğimizde 1001 den devam eder.Ancak TRUNCATE indexleri de SİLER!

--4.6_TruncateKomutu

-- Veriyi ve ID/Index'i Silme: TRUNCATE

--*Üstte tüm verileri silmiştim. Yeni bir müşteri ekleyelim

INSERT INTO

CUSTOMERS(CUSTOMERNAME,CITY,DISTRICT,BIRTHDATE,GENDER) VALUES ('Volkan ÇEKİP','Bartın','Kurucaşile','1997-10-24 00:00:00.000','E')

--*Tabloyu getirelim

Select * from CUSTOMERS

Delete from CUSTOMERS

TRUNCATE TABLE CUSTOMERS

SQLQuery2

-- 4.7_WhereSartiKavramiVeUygulama

--*WHERE kullanarak Sadece İstanbulda oturanları,Sadece Erkekleri,Belirli bir yaşıta olan verileri çektirmek isteyebiliriz.

SELECT * FROM CUSTOMERS --Tüm tabloyu getirmek içindi

--*Not:asağıdaki iki satırlık kodu aynı anda Excute(F5) etmemiz gerekir !

SELECT * FROM CUSTOMERS --Bunu WHERE ile şartlandırırsak şart ne ise ona göre veriyi getirir.

WHERE CUSTOMERNAME='Irmak TAHSİNOĞLU' --Sadece belirli bir kişiyi getirmek için

--*Not:asağıdaki iki satırlık kodu aynı anda Excute(F5) etmemiz gerekir !

--*Yada belirli bir şehirdekileri getirelim:

SELECT * FROM CUSTOMERS --Bunu WHERE ile şartlandırırsak şart ne ise ona göre veriyi getirir.

WHERE CITY='İSTANBUL' --Sadece belirli bir şehirdekileri getirmek için

-- <> ya da NOT Operatörü:

--*(Her ikiside Aynı işlevi görür)

--*Not:asağıdaki iki satırlık kodu aynı anda Excute(F5) etmemiz gerekir !

--*Belirttiğimiz şehir haricindekileri getirelim:

SELECT * FROM CUSTOMERS --Bunu WHERE ile şartlandırırsak şart ne ise ona göre veriyi getirir.

WHERE CITY<>'İSTANBUL' --İstanbul haricindeki şehirleri listeler/getirir.

-- NOT kullanımı:

SELECT * FROM CUSTOMERS --Bunu WHERE ile şartlandırırsak şart ne ise ona göre veriyi getirir.

WHERE NOT CITY='İSTANBUL' --İstanbul haricindeki şehirleri listeler/getirir.

--*Cinsiyeti erkek olanları getirelim:

SELECT * FROM CUSTOMERS

WHERE GENDER='E'

--*Doğum tarihine göre getirelim:

-- >,< Operatörleri(Büyükür-Küçükür operatörleri)

SELECT * FROM CUSTOMERS

WHERE BIRTHDATE>'1990-01-01' --*1990 Dan büyük olanları getirdik

--!!Tarihi bu şekilde de yazabiliz: **WHERE BIRTHDATE<'1990-01-01'**

--Not: 1990-01-01 ---> YılAyGün

-- >= <= Operatörleri

SELECT * FROM CUSTOMERS

WHERE BIRTHDATE>='1997-10-24' --*1997 ve 1997 den büyükleri getirir!

-- Belirli bir aralıktaki yaşa göre getirme/listeleme: BETWEEN
SELECT * FROM CUSTOMERS
WHERE BIRTHDATE BETWEEN '19971024' AND '19991102' --*1997 ve 1999 arasında doğan müşterileri getirir! 1997 ve 1999 dahildir !

--*Yaş ile işlem yapmak için yaş stünu boştu onu tekrar hesaplatalım müşterileri için:
UPDATE CUSTOMERS SET AGE=DATEDIFF(YEAR,BIRTHDATE,GETDATE())

--*Tablodaki yaşıları kontrol etme için tablodaki tüm değerleri getirelim:
SELECT * FROM CUSTOMERS

--!!22 ve 30 yaş arasındakiları listeleyelim:
SELECT * FROM CUSTOMERS
WHERE AGE BETWEEN 22 AND 30

--*İle başlayan,İle biten,içerenleri getirme getirme: LIKE
--*Belirli bir isim ile başlayanı getirelim:
SELECT * FROM CUSTOMERS
WHERE CUSTOMERNAME LIKE 'ALİ%' --Ali ile başlayanları getirdik

--*Belirli bir isim ile bitenleri getirelim:
SELECT * FROM CUSTOMERS
WHERE CUSTOMERNAME LIKE '%ALP' --ALP ile bitenleri getirdik

--*Ad ve soyad içerisinde bir harf ve kelime içerenleri getirelim:
SELECT * FROM CUSTOMERS
WHERE CUSTOMERNAME LIKE '%ALAN%' --Ad ve soyad içerisinde ALAN içerenleri getirdik

--*ismi belirttiğimiz bir harf veya kelime ile başlamayanları getirelim.
SELECT * FROM CUSTOMERS
WHERE CUSTOMERNAME NOT LIKE 'ALİ%' --ismi ALİ ile başlamayanları getirdik.

-- IN
SELECT * FROM CUSTOMERS
WHERE CITY IN ('İSTANBUL','ANKARA','İZMİR') --Şehir'i istanbul,izmir,ankara olanları getirdik.

-- WHERE İŞLEMİNİ BAŞKA KOMUTLARDADA KULLANMA

-- UPDATE İŞLEMİ İLE WHERE ŞARTININ KULLANIMI

--*Genel olarak kullanım aynıdır. Buraya kadar bir WHERE koşulunu SELECT işlemi için kullandık.UPDATE gibi diğer işlemlerde de kullanabiliriz.

--*Mesela cinsiyet olarak K ve E olarak tek bir char ile tutmuştu. Şimdi ise Kadın ve Erkek olarak yazdırıralım.

--*!!!Not :Ancak bunun için CUSTOMERS tablosuna sağ tıklayıp Desing tıkladıktan sonra GENDER deki veri türünü varchar(1) 'i 10 olarak değiştirmeyi unutma !

--*Ancak bunu değiştirdikten sonra kapat ve kayet de. Burda bir hata verecek !Bunu çözmek için: DataTypeChangeSetting.jpg ye bak ! yada video olarak 4.8_WhereSartiUygulama 10:00 dk da anlatımı mevcut !

--*Resime baktıktan sonra Tekrar varchar(1) 'i 10 olarak değiştir ve kaydet(HATIRLATMA: Kapatırken zaten kaytmek istiyorumusun diye soruyor !

UPDATE CUSTOMERS SET GENDER='ERKEK' WHERE GENDER='E' --E leri Erkek olarak değiştirdik !

--*Aynısını kadınlar için yapalım

UPDATE CUSTOMERS SET GENDER='KADIN' WHERE GENDER='K' --K leri KADIN olarak değiştirdik !

-- DELETE İŞLEMİ İLE WHERE ŞARTININ KULLANIMI

DELETE FROM CUSTOMERS WHERE ID=18 --*Sadece 18 ID olanı sildirdik

SQLQuery3

-- 4.9_AND_OR_Operatorleri

--*AND her iki şartın karşılaması gereklidir.Karşılamazsa False Döndürür.

--*OR En az bir şartı karşılaması gereklidir.Bir tane şartı karşılarsa True döndürür

-- 4.10_AND_OPERATORU_UYGULAMA

--Uygulama 1

SELECT * FROM CUSTOMERS

WHERE CITY='İSTANBUL' AND DISTRICT='Beylikdüzü' -- istanbul'da beylikdüzü'n'de ikmaet edenleri getirir.

--Uygulama 2

SELECT * FROM CUSTOMERS

WHERE CITY='İSTANBUL' AND DISTRICT='Beylikdüzü' AND DISTRICT='Esenler' -- istanbul'da hem beylikdüzü'n'de hem de Esenlerde ikmat edenler olmayacağından FALSE döndürür ve birşey getirmez

--Uygulama 3

SELECT * FROM CUSTOMERS

WHERE CITY='İSTANBUL'

AND GENDER='ERKEK' AND DISTRICT='ESENLER' -- istanbul'da Esenlerde ikmat eden Erkekleri getirir. Sadece 1 tane var !

--Uygulama 4

SELECT * FROM CUSTOMERS

WHERE CITY='İSTANBUL'

AND GENDER='ERKEK' AND BIRTHDATE BETWEEN '19900101' AND '20000101' -- istanbul'da 1990 ve 2000 arasında olan Erkekleri listeler

-- 4.11_OR_OPERATORU_UYGULAMA

--Uygulama 5

SELECT * FROM CUSTOMERS

WHERE CITY='İSTANBUL' OR CITY='İZMİR' OR CITY='BURSA' --İstanbul,izmir,bursa şehrinde ikamet edenlerin hepsini listeler.

--Uygulama 6

SELECT * FROM CUSTOMERS

WHERE CITY IN ('İSTANBUL','İZMİR','BURSA') --Uygulama 5'in aynısı bu şekilde de yazılabilir.

--Uygulama 7

SELECT * FROM CUSTOMERS

WHERE BIRTHDATE BETWEEN '19900101' AND '19981231' --Doğum tarihleri 1990 ve 1998 arasında olanlar

--Uygulama 8

SELECT * FROM CUSTOMERS

WHERE BIRTHDATE >='19900101' AND BIRTHDATE<='19981231' ---uygulama 7 nin ile aynı sonucu verir

--Uygulama 9

SELECT * FROM CUSTOMERS

WHERE NOT BIRTHDATE BETWEEN '19900101' AND '19981231' --Doğum tarihleri 1990
ve 1998 arasında olanları değil bu tarihler arasında OLMAYANLARI getirir

--Uygulama 10

SELECT * FROM CUSTOMERS

WHERE BIRTHDATE <='19900101' AND BIRTHDATE>='19981231' ---uygulama 9 ile aynı
sonucu verir

SQLQUERY4

-- 4.12_DISTINCT_KOMUTU

--Tekrar eden satırları tekilleştirmek için kullanılan komuttur

--Benim tabirimle 1 veriyi tekrarsız bir şekilde listeler !

--NOT: Bu arada tablomuzdaki DISTRICT(Bölge) ile kesinlikle karıştırmayalım ! :D

--Uygulama 1

--Türkiye'nin hangi illerinden müşterimiz var ona bakalım:

--NOT:aynı ili sadece bir kere getirir!

SELECT DISTINCT CITY FROM CUSTOMERS

--Uygulama 2

SELECT CITY FROM CUSTOMERS --Tüm şehirler listeler ve birden fazla varsa onları da listeler!

--Uygulama 3

SELECT CITY,DISTRICT FROM CUSTOMERS --İstanbul şehrinin tabloda ki tüm bölgelerini(DISTRICT) listeler. Birden fazla aynı bölgeden varsa onlarında listeler.

WHERE CITY='İSTANBUL'

--Uygulama 4

SELECT DISTINCT CITY,DISTRICT FROM CUSTOMERS --İstanbul şehrinin tabloda ki tüm bölgelerini(DISTRICT) aynı bölgeden sadece BİR tanesini listeler.

WHERE CITY='İSTANBUL'

--Uygulama 5

SELECT DISTINCT GENDER FROM CUSTOMERS --İstanbul şehrinde hangi cinsiyetten müşterilerimiz var. Tabi DISTINCT kullandığımız için sadece iki cinsiyet olması sebebiyle en fazla 2 satır getirebilir. Sonuçtada her iki cinstende müşterimiz varmış !

WHERE CITY='İSTANBUL'

--Uygulama 6

--İstanbuldaki müşterilerimin yaşlarını() listeleyelim. DISTINCT methodunu kullandığımız için aynı yaşı birden fazla listelemez !

SELECT DISTINCT AGE FROM CUSTOMERS

WHERE CITY='İSTANBUL'

SQLQUERY5

--4.13_ORDERBY_KOMUTU

--Yaşına,adına,sisteme eklenme sırasına göre sıralı bir şekilde getirebiliriz.

--İşte bu tür sıralama işlemlerini kullandığımız komutun adı ORDER BY komutudur.

--ASC --> A dan Z de ya da küçükten büyüğe göre sıralama

--DESC --> Z dan A de ya da büyükten küçüğe göre sıralama (Harf olarak Z A'dan büyüktür mantığı ile sıralıyor.)

--

--Uygulama 1

SELECT * FROM CUSTOMERS

ORDER BY ID ASC --id si küçükten büyüğe doğru getirir. (1 2 3... şeklinde)

--Uygulama 2

SELECT * FROM CUSTOMERS

ORDER BY ID DESC --id si büyükten küçüğe doğru getirir. (1000 999 998... şeklinde)

--Uygulama 3

SELECT * FROM CUSTOMERS

ORDER BY ID --Hiç birşey yazmasam default(Varsayılan) değeri ASC dir.

--Uygulama 4

SELECT * FROM CUSTOMERS

ORDER BY CUSTOMERNAME --Defaultu ASC olduğu için müşteri isimlerini A B C... şeklinde sıralar.

--Uygulama 5

SELECT * FROM CUSTOMERS

ORDER BY CUSTOMERNAME DESC -- müşteri isimlerini Z Y V... şeklinde sıralar.

--Uygulama 6

SELECT * FROM CUSTOMERS

ORDER BY BIRTHDATE DESC -- müşteri doğum tarihlerine göre sıralar 1999 1998 1997 şeklinde sıralar

--Uygulama 7

SELECT * FROM CUSTOMERS

ORDER BY CITY -- müşterinin şehirlerini A dan Z ye sıralar. Aynı şehirdekilerin sırasını ise ID ile sıralar. Mesela Adana şehrindeki en başta en büyük ID olan ile başlar.

--Uygulama 8

--Şimdi da şehri aynı olanları müşteri isimlerine göre sıralayarak getirelim.

SELECT * FROM CUSTOMERS

ORDER BY CITY,CUSTOMERNAME

--OUTPUT:Aşağıda görüldüğü gibi Şehir ismine göre sıraladık. Birde aynı şehirdekileri de kendi arasında müşteri ismine göre sıraladık.(CUSTOMERNAME)

--Azad Batuhan ... şeklinde

--OUTPUT:

ID	CUSTOMERNAME	CITY	BIRTHDATE	DISTRICT	GENDER	NATION	AGE
1	Azad ÖNÜR AŞÇI	Adana	1989-03-23	Karataş	ERKEK	NULL	32
2	Batuhan İNCEDAYI	Adana	1983-03-19	Seyhan	ERKEK	NULL	38
3	Birsen İSFEN	Adana	1950-01-25	Pozantı	K	NULL	71
4	Cemal KILAVUZ	Adana	1999-10-12	Yüreğir	ERKEK	NULL	22
5	Çağla SALONU	Adana	1999-04-04	Yüreğir	K	NULL	22
6	Çığdem SEVENCAN	Adana	1948-07-23	Karataş	K	NULL	73
7	Emir SEZİK	Adana	1975-01-18	Sançam	ERKEK	NULL	46
8	Emrah TOPALAN	Adana	1944-09-25	Ceyhan	ERKEK	NULL	77
9	Gülsüm BİKEÇ	Adana	1970-07-22	Seyhan	K	NULL	51
10	Hayrettin İLTAR	Adana	1959-12-23	Tufanbeyli	ERKEK	NULL	62
11	Helin ÖZMEN	Adana	1967-07-28	İmamoğlu	K	NULL	54
12	Kezban TEM.MAD.SAN.TİC.LTD	Adana	1946-11-23	Sançam	K	NULL	75

--Uygulama 9

SELECT * FROM CUSTOMERS

ORDER BY CITY,CUSTOMERNAME DESC --Şehir müşteri ismine göre sıraladık. Birde Aynı şehirdekileri de kendi arasında müşteri ismine göre sıraladık.(CUSTOMERNAME) Z den A ya olacak ... şeklinde

--Uygulama 10

SELECT * FROM CUSTOMERS

ORDER BY CITY,DISTRICT,CUSTOMERNAME--İlk Şehire(CITY) göre sıralar. Aynı şehirdekiler Bölgeye(DISTRICT) göre sıralar. Aynı bölgedekileri müşteri ismine(CUSTOMERNAME) göre A dan Z ye göre sıralar

--OUTPUT:

	ID	CUSTOMERNAME	1	CITY	BIRTHDATE	2	DISTRICT	GENDER	NATION	AGE
1	617	Ehrah TOPALAN	1	Adana	1944-09-25	2	Ceyhan	ERKEK	NULL	77
2	526	Onur AKSARAY		Adana	1979-10-18		Çukurova	ERKEK	NULL	42
3	25	Muzaffer AĞAÇKESEN		Adana	1950-04-28		Feka	ERKEK	NULL	71
4	690	Helin ÖZMEN		Adana	1967-07-28		İmamoğlu	K	NULL	54
5	98	Azad ÖNÜR		Adana	1989-03-23		Karataş	ERKEK	NULL	32
6	789	Çigdem SEVENCAN		Adana	1948-07-23		Karataş	K	NULL	73
7	506	Birsen İSFEN		Adana	1950-01-25		Pozantı	K	NULL	71
8	196	Nisanur SARICAM		Adana	1950-06-11		Saimbeyli	K	NULL	71
9	492	Süleyman MOĞOLKANLI		Adana	1984-09-19		Saimbeyli	ERKEK	NULL	37
10	739	Emir SEZİK		Adana	1975-01-18		Sançam	ERKEK	NULL	46
11	203	Kezban TEM.MAD.SAN.TİC.LTD		Adana	1946-11-23		Sançam	K	NULL	75
12	842	Batuhan İNCEDAYI		Adana	1983-03-19		Seyhan	ERKEK	NULL	38

--Uygulama 11

```
SELECT * FROM CUSTOMERS
```

WHERE CITY='İSTANBUL'

ORDER BY CITY,DISTRICT,CUSTOMERNAME --Şehire(CITY) göre sıralar ama zaten sadece istabulu sıralattırdık(WHERE CITY='İSTANBUL').Daha sonra İstabulda kileri Bölgeye(DISTRICT) göre sıralar. Aynı bölgedekileri müşteri ismine(CUSTOMERNAME) göre A dan Z ye göre sıralar

--Uygulama 12

```
SELECT * FROM CUSTOMERS
```

WHERE CITY='İSTANBUL'

ORDER BY 1 --Birinci stüna göre yani ID ye göre sıralar...

--Yani ORDER BY 1 ile ORDER BY ID aynı manadadır

--Ya da ORDER BY 2 ile ORDER BY CUSTOMERSNAME aynı manadadır.

--OUTPUT:

1.Sıra	2. Sıra	3.	4.	5.	6.	7.	8.
ID	CUSTOMERNAME	CITY	BIRTHDATE	DISTRICT	GENDER	NATION	AGE
1	Rıkret DENİZALP	İstanbul	1995-03-17	Ümraniye	ERKEK	NULL	26
2	Hüseyin VOLKAN	İstanbul	1958-08-14	Esenler	ERKEK	NULL	63
3	Muhammed Emin SARIGOL	İstanbul	1985-06-26	Kağıthane	ERKEK	NULL	36
4	Sami ALEMDAROĞLU	İstanbul	1949-06-13	Avalar	ERKEK	NULL	72
5	Rümeysa İNCEDAL	İstanbul	1998-03-18	Beyikdüzü	K	NULL	23
6	Gönül ATILANEVLAT	İstanbul	1991-01-20	Adalar	K	NULL	30
7	Can TAŞER	İstanbul	1953-11-04	Bayampaşa	ERKEK	NULL	68
8	Saadet BALIMRE	İstanbul	1987-07-24	Beyikdüzü	K	NULL	34
9	Ufuk SARİDEMİR	İstanbul	1982-05-16	Gürgören	ERKEK	NULL	39
10	Lina BİRİNCİ	İstanbul	1960-08-02	Beyikdüzü	K	NULL	61
11	Reyhan KOVULMAZ	İstanbul	1995-10-05	Beşiktaş	K	NULL	26
12	Fatma TİCİ TD STİ	İstanbul	1941-09-18	Büyükköy	K	NULL	80

--Uygulama 13

```
SELECT * FROM CUSTOMERS
```

ORDER BY CITY ASC,CUSTOMERNAME DESC

--aynısı şu şekilde listelenir:

```
SELECT * FROM CUSTOMERS
```

ORDER BY 3 ASC,2 DESC --ORDER BY CITY,CUSTOMERNAME DESC ile aynı manadadır !

SQLQUERY6

--4.14_TOP_KOMUTU

Veri setimizden 1000 veriden 50 tanesini ya da %50 sini görmek için bu komutu kullanabiliriz

--Uygulama 1

SELECT

TOP 10 --ilk 10 veriyi getirdi

*** FROM CUSTOMERS**

--Uygulama 2

SELECT

TOP 10 PERCENT --ilk %10 veriyi getirdi !!

*** FROM CUSTOMERS**

--Uygulama 3

SELECT

TOP 10--isimleri sıralanmış listeden(ORDER BY CUSTOMERNAME) ilk 10 veriyi getirdi

--Denemek için istersen **TOP 10** u sil ve uygulama 3 ü çalıştır daha sonra o listedeki 10 satırı bak ve **TOP 10** yazıp çalıştır ve ilk 10 müşterinin geldiğini görebilirsin.

*** FROM CUSTOMERS**

ORDER BY CUSTOMERNAME

SQLQUERY7

--5.2_AGGREGATE_FUNCTIONS

```
/*
MIN(TOTALPRICE)
MAX()
COUNT()
SUM()
AVG()--Ortalama
*/
```

SELECT * FROM SALES

--Uygulama 1

SELECT COUNT (*) FROM SALES -- Satır sayısını verir

--Uygulama 2

SELECT * FROM SALES ORDER BY AMOUNT --Müşterileri arasında en az miktar satın alandan en çok miktar satın alanlara doğru listeler

--Uygulama 3

SELECT MIN(AMOUNT) FROM SALES --En az ürün alma miktarını getirir. minimum 1 tane ürün alınmış

--Uygulama 4

SELECT

MIN(AMOUNT),MAX(AMOUNT),COUNT(FICHENO),SUM(AMOUNT),AVG(AMOUNT)
FROM SALES

--MIN(AMOUNT) --> En az ürün alma miktarını

--MAX(AMOUNT) --> Max ürün alma miktarını

--COUNT(FICHENO) -->Toplam fiş miktarını aslında satış sayısını da gösterir. Çünkü her alımda fiş verdiğimiz için.

--SUM(AMOUNT) --> Toplam satış miktarı

--AVG(AMOUNT) -->1 satışta ortalama kaç adet satış yapmışım.(COUNT(FICHENO)/SUM(AMOUNT) ya denk geliyor)

--Uygulama 5

SELECT * FROM SALES ORDER BY TOTALPRICE --en düşük satış fiyatından en yüksek satış fiyatına doğru sıralar

--Uygulama 6

SELECT

MIN(TOTALPRICE),MAX(TOTALPRICE),COUNT(FICHENO),SUM(TOTALPRICE),AVG(TOTALPRICE) FROM SALES

--MIN(TOTALPRICE) --> minimum satış fiyatı

--MAX(TOTALPRICE) --> Max satış fiyatı

--COUNT(FICHENO) -->Toplam fiş miktarını aslında satış sayısını da gösterir. Çünkü her alımda fiş verdiğimiz için.

--SUM(TOTALPRICE) --> Toplam Ciro(Toplam satış fiyatı)

--AVG(TOTALPRICE) -->ortalama 1 müşteriye satılan toplam satış fiyatı

--Uygulama 7

--Burda da yukarıda ki min satış,max satış vs.. değerleri şehir bazında hesapladık. ADANA VE İSTANBUL u karşılaştırdık

SELECT

MIN(TOTALPRICE),MAX(TOTALPRICE),COUNT(FICHENO),SUM(TOTALPRICE),AVG(TOTALPRICE)

FROM SALES WHERE CITY='ADANA' -- adana şehiri için min satış,max satış,toplam ciro vs listele

SELECT

MIN(TOTALPRICE),MAX(TOTALPRICE),COUNT(FICHENO),SUM(TOTALPRICE),AVG(TOTALPRICE)

FROM SALES WHERE CITY='İSTANBUL'

--5.3_GROUPBY_KAVRAMI

--örneğin şehirlere göre toplam satışları listelemek için kullanabiliriz.

--!! Uygulama 7 deki 2 şehri karşılaştırmak için 2 ayrı kod yazdık.81 ili karşılaştırmak için ise 81 ayrı kod yazmamız gereklidir ama bunu GROUP BY ile kısaca halledebiliriz !!

--5.4_GROUPBY_KULLANIMI1

--Uygulama 8

--Burada her bir şehirdeki:

--MIN(TOTALPRICE)

--MAX(TOTALPRICE)

--COUNT(FICHENO)

--SUM(TOTALPRICE)

--AVG(TOTALPRICE) değerlerini görmek için GROP BY CITY yazdığımızda TÜM ŞEHİRLERİN min satış,max satış vs değerlerini getirir

```
SELECT  
CITY,  
MIN(TOTALPRICE) AS MINPRICE, --Geçici Stunları anlamlandırma(isimlendirmek) için AS  
MAXPRICE yazdık  
MAX(TOTALPRICE) AS MAXPRICE,  
COUNT(FICHENO)ROWCOUNT_, --toplam kaç kere satış yapmışım(şehir bazında)  
SUM(TOTALPRICE ) TOTALPRICE,  
AVG(TOTALPRICE) AVGPRICE  
FROM SALES
```

GROUP BY CITY --Üstteki değerleri(MIN(TOTALPRICE),... vs) şehir bazında hesapladık
ORDER BY CITY --şehirleri A dan Z ye göre listeledik

--Uygulama 9

--Bu arada sadece üsttekinden farklı olarak TOTAP PRICE a göre sıralama yaptırdık
SELECT

```
CITY,  
MIN(TOTALPRICE) AS MINPRICE,  
MAX(TOTALPRICE) AS MAXPRICE,  
COUNT(FICHENO)ROWCOUNT_,  
SUM(TOTALPRICE ) TOTALPRICE,  
AVG(TOTALPRICE) AVGPRICE  
FROM SALES
```

GROUP BY CITY

ORDER BY (TOTALPRICE) --uygulama 8 den tek farkı şehrre göre değilde
toplamciro(TOTALPRICE) ya göre listeledik

--Uygulama 10

--En çok satış yapan şubeden en az satış yapana göre listeleyelim

SELECT

```
CITY,SUM(TOTALPRICE)  
FROM SALES
```

GROUP BY CITY

ORDER BY SUM(TOTALPRICE) DESC

--OUTPUT(sonuca göre İstanbuldaki mağaza en çok satışa sahipmiş)

--Uygulama 11

--En çok satış yapan mağazadan en az satış yapana göre İLK 10 mağazayı listeleyelim.

SELECT TOP 10 --TOP 10 diyerek ilk 10 mağazayı listelemiş olcاز

```
CITY,SUM(TOTALPRICE)  
FROM SALES
```

```
GROUP BY CITY  
ORDER BY SUM(TOTALPRICE) DESC
```

--5.5_GROUPBY_KULLANIMI2_BIRMAZANIN_GUNLERE_GORE_SATISLARI1
(5.5_GROUPBY_KULLANIMI1)

--uygulama 12

--burada sadece eski tarihden yeni tarihe doğru Ankara'daki satışları listeledik

SELECT *

```
FROM SALES WHERE CITY='ANKARA'
```

```
ORDER BY DATE_
```

--uygulama 13

--Burada analiz gerçekleştireceğiz ancak ondan önce buradaki verimizde tarih ve saat DATE stünunda gözükmekte. Bu sebeple tarihleri saatten ayırmamız gerek.(HATIRLATMA:"2019-01-01 08:07:24.000" verilerimiz içinde tarih ve saat TARİH stününde ekliydi).

--Aslında burdaki amacımız eğer birgün bu şekilde bir veriyle karşılaşırsak bu iki veriyi nasıl ayırt ederiz buna çözüm üretmek için yaptım.Sırf bunlarla karşılaşırsak bunları nasıl ayırt ederiz ve CONVERT i ne amaçla kullanırız bunu anlatmak için Hoca saat ve tarih verilerini özellikle ayrı stünlarda tutmadığını vurguladı

```
SELECT CONVERT(TIME,'2019-01-01 08:07:24.000')-- tarih ve saatin aynı anda bulunduğu  
bu veriden sadece SAATİ getirdi
```

```
SELECT CONVERT(DATE,'2019-01-01 08:07:24.000')-- tarih ve saatin aynı anda bulunduğu  
bu veriden sadece TARİHİ getirdi
```

```
SELECT CONVERT(DATETIME,'2019-01-01 08:07:24.000')-- tarih ve saatin aynı anda  
bulunduğu bu veriden sadece TARİHİ VE SAATİ AYNI ANDA getirdi
```

--Uygulama 14

-- DATE2 diye bir alana sadece tarihleri atadık. (Bu alan tamamen geçicidir.Aşıl tabloya(Veri Tabanına)Kaydedilmez !)

```
SELECT CONVERT(DATE,DATE_) AS DATE2, *
```

```
FROM SALES WHERE CITY='ANKARA'
```

```
ORDER BY DATE_
```

--Uygulama 15

--Uygulama 14 de tamamen geçici olan DATE2 stünuna tarihleri eklemiştik ama şimdi bunu geçici değilde kalıcı bir şekilde tabloya ekleyelim:

--1. Bunun için ilk olarak dbo.SALES tablosuna desing ile DATE2 stünu ekledik.RESİMLİ ANLATIM:

1.dbo.SALES e sağ tıkla ve desing i seç

2.Daha sonra column name 'de sağ tıklayıp insert column dedikten sonra Column Name: DATE2 ve Data Type: date diyecek gelen uyarıları evet diyecek kaydet!

SELECT * FROM SALES -- SALES tablosuna DATE' stünunu görmek için tüm tabloyu getirdik(DATE2 stünü ilk başta NULL olarak getirir)

--Şimdi ise date 2 alanına tarihleri Convert edelim (Kalıcı olarak !)
UPDATE SALES SET DATE2=CONVERT(DATE,DATE_)

--tabloyu görmek/kontrol etmek için:
SELECT * FROM SALES

--Şimdi ise sadece belirli bir tarihi listeleyelim

--NOT:Normalde verimiz DATETIME şekilde tutulmuş.Yani tarih stünunda hem tarih hemde saat var.Ama biz tarihi DATE2 e atadık ve böylelikle saat yazmadan sadece tarih ile istediğimiz tarihi listeleyebildik

SELECT * FROM SALES

WHERE CITY='ANKARA' --hatırlatma: ankara şehrindekileri listeledik

--AND DATE2='2019-01-01' --sadece belirli tarihi listeledik --> önemli satır burası diğerleri sadece hatırlatma

ORDER BY DATE_ --hatırlatma: tarihe göre sıraladık

--Uygulama 16

--Bir mağazanın(Ankara) günlere göre satışını göstermek için ilk başta tabloyu getirelim:

SELECT * FROM SALES

WHERE CITY='ANKARA'

ORDER BY DATE_

--Ve çıktı ekranındaki listeyi kopyalayalım excell'e yapıştalım.Burdan sonraki kısımlar excell de yapıyoruz !

SQLQUERY8

--

5.6_GROUPBY_KULLANIMI3_BIRMAZANIN_GUNLERE_GORE_SATISLARI_SQL
--Excel de ki BIR MAGAZANIN GUNLERE GORE SATISLARI'ni burda uygulayalım:

--Uygulama 1

SELECT CITY, DATE2, SUM(TOTALPRICE) AS TOTALPRICE -- AS TOTALPRICE sadece geçici olarak stunu isimlendirir.

```
FROM SALES WHERE CITY='ANKARA'  
GROUP BY CITY,DATE2  
ORDER BY CITY,DATE2
```

--5.7_GROUPBY_KULLANIMI4_BirGuneGoreMagazalarinSatisRaporu
--Mesela Ocak aynın 1. gününde hangi mağaza ne kadar satmış onu analiz edeceğiz.

--Uygulama 2

```
SELECT DATE2,CITY,SUM(TOTALPRICE) AS TOTALPRICE
```

```
FROM SALES  
WHERE DATE2='2019-01-02'
```

```
GROUP BY DATE2,CITY  
ORDER BY DATE2,CITY
```

--Uygulama 3

```
SELECT DATE2,CITY,SUM(TOTALPRICE) AS TOTALPRICE
```

```
FROM SALES  
WHERE DATE2='2019-01-02'
```

```
GROUP BY DATE2,CITY  
ORDER BY DATE2,SUM(TOTALPRICE) DESC --Uygulama 3 den tek farkı toplam cirosun  
en fazladan en azı sıralama yaptık
```

SQLQUERY9

--5.8_GROUPBY_KULLANIMI5_SehirlerinAylaraGoreSatisRaporlari
--Uygulama 1 EXCEL(Sehirlerin Aylara Gore Satis Raporlari)

--Adım 1:

```
SELECT DATEPART(MONTH,'2019-08-02')#Sadace tarihte ki ayı listeler(8/ay)
```

--Adım 2:

```
UPDATE SALES SET MONTHNAME_='01.OCAK' WHERE  
DATEPART(MONTH,DATE2)=1  
UPDATE SALES SET MONTHNAME_='02.ŞUBAT' WHERE  
DATEPART(MONTH,DATE2)=2  
UPDATE SALES SET MONTHNAME_='03.MART' WHERE  
DATEPART(MONTH,DATE2)=3  
UPDATE SALES SET MONTHNAME_='04.NİSAN' WHERE  
DATEPART(MONTH,DATE2)=4  
UPDATE SALES SET MONTHNAME_='05.MAYIS' WHERE  
DATEPART(MONTH,DATE2)=5  
UPDATE SALES SET MONTHNAME_='06.HAZİRAN' WHERE  
DATEPART(MONTH,DATE2)=6  
UPDATE SALES SET MONTHNAME_='07.TEMMUZ' WHERE  
DATEPART(MONTH,DATE2)=7  
UPDATE SALES SET MONTHNAME_='08.AĞUSTOS' WHERE  
DATEPART(MONTH,DATE2)=8  
UPDATE SALES SET MONTHNAME_='09.EYLÜL' WHERE  
DATEPART(MONTH,DATE2)=9  
UPDATE SALES SET MONTHNAME_='10.EKİM' WHERE  
DATEPART(MONTH,DATE2)=10  
UPDATE SALES SET MONTHNAME_='11.KASIM' WHERE  
DATEPART(MONTH,DATE2)=11  
UPDATE SALES SET MONTHNAME_='12.ARALIK' WHERE  
DATEPART(MONTH,DATE2)=12
```

##Başlarına sayı yazmamızın sebebi aya göre sıralama yaparken başındaki numaralara göre sıralayacak.

--Adım3

```
SELECT * FROM SALES
```

--Buraya kadar olanlar adımları excel için yapmıştık. Şimdi ise SQL de yani burada yapalım:

--Uyglama 2 SQL(Sehirlerin Aylara Gore Satis Raporları)

--Şehirlerin aylara göre satışları

```
SELECT CITY,MONTHNAME_,SUM(TOTALPRICE) AS TOTALPRICE  
FROM SALES
```

```
GROUP BY CITY,MONTHNAME_  
ORDER BY CITY,MONTHNAME_
```

--Aylara göre şehirlerin satışları

```
SELECT MONTHNAME_,CITY,SUM(TOTALPRICE) AS TOTALPRICE  
FROM SALES
```

```
GROUP BY CITY,MONTHNAME_  
ORDER BY MONTHNAME_,CITY
```

--Aylara göre şehirlerin satışları ve O ayda en çok satış yapan şehirden en az satış yapan şehrre doğru sıralama yapılmasını istersek:

```
SELECT MONTHNAME_,CITY,SUM(TOTALPRICE) AS TOTALPRICE  
FROM SALES
```

```
GROUP BY CITY,MONTHNAME_  
ORDER BY MONTHNAME_,SUM(TOTALPRICE) DESC --Sadece CITY yerine  
SUM(TOTALPRICE) ekledik
```

SQLQUERY10

--5.9_GROUPBY_KULLANIMI6_UrunKategorilerineGoreSatisRaporlari

--Uygulama 1 EXCEL(UrunKategorilerineGoreSatisRaporlari)

```
SELECT * FROM SALES ORDER BY ID DESC
```

```
DELETE FROM SALES WHERE ID=115024
```

```
DELETE FROM SALES WHERE ID=115023
```

--Uygulama 2 SQL(UrunKategorilerineGoreSatisRaporlari)

```
SELECT CATEGORY1,CATEGORY2,CATEGORY3,CATEGORY4,  
SUM(TOTALPRICE) AS TOTALPICE,  
COUNT(*) AS ROWCOUNT_,  
SUM(AMOUNT) AS TOTALAMOUNT
```

FROM SALES

**GROUP BY CATEGORY1,CATEGORY2,CATEGORY3,CATEGORY4
ORDER BY CATEGORY1,CATEGORY2,CATEGORY3,CATEGORY4**

SQLQUERY11

--5.10_GROUPBY_KULLANIMI7_MagazalarinMusteriSayilariniBulma

--Magazalarin(aslında her şehir bir mağaza olarak algılıyoruz) müşteri sayılarını bulmak için:
--CITY ve FICHENO stünlarını kullanacağız.

--Uygulama 1

```
SELECT * FROM SALES  
WHERE MONTHNAME_ = '01.OCAK'  
AND CITY='ADANA'
```

--Uygulama 2

--Aslında bazen birden fazla barkod okutulduğunu varsayırsak ve bazı durumlarda birden fazla
Fiş kesildiğini düşünürsek

--ki aşağıdaki verilerden hareketle 000043 nolu fiş(FICHENO) bir sebepten dolayı birden fazla kez kesilmek zorunda kalmış. Bu sebeple asıl

	ID	FICHENO	DATE_	DATE2	MONTHNAME_	IT
1	291	0000000043	2019-01-22 11:19:10.000	2019-01-22	01.OCAK	95
2	293	0000000043	2019-01-11 11:19:10.000	2019-01-11	01.OCAK	23
3	295	0000000043	2019-01-24 11:19:10.000	2019-01-24	01.OCAK	15
4	296	0000000043	2019-01-01 11:19:10.000	2019-01-01	01.OCAK	94
5	855	0000000158	2019-01-07 13:34:29.000	2019-01-07	01.OCAK	2
6	857	0000000158	2019-01-04 13:34:29.000	2019-01-04	01.OCAK	19
7	1630	0000000314	2019-01-21 08:55:12.000	2019-01-21	01.OCAK	64
8	1670	0000000000	2019-01-21 08:55:12.000	2019-01-21	01.OCAK	64

```
SELECT  
CITY,FICHENO,COUNT(*) AS FISHSAYISI  
FROM SALES  
WHERE MONTHNAME_ = '01.OCAK'  
AND CITY='ADANA'
```

```
GROUP BY CITY,FICHENO
```

--Uygulama 3
-- Tüm şehirleri görelim

```
SELECT  
CITY,FICHENO,COUNT(*)  
FROM SALES  
WHERE MONTHNAME_ = '01.OCAK'  
AND CITY='ADANA'
```

```
GROUP BY CITY,FICHENO  
ORDER BY CITY --şehirleri A dan Z ye doğru sıralar
```

--Ama Adana da kesilen fişleri tek bir satılarda göstermek için yani tekilleştirmek için COUNT(DISTINCT FICHENO) kullandık:

```
SELECT  
CITY,COUNT(DISTINCT FICHENO),COUNT(*)  
FROM SALES  
WHERE MONTHNAME_ = '01.OCAK'
```

```
GROUP BY CITY  
ORDER BY CITY
```

--Uygulama 4
--Bir müşteri birden fazla gelmişse bunu da tekilleştirelim ki onu birden fazla kez saymayağım.

```
SELECT  
CITY,COUNT(DISTINCT CUSTOMERNAME) AS UNIQUECUSTOMER_TEKILMUSTERİ,  
COUNT(DISTINCT FICHENO) AS CUSTOMERCOUNT,  
COUNT(*) AS ITEMCOUNT  
FROM SALES  
WHERE MONTHNAME_ = '01.OCAK'
```

```
GROUP BY CITY  
ORDER BY CITY
```

SQLQUERY12(KONTORL ET)

--5.11_GROUPBY_KULLANIMI8_BelliBirCironunUzerindeSatisYapanMagazalariListeleme

--Uygulama 1

```
SELECT CITY,SUM(TOTALPRICE)  
FROM SALES  
GROUP BY CITY  
ORDER BY SUM(TOTALPRICE) DESC
```

--Uygulama 2

--Şimdi ise 40000 Den fazla olanları getirelim ancak bunu WHERE şartı ile yapamayız hata verir !
--NOT: aggregate function(SUM, COUNT, AVG, MIN, MAX) nunu [GROUP BY ile kullanmışsak--bunu kontrol et] WHERE şartında kullanamayız onun yerine HAVING ile kullanabiliriz:

```
SELECT CITY,SUM(TOTALPRICE)  
FROM SALES  
--WHERE SUM(TOTALPRICE)>40000 ---> Bu şekilde aggregate function(SUM, COUNT, AVG, MIN, MAX) nunu WHERE şartında kullanamayız !  
--onun yerine HAVING ile kullanabiliriz:  
GROUP BY CITY  
HAVING SUM(TOTALPRICE)>40000 --40000 den fazla olanları getirdik
```

```
ORDER BY SUM(TOTALPRICE) DESC
```

--Uygulama 3

--Bu uygulamada ise ANKARA,İSTANBUL,İZMİR, AYDIN,HATAY şehirleri arasından toplam fiyatı 40000 den az olanı getirelim:
SELECT CITY,SUM(TOTALPRICE)

```
FROM SALES  
WHERE CITY IN ('ANKARA','İSTANBUL','İZMİR','AYDIN','HATAY')  
GROUP BY CITY  
HAVING SUM(TOTALPRICE)<40000  
  
ORDER BY SUM(TOTALPRICE) DESC
```

--Uygulama 4

--Bu uygulamada ise şehirlere göre müşteri sayısı ve 40000 den fazla cirosu olanları listeleyelim:

SELECT CITY,SUM(TOTALPRICE) AS TOTALPRICE,

COUNT(DISTINCT CUSTOMERNAME) CUSTOMERCOUNT

FROM SALES

GROUP BY CITY

HAVING SUM(TOTALPRICE)>40000

ORDER BY SUM(TOTALPRICE) DESC

--Uygulama 5

--müşteri sayısı 500 den ve toplam cirosu 300000 den fazla olanı listeleyelim:

SELECT CITY,SUM(TOTALPRICE) AS TOTALPRICE,

COUNT(DISTINCT CUSTOMERNAME) AS CUSTOMERCOUNT

FROM SALES

GROUP BY CITY

HAVING COUNT(DISTINCT CUSTOMERNAME)>500 --müşteri sayısı 500 den fazla olan

AND SUM(TOTALPRICE)>300000 -- toplam cirosu 300000 den fazla olan şehirler

ORDER BY SUM(TOTALPRICE) DESC

SQLQUERNONE

--6.1_TamSayıVeriTipleri

SAYISAL VERİ TİPLERİ

bigint	Minimum: -2^63 (-9,223,372,036,854,775,808)	8 Byte	
	Maksimum: 2^63-1 (9,223,372,036,854,775,807)		
int	Minimum: -2^31 (-2,147,483,648)	4 Byte	müşteri sayımız 2 milyondan az ise bunu kullanabiliriz
	Maksimum: 2^31-1 (2,147,483,647)		
smallint	Minimum: -2^15 (-32,768)	2 Byte	mesela ürün tablosu tuttuğumuzda ürün sayısı 32000 i geçmiyorsa bunu kullanabiliriz.
	Maksimum: 2^15-1 (32,767)		
tinyint	Minimum: 0	1 Byte	Ülkeleri bu veri tipi ile tutabiliriz.
	Maksimum: 255		
bit	0 ya da 1 değerini alır.		Eğer tabloda 8 ya da daha az bit kolonu varsa 1 byte, 8'den fazla ise 2 byte yer kaplar.
			0 ve 1 değerini tutar yada cisiyeti bu veri tipiyle tutabiliriz. Kız ve Erkek şeklinde...

SQLQUERY13

--6.2_OndalıkSayıVeriTipleri

SAYISAL VERİ TİPLERİ

decimal/ numeric	Minimum: - 10^38 +1	Hassasiyetine göre diskte kapladığı alan değişir.
		1'den 9'a kadar Hassasiyet için: 5 byte
	Maksimum: 10^38 – 1.	10'dan 19'a kadar Hassasiyet için: 9 byte
		20'den 28'a kadar Hassasiyet için: 13 byte
		29'dan 38'e kadar Hassasiyet için: 17 byte
money	Minimum: -922,337,203,685,477.5808 Maksimum: 922,337,203,685,477.5807	8 Byte
smalldmoney	Minimum: - 214,748.3648	4 Byte
	Maksimum: 214,748.3647	
float	-1.79308 ile -2.23308, 0	7 basamağa kadar 4 Byte
	2.23308 ile 1.79308	15 basamağa kadar 8 Byte
Real	-3.438 ile -1.1838, 0	4 Byte
	1.1838 ile 3.438	

--6.3_MetinVeritipleri1_VarcharCharVeriTipleri

--Uygulama1

```
--NAME lere isim atadık.  
INSERT INTO CUSTOMERS2  
(NAME1,NAME2,NAME3,NAME4,NAME5,NAME6,NAME7,NAME8)
```

```
VALUES  
('ÖMER','ÖMER','ÖMER','ÖMER','ÖMER','ÖMER','ÖMER','ÖMER')
```

--Uygulama 2

--char(50) kopyalandığımızda 50 karakteri doldurmak için boşluk atıyor ama varchar'da boşlukları doldurmuyor!

```
SELECT NAME1,NAME2,NAME3,NAME4,NAME5,NAME6,NAME7,NAME8 FROM CUSTOMERS2
```

```
ÖMER  
ÖMER  
ÖMER  
ÖMER  
ÖMER  
ÖMER  
ÖMER
```

--char da kendi boşluk doldurduğu için uzunluğunu sabit tutuyor. Bu da arama da çok büyük kolaylık sağlar !

--Char'ı genelde sabit uzunlukta ise TC Kimlik,Telefon Numarası gibi bu alanlarda kullanabiliriz.

--6.4_MetinVeritipleri2_NVarcharNCharVarcharMaxVeriTipleri

Japonca da kitap olan 本 kelimeyi customers2 tablosuna ekledik ancak bazı stünlarda soru işaretini olara gözükmüyor. Bunun sebebi 255 karakter arasında olmayan bir karakter olduğu için.

The screenshot shows the Microsoft SQL Server Management Studio (SSMS) interface. In the Object Explorer, the database structure is visible, including the ETRADE database and its tables. A query is being run against the CUSTOMERS2 table, specifically selecting columns NAME1 through NAME8. The results show three rows of data, where all columns except NAME1 contain the character 'ÖMER' or a question mark. The status bar at the bottom indicates the query was executed successfully.

	NAME1	NAME2	NAME3	NAME4	NAME5	NAME6	NAME7	NAME8
1	ÖMER	ÖMER	ÖMER	ÖMER	ÖMER	ÖMER	ÖMER	ÖMER
2	ÖMER	ÖMER	ÖMER	ÖMER	ÖMER	ÖMER	ÖMER	ÖMER
3	?	本	本	本	本	?	?	?

Query executed successfully. | (local) (14.0 RTM) | SA (59) | ETRADE | 00:00:00 | 3 rows

Char, text, varchar da bu harfler tutulmaz ancak nchar,ntext ve nvarchar da bu karakter tutulabilir ! Bu sebeple eğer uluslararası bir yazılım yapıyororsak nchar,ntext ve nvarchar kullanmalıyız !

(Aşağıdaki resim değişkenlerin veri tiplerini görmek ve kıyaslamak için eklendi.)

The screenshot shows the Microsoft SQL Server Management Studio (SSMS) interface. The title bar reads "SQLEGITIM.ETRADE - dbo.CUSTOMERS2 - Microsoft SQL Server Management Studio (Administrator)". The menu bar includes File, Edit, View, Project, Table Designer, Tools, Window, and Help. The toolbar has various icons for database management tasks.

The Object Explorer on the left shows the database structure for "dbo.CUSTOMERS2", including columns, keys, constraints, triggers, indexes, and statistics.

The main pane displays the "CUSTOMERS2" table definition. The columns are:

Column Name	Data Type	Allow Nulls
ID	int	<input type="checkbox"/>
NAME1	char(50)	<input checked="" type="checkbox"/>
NAME2	nchar(50)	<input checked="" type="checkbox"/>
NAME3	ntext	<input checked="" type="checkbox"/>
NAME4	nvarchar(50)	<input checked="" type="checkbox"/>
NAME5	nvarchar(MAX)	<input checked="" type="checkbox"/>
NAME6	text	<input checked="" type="checkbox"/>
NAME7	varchar(50)	<input checked="" type="checkbox"/>
NAME8	varchar(MAX)	<input checked="" type="checkbox"/>
COUNTRYID	tinyint	<input checked="" type="checkbox"/>
CITYID	smallint	<input checked="" type="checkbox"/>

A tooltip for "NAME7" indicates it is currently selected. The "Column Properties" pane at the bottom shows "(General)" properties for the selected column.

Text ve ntext'i eski sürümlerle uyumluluk olması açısından böyle bir veri tipi var. Normalde daha çok daha yeni olan nvarchar ve varchar kullanılır !

6.5_TarihSaatVeriTipleri

TARİH-SAAT VERİ TİPLERİ

date	Minimum: 0001-01-01	4 Byte
	Maksimum: 9999-12-31	
smalldate	Minimum: 1900-01-01	3 Byte
	Maksimum: 2079-06-06	
datetime	Minimum: 1753-01-01 00:00:00.000	8 Byte
	Maksimum: 9999-12-31 23:59:59.997	
datetime2	Minimum: 0001-01-01 00:00:00.000000	1-2 Hassasiyet İçin = 6 Byte
	Maksimum: 9999-12-31 23:59:59.999999	3-4 Hassasiyet İçin = 7 Byte
		5-7 Hassasiyet İçin = 8 Byte
datetimeoffset	Minimum: 0001-01-01 00:00:00.000000	1-2 Hassasiyet İçin = 8 Byte
	Maksimum: 9999-12-31 23:59:59.999999	3-4 Hassasiyet İçin = 9 Byte
	Time zone offset Aralığı: -14:00 / +14:00	5-7 Hassasiyet İçin = 10 Byte
time	Minimum: 00:00:00.000000	5 Byte(Default olarak kullanılırsa)
	Maksimum: 23:59:59.999999	

Datetimeoffset'da time zone aralığını da tutabiliyoruz yani İstanbul da saat bu iken Londrada bu gibi tutabiliyoruz.

Datetime2(7) veri tipi ile aşağıdaki gibi saatte saniyeyi(30) ve 7 haneli(7777777) milisaniyeyi bile tutabiliyoruz.

1995-05-14 14:20:30.7777777

Ya da tarihler ve saatleri ayrı şekilde tutabiliriz. (Date ve time türünde iki tane değişken oluşturarak)

6.5_TarihSaatVeriTipleri

DİĞER VERİ TİPLERİ

image		Maksimum değeri: 2^31-1 (2,147,483,647) Byte
binary	0 ile 8000 arasında	Tanımladığı değer kadar Byte. Binary(10) -> 10 Byte
varbinary	0 ile 8000 arasında	Tanımladığı değer + 2 Byte
varbinary(MAX)	0 ile 2 147 483 647 arasında	Tanımladığı değer + 2 Byte Maksimum değeri: 2^31-1 (2,147,483,647) Byte
sql_variant		Bazı veri tiplerinin değerlerini saklamak için kullanılır. Aşağıdakiler hariç: varchar(max),varbinary(max),nvarchar(max),xml,text, ntext,image, rowversion(timestamp),sql_variant, geography,hierarchyid,geometry,User-defined types, datetimeoffset
Xml		Xml veriler için kullanılır.
Table		Sonradan kullanım amacıyla bir sonuç kümesini saklamak için kullanılır.

DİĞER VERİ TİPLERİ

uniqueidentifier	GUID(global olarak tekilliği garanti eder) veriyi tutar.
	select NEWID() script'ini çalıştırıldığınızda aşağıdaki gibi bir GUID veri oluşturur.
	A4C5DB26-7F18-4B4F-A898-E7DE26A8446A
	Bazen veritabanlarında tekilliği sağlamak için kullanılır. Arma bu amaçla kullanıldığından genelde performansı düşürür.
	Hiyerarşik yapılarda, hiyerarşideki pozisyonları temsil etmek için kullanılır.
geography	Dünyadaki koordinat sistemini tutar.
	Dünyanın eğimlerini de hesaba katarak.
geometry	Euclidean (flat) sistemi ile koordinat sistemini tutar.
	Sadece 2 düzlem üzerinden hesaplanır.
	Dünyanın eğimlerini hesaba katmaz.

Uniqueidentifier: Global olarak tekilliği garanti eder yani dünyada örneği olmasın dediğimiz verileri tutmak için

Bölüm Sonu Soruları--Kendini Test Et !

- **SQL Server'da Veri Tipi Kavramı ne demektir?**
- **Sayısal veri tipleri nelerdir?**
- **Integer, SmallInt,BigInt,TinyInt arasındaki farklar nelerdir?**
- **Float, Decimal, Money veri tipleri arasındaki farklar nelerdir?**
- **Varchar,Char, Nchar, Nvarchar arasındaki farklar nelerdir?**
- **Hangi veri tipi hafızada ne kadar yer tutar?**
- **Hangi veritipini seçeceğimize neye göre karar veririz?**
- **Date, Time ve Datetime arasında ne fark bulunmaktadır?**

7.1_IlliskiselVeritabaniKavramı(RDMS- Relational Database Management System)

İlliskisel veritabanında amaç tekrar eden verileri azaltmak veya minimuma indirmektir.

Şu ana kadar hep tek tabloda çalıştık ve bu tabloda hep tekrar eden veriler vardır. Örneğin mağazanın adı, mağazanın şehri birden fazla şekilde tekrar ediyor ya da müşteri adı birden fazla şekilde tekrar ediyor. Bu raporlamak için uygun ancak bu şekilde bir veri tabanı oluşturmak mantıklı değil sebebi ise örneğin müşteri adını değiştirdiğimizde diğer bilgileri de değiştirmek gereklidir.

ÖRNEK E TİCARET SİSTEMİ

1. Kullanıcı sisteme login olur.

Adres Telefon Email gibi bilgileri sistemde kayıtlıdır.

2. Seçtiği ürün ya da ürünler sepete ekler

Adres listesinden adres seçer

3. Sepete eklediği ürünler ödeme ekranına gider.

Bu sırada siparişi oluşturulur.

Kredi kartı ödemesi gerçekleşir.

5. Ürün sevk edilir.

Faturası kesilir ve ürün sevk edilir.

Tam anlamıyla **İlişkisel Veritabanı** kavramını daha iyi algılamak adına bir örnek üzerinden devam etmek istiyorum. Arkadaşlar Örneğin bir e-ticaret sistemini düşünün hepimiz internetten alışveriş yapıyoruz ve burada yaparken üç aşağı beş yukarı yapılan adımlar belli arkadaşlar ne yapıyoruz sisteme Kullanıcı olarak login oluyoruz. Eğer sistemde tanımlı bir kullanıcı değilsek ilk önce kaydımızı gerçekleştiriyoruz ve sonrasında login oluyoruz. Tabii bu sırada adres telefon email gibi bilgilerimiz sistemde zaten kayıtlı. Sonrasında ürünleri seçiyoruz ürünlerle Bakiyoruz ya baktığımız ürünleri etiklerimizi almak istediklerini de sepete ekliyoruz sonra sepete eklediğiniz ürünlerin ödeme ekranına gidiyoruz ve ödeme ekranında adres hissesinden bana ait olan adres sitesinden adreslerden birini seçiyoruz sonra kredi kartı ödemesini gerçekleştiriyoruz ve bu sırada bizim adımıza otomatik olarak bir sipariş oluşturuluyor oluşturulan Bu sipariş hazırlandıktan sonra bize yukarıda seçmiş olduğunuz adrese sevk ediliyor ve sevk esnasında da faturası kesiliyor ve ürün bize gönderiyor temel anlamda en basit anlamda bir e-ticaret sistemi bu şekilde çalışıyor Pakistan bu şekilde çalışırken biz bunun veritabanı altyapısı nasıl oluşturabiliriz çok basit anlamda veritabanı altyapısı nasıl oluşturulabiliriz ve o alt yapıyı oluştur kendi ilişkisel veritabanı sistemini kullanarak bu işi nasıl gerçekleştirebiliriz şu an bütün model veritabanı sistemleri ilişkisel olarak çalışmayı destekliyor ve sektörde bilinen uygulamaların neredeyse tamamı ilişkisel üzerinden çalışmasına devam ediyor.

7.2_IlliskiselVeritabaniKavrami1

7.1 den 7.6 ye kadar EXCELL de “ticaret sitesine” ait veri tabanı tasarladık. Tasarladığımız veritabanı ve arasındaki ilişkiler ise aşağıda verilmiştir.(Excell doyasının ismi: VeritabaniOlusturma.xlsx)

Tablo:7.2.1

Database Schema Diagram														
Users		ADDRESS			COUNTRIES			CITIES			TOWNS		DISTRICTS	
1	ID	ID	INT IDENTITY(1,1)	ID	ID	TINYINT IDENTITY(1,1)	ID	ID	TINYINT IDENTITY(1,1)	COUNTRY	ÜLKE ADI	VARCHAR(50)		
2	USERNAME_	KULLANICI ADI	VARCHAR(50)	USERID	KULLANICI ID	INT	COUNTRYID	ÜLKE ID	TINYINT					
3	PASSWORD_	ŞİFRE	VARCHAR(50)	CITYID	ŞEHİR ID	SMALLINT								
4	NAMESURNAME	AD SOYAD	VARCHAR(50)	TOWNID	İLÇE ID	INT	DISTRICTID	SEMT ID	INT					
5	EMAIL	EMAIL	VARCHAR(50)	POSTALCODE	POSTAKODU	VARCHAR(10)								
6	GENDER	CİNSİYET	VARCHAR(1)	ADDRESSTEXT	AÇIK ADRES	VARCHAR(500)								
7	BIRTHDATE	DOĞUM TARİHİ	DATE											
8	CRATEDATE	OLUŞTURMA TARİHİ	DATETIME											
9	TELNR1	TEL1	VARCHAR(15)											
10	TELNR2	TEL2	VarCHAR(15)											
11	ADRESS													
12														
13														
14														
15														
16														
17														
18														
19														
20														
21														

Özetle Bu her bir tablo arasındaki ilişki primary key ve foreign key olarak açıklayacak olursak örneğin COUNTRIES ve CITIES tablosu arasındaki ilişkiyi sağlayan değişkenler aşağıda verilmiştir.

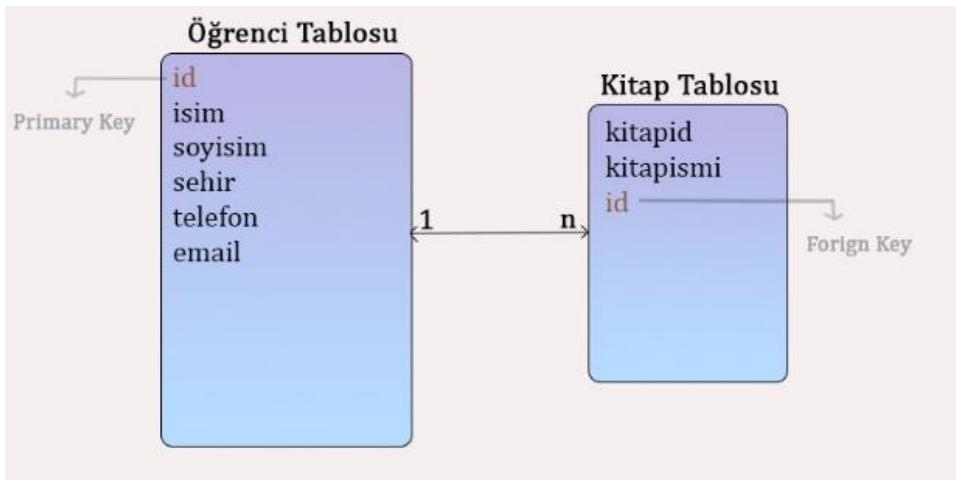
Tablo:7.2.2

Primary key

I	J	K	L
COUNTRIES			
ID	ID	TINYINT IDENTITY(1,1)	
COUNTRY	ÜLKE ADI	VARCHAR(50)	
CITIES			
ID	ID	SMALLINT IDENTITY(1,1)	
COUNTRYID	ÜLKE ID	TINYINT	
CITY	ŞEHİR	VARCHAR(50)	

Foreign key

Başka bir örnek olarak kendi yapmış olduğum projenin veri tabanına ait tasarımını:



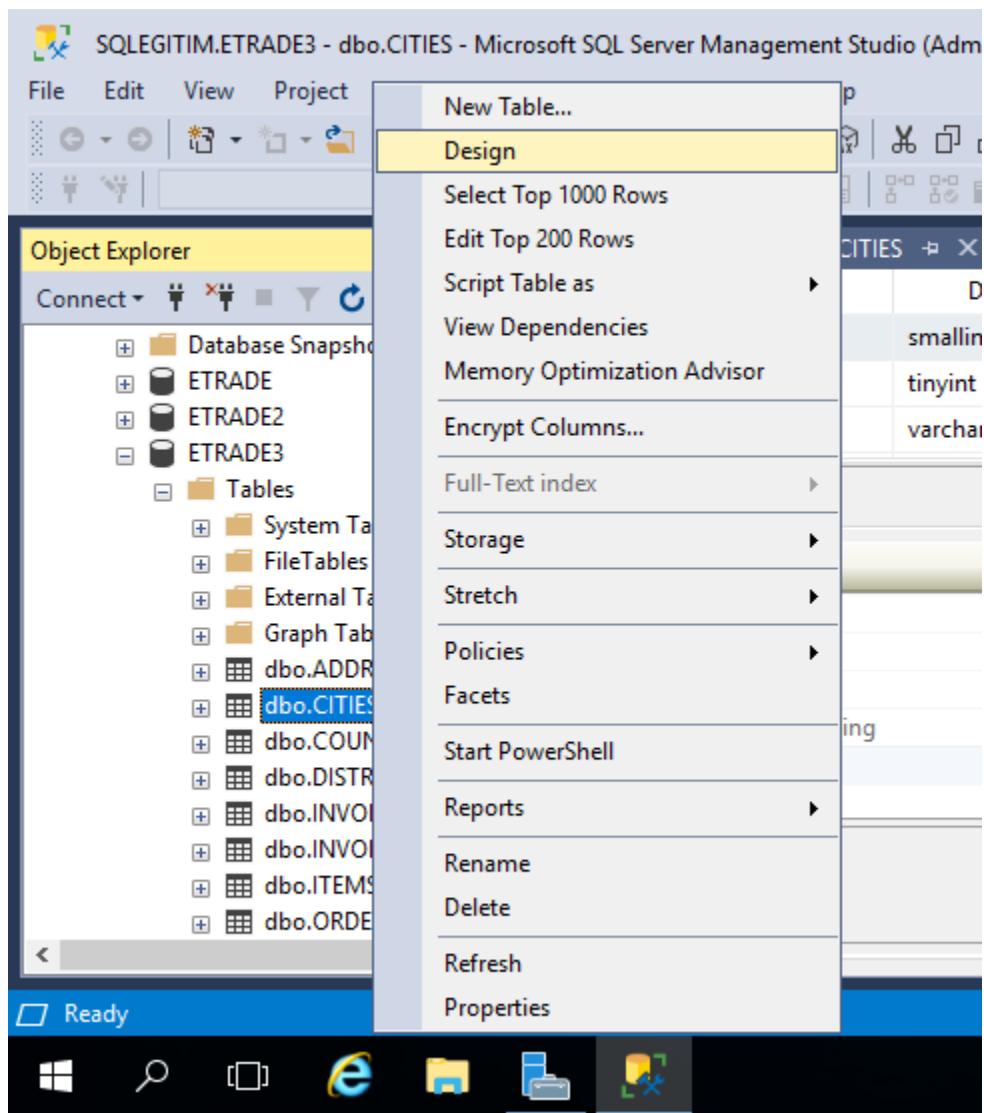
Tabi burada Kitap Tablosundaki id ye Öğrenci Tablosunun id'i si ile ilişkili olduğu için `ogrenciid` dersek daha tatlı olurdu çünkü hangi tabloyla ilişkili olduğunu da değişken isminden bile anlardık. Tabi ki burdaki tablolar sadece 2 tane olduğu için bu pekte önemli değil. Aslında buradan da anlaşıldığı gibi iki tablo arasındaki ilişkiyi kodlarla ilişkili hale getireceğimiz için aynı isim veya farklı isim almasının hiçbir manası yok.

7.7_IlliskiselVeritabaniTasarimi_ManagementStudio

Burda ise Tablo 7.2.1 deki tasarımını SQL de yaptık

7.8_RelationKavramı

Tablo 7.2.2 de bahsettiğim ilişkiyi biz SQL de Relation kavramı ile yapıyoruz. Yine COUNTRIES ve CITIES tablosunu ele alacak olursak CITIES tablosunda sağa tıklayarak Design edelim.



Daha sonra ise Açılan tablo üzerinde sağa tıklayıp Relationships ayarlamak için gerekli adımlara geçelim(hangi değişken üzerinde olduğu farketmez!)

SQLEGITIM.ETRADE3 - dbo.CITIES - Microsoft SQL Server Management Studio (Admin...)

File Edit View Project Table Designer Tools Window Help

New Query MDX DMX XMLA DAX Execute

Object Explorer

Connect Database Snapshots ETRADE ETRADE2 ETRADE3 Tables System Tables FileTables External Tables Graph Tables dbo.ADDRESS dbo.CITIES dbo.COUNTRIES dbo.DISTRICTS dbo.INVOICEDETAILS dbo.INVOICES dbo.ITEMS dbo.ORDERDETAILS

SQLGITIM.ETRADE3 - dbo.CITIES

Column Name	Data Type	Allow Nulls
ID	smallint	<input type="checkbox"/>
COUNTRYID	tinyint	<input checked="" type="checkbox"/>
CITY		

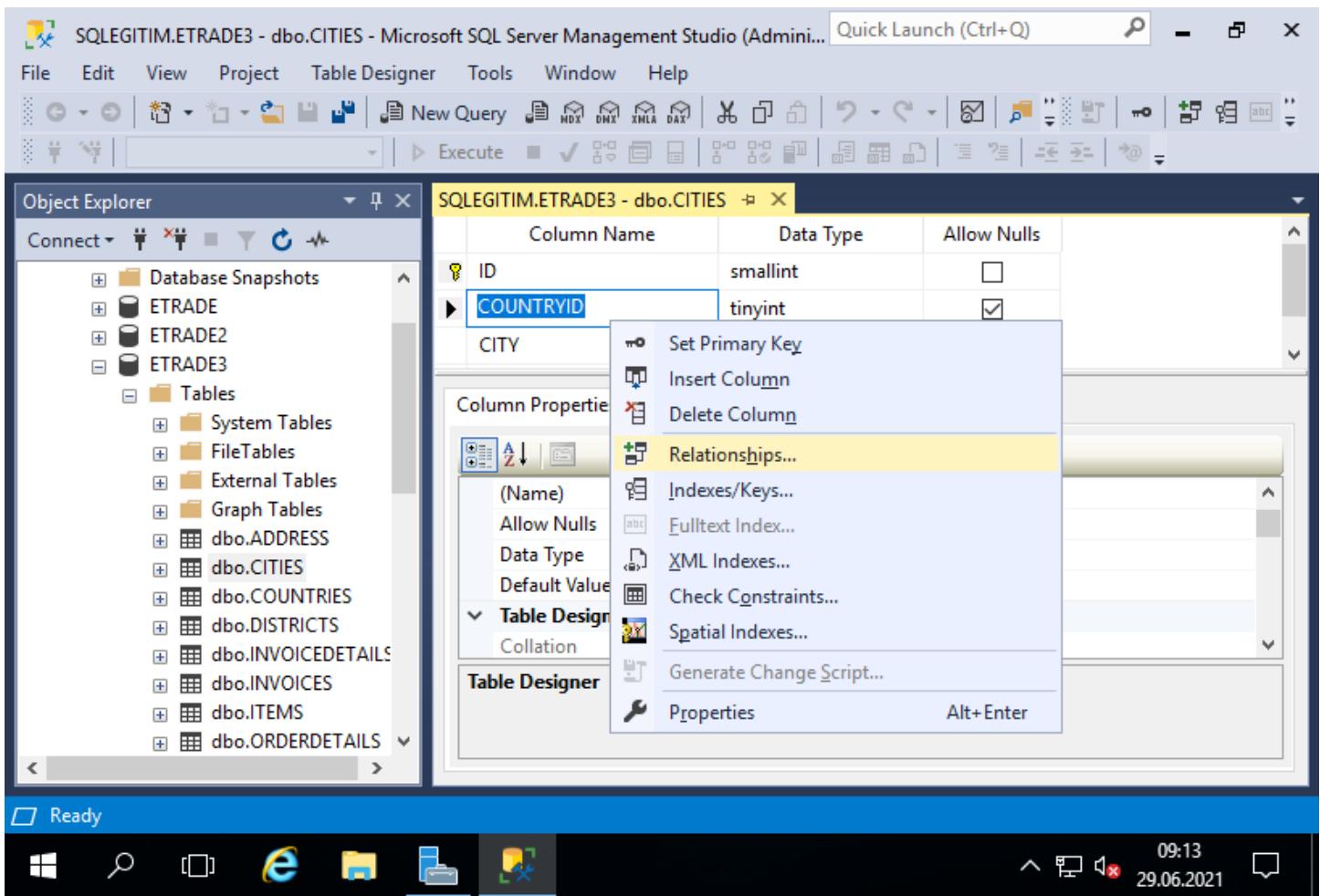
Column Properties

- (Name)
- Allow Nulls
- Data Type
- Default Value
- Table Design
- Collation

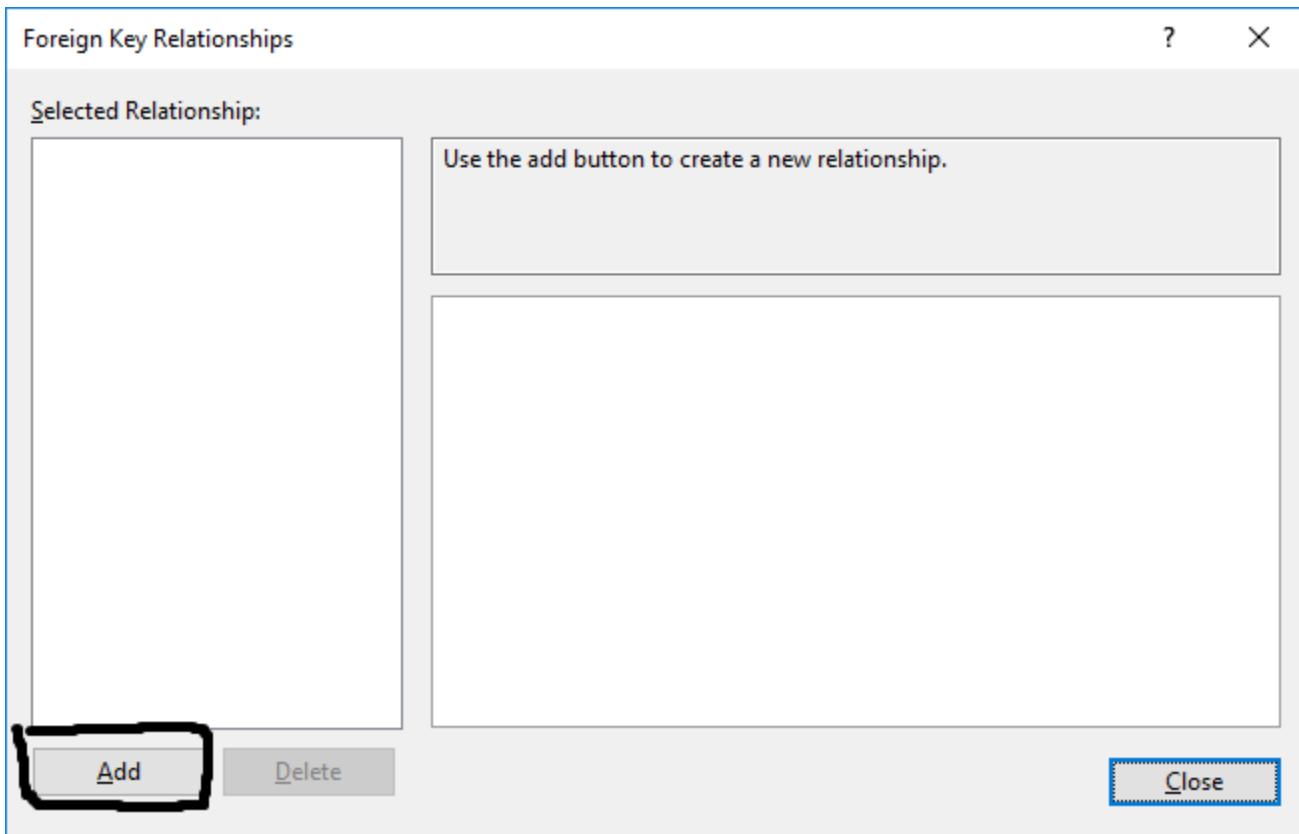
Relationships...

- Indexes/Keys...
- Fulltext Index...
- XML Indexes...
- Check Constraints...
- Spatial Indexes...
- Generate Change Script...
- Properties

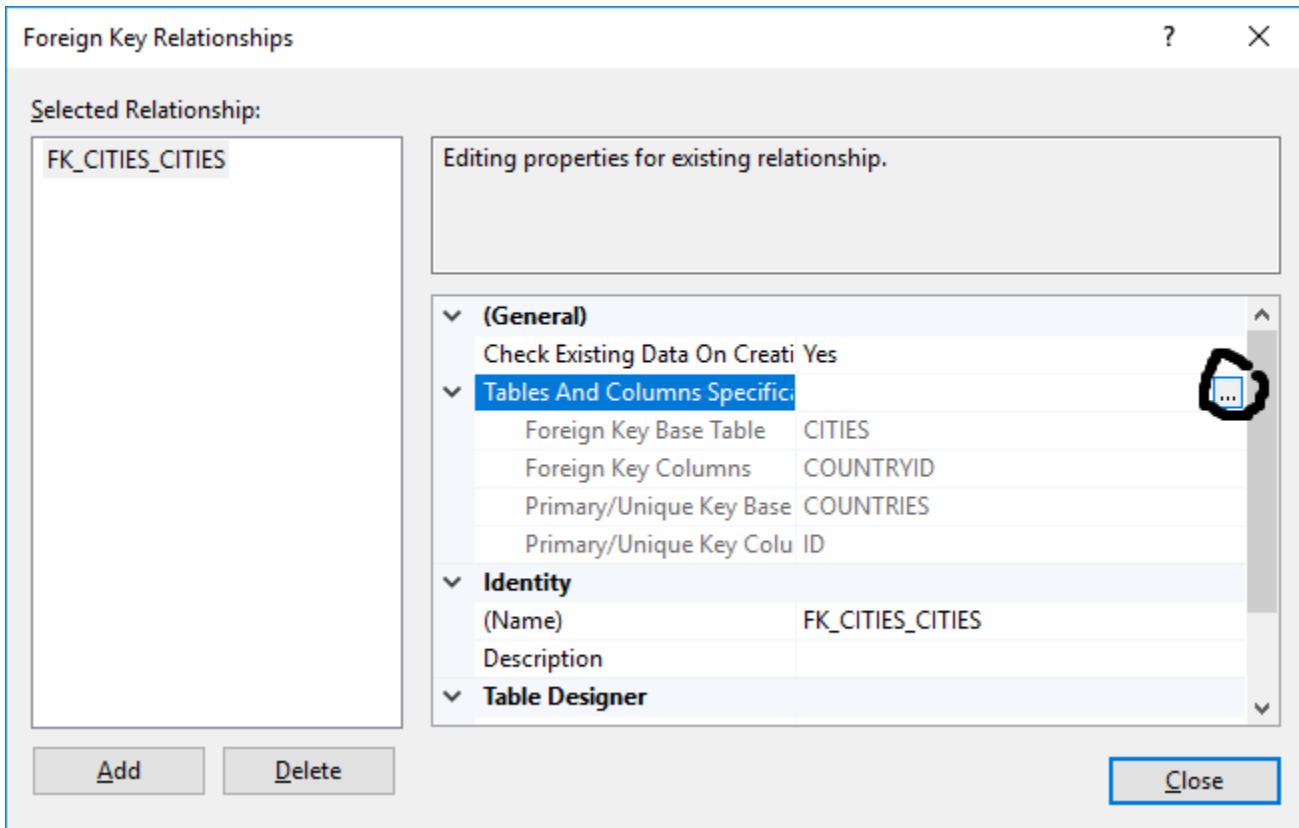
Ready 09:13 29.06.2021



Relationships e tıkladıktan sonra böyle bir pencere ile karşılaşacağız. İlişki kurmak için Add butonuna tıklayalım.



Hemen ardından sağ bölümde Tables And Columns Spec... sağında bulunan ‘...’ kısma tıklayarak COUNTRIES ve CITIES Tablosu arasındaki ilişkiyi kurmaya geçelim.



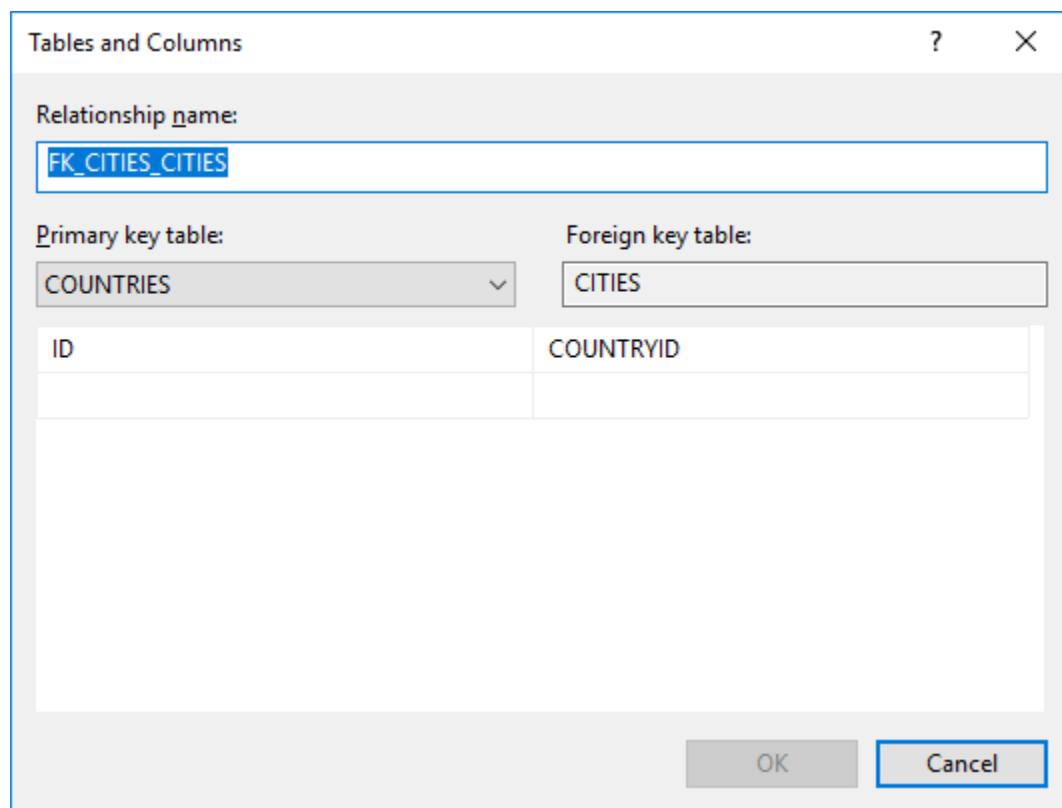
Tablo:7.2.2 dende hatırladığımız gibi COUNTRIES için id miz COUNTRIES tablosu için kendi id'si CITIES tablosuyla ilişkilendirmek için kullandığımız değişkenimiz ise COUNTRYID idi (yani Foreign keyimiz).

Primary key

I	J	K	L
COUNTRIES			
ID	ID	TINYINT IDENTITY(1,1)	
COUNTRY	ÜLKE ADI	VARCHAR(50)	
CITIES			
ID	ID	SMALLINT IDENTITY(1,1)	
COUNTRYID	ÜLKE ID	TINYINT	
CITY	ŞEHİR	VARCHAR(50)	

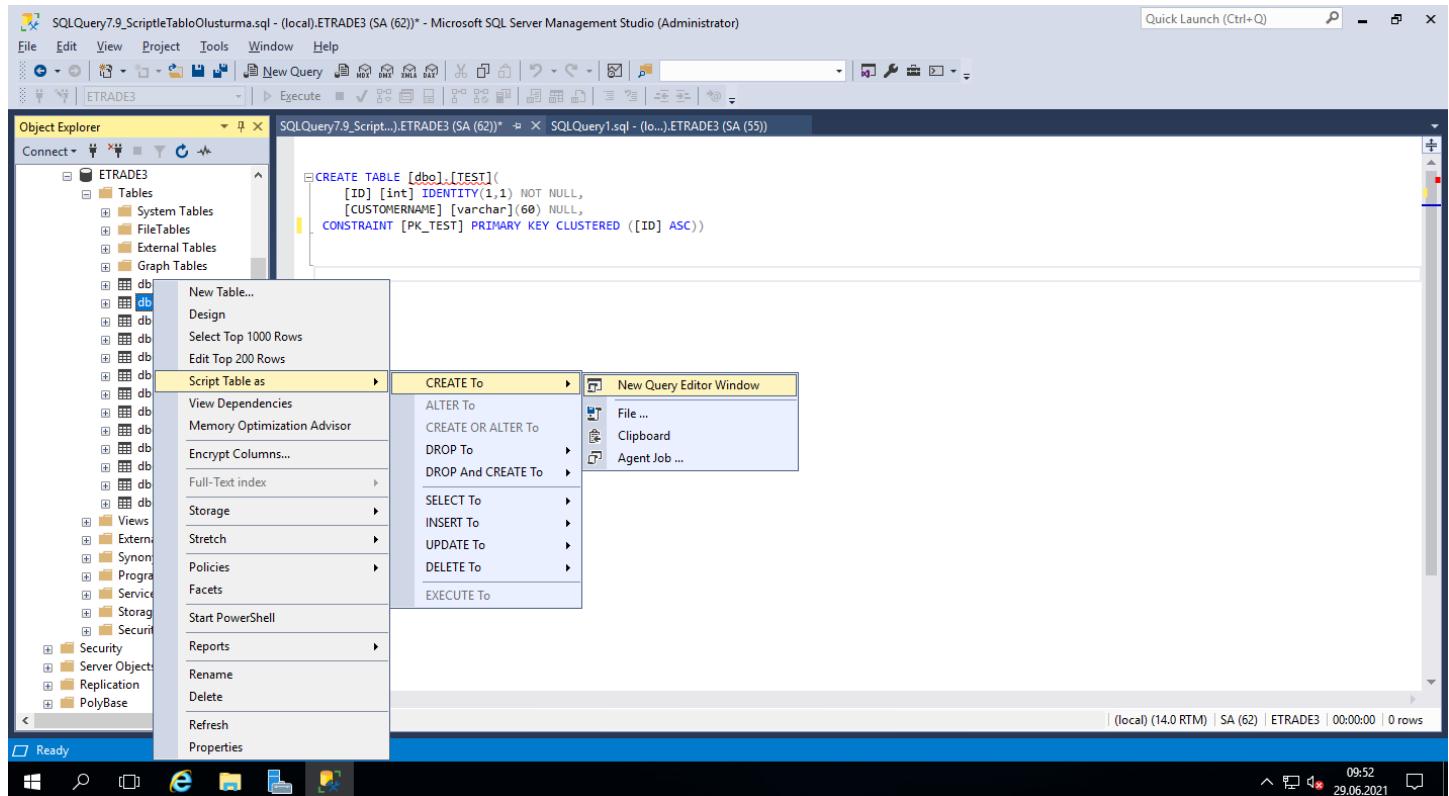
Foreign key

Bu sebeple COUNTRIES için ID CITIES için COUNTRYID deki bu ilişkiyi SQL e bu şekilde tanıtmış olduk!



7.9_ScriptİleTabloOlusturma

Örnek olarak biz burada CITIES tablosunu ele alacağız. CITIES tablosunda sağa tıklayarak **Script Table as > CREATE To > New Query Editor Window** ile tablomuzun scriptini yeni ekranında getirilebiliriz.



Aşağıdaki gibi bir Scriptle karşılaşacağız. İlk başta karmaşık gelebilir ancak biz bunu sadeleştireceğiz

```
USE [ETRADE3]
GO

/******** Object: Table [dbo].[CITIES] Script Date: 29.06.2021 09:55:01 *****/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [dbo].[CITIES](
    [ID] [smallint] IDENTITY(1,1) NOT NULL,
    [COUNTRYID] [tinyint] NULL,
    [CITY] [varchar](50) NULL,
    CONSTRAINT [PK_CITIES] PRIMARY KEY CLUSTERED
(
    [ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON)
) ON [PRIMARY]
GO
```

Excelde kullanacak şekilde bu şekilde sadeleştirdik

```
CREATE TABLE [dbo].[CITIES](
    [ID] [smallint] IDENTITY(1,1) NOT NULL,
    [COUNTRYID] [tinyint] NULL,
    [CITY] [varchar](50) NULL,
    CONSTRAINT [PK_CITIES] PRIMARY KEY CLUSTERED
(
    [ID] ASC
)
```

Ve Excel de iş yapmak için bunu aşağıdaki şekilde düzenleyebiliriz.

CREATE TABLE [dbo].[CITIES](

```
[ID] [smallint] IDENTITY(1,1) NOT NULL,  
[COUNTRYID] [tinyint] NULL,  
[CITY] [varchar](50) NULL,  
CONSTRAINT [PK_CITIES] PRIMARY KEY CLUSTERED ([ID] ASC))
```

Başka Tablo için bu komutu kullanacak olursak :

CITIES ---> Tablo ismi

ID ---> Tablonun Primary Keyi

, ---> ve ilgili tablonun son değişkenindeki virgülü kaldırarak rahatlıkla kullanabiliriz. Son hali:

```
CREATE TABLE [dbo].[CITIES](  
[ID] [smallint] IDENTITY(1,1) NOT NULL,  
[COUNTRYID] [tinyint] NULL,  
[CITY] [varchar](50) NULL  
CONSTRAINT [PK_CITIES] PRIMARY KEY CLUSTERED ([ID] ASC))
```

Aşağıdaki gibi son değişkendeki virgülü(Tüm tablolardan) kaldırımıda unutmayalım !

7.10_Bolumsonu_RDMS

- **İlişkisel veritabanı ne demektir?(RDMS)**
- **RDMS sistemine sahip bir yapı temel anlamda nasıl çalışır? (Fatura örneği)**
- **Bir E-Ticaret sistemi temel olarak nasıl çalışır?**
- **RDMS bir yapıda veritabanı mimarisi nasıl oluşturulur?**

8.1_OrnekDataAtma

Excell den hızlıca özellikle az sayıda veri atmak için bir yönteme değinelim:

ETRADE3 database mizdeki CITIES tablosuna örnek veri ekleyelim. Bunun için ilk verimizi TRUNCATE TABLE komutu verilerin tamamını (otomatik artan id ler dahil) siliyoruz.CITIES tablomuz tamamen temizlendikten sonra

```
Truncate table CITIES
```

ID	COUNTRYID	CITY
----	-----------	------

Query executed successfully.

CITIES tablomuza sağ tıklayarak Edit Top 200 Rows ile tablomuza ulaşalım

- New Table...
- Design
- Select Top 1000 Rows
- Edit Top 200 Rows**
- Script Table as
- View Dependencies
- Memory Optimization Advisor
- Encrypt Columns...
- Full-Text index
- Storage
- Stretch
- Policies
- Facets
- Start PowerShell
- Reports
- Rename
- Delete
- Refresh
- Properties

COUNTRYID	CITY
NULL	NULL

Hemen arkasından aşağıdaki gibi sağ üstte bulunan SQL e tıklayalım

The screenshot shows the Microsoft SQL Server Management Studio interface. The title bar reads "SQLEGITIM.ETRADE3 - dbo.CITIES - Microsoft SQL Server Management Studio (Administrator)". The toolbar has items like File, Edit, View, Project, Query Designer, Tools, Window, Help, and a "SQL" button circled in yellow. The Object Explorer on the left shows the database structure under "ETRADE3". The main pane displays a results grid for the "CITIES" table:

	ID	COUNTRYID	CITY
*	NULL	NULL	NULL

Daha sonra karşımıza böyle bir komut satırı gelecek:

The screenshot shows the Microsoft SQL Server Management Studio interface. The title bar reads "SQLServer - VMware Workstation 15 Player (Non-commercial use only)" and "SQLEGITIM.ETRADE3 - dbo.CITIES - Microsoft SQL Server Management Studio (Administrator)". The toolbar has items like File, Edit, View, Project, Query Designer, Tools, Window, Help, and a "SQL" button. The Object Explorer on the left shows the database structure under "ETRADE3". The main pane displays a query window with the following SQL code and a results grid:

```
SELECT TOP (200) ID, COUNTRYID, CITY
FROM CITIES
```

	ID	COUNTRYID	CITY
*	NULL	NULL	NULL

Zaten ID miz otomatik artan olduğu için oraya herhangi bir veri girmemize gerek olmadığı için ID dahil olmak üzere "TOP (200) ID, " kısmını siliyoruz. Ve son hali aşağıda görüldüğü gibi olmalı.

SQLEGITIM.ETRADE3 - dbo.CITIES - Microsoft SQL Server Management Studio (Administrator)

File Edit View Project Query Designer Tools Window Help

New Query Execute

Object Explorer

Connect ETRADE3 Tables System Tables FileTables External Tables Graph Tables dbo.ADDRESS dbo.CITIES dbo.COUNTRIES dbo.DISTRICTS dbo.INVOICEDETAILS dbo.INVOICES dbo.ITEMS dbo.ORDERDETAILS dbo.ORDERS dbo.PAYMENTS dbo.TEST dbo.TOWNS dbo.USERS Views

SQLQuery3.sql - (lo...).ETRADE3 (SA (60))*

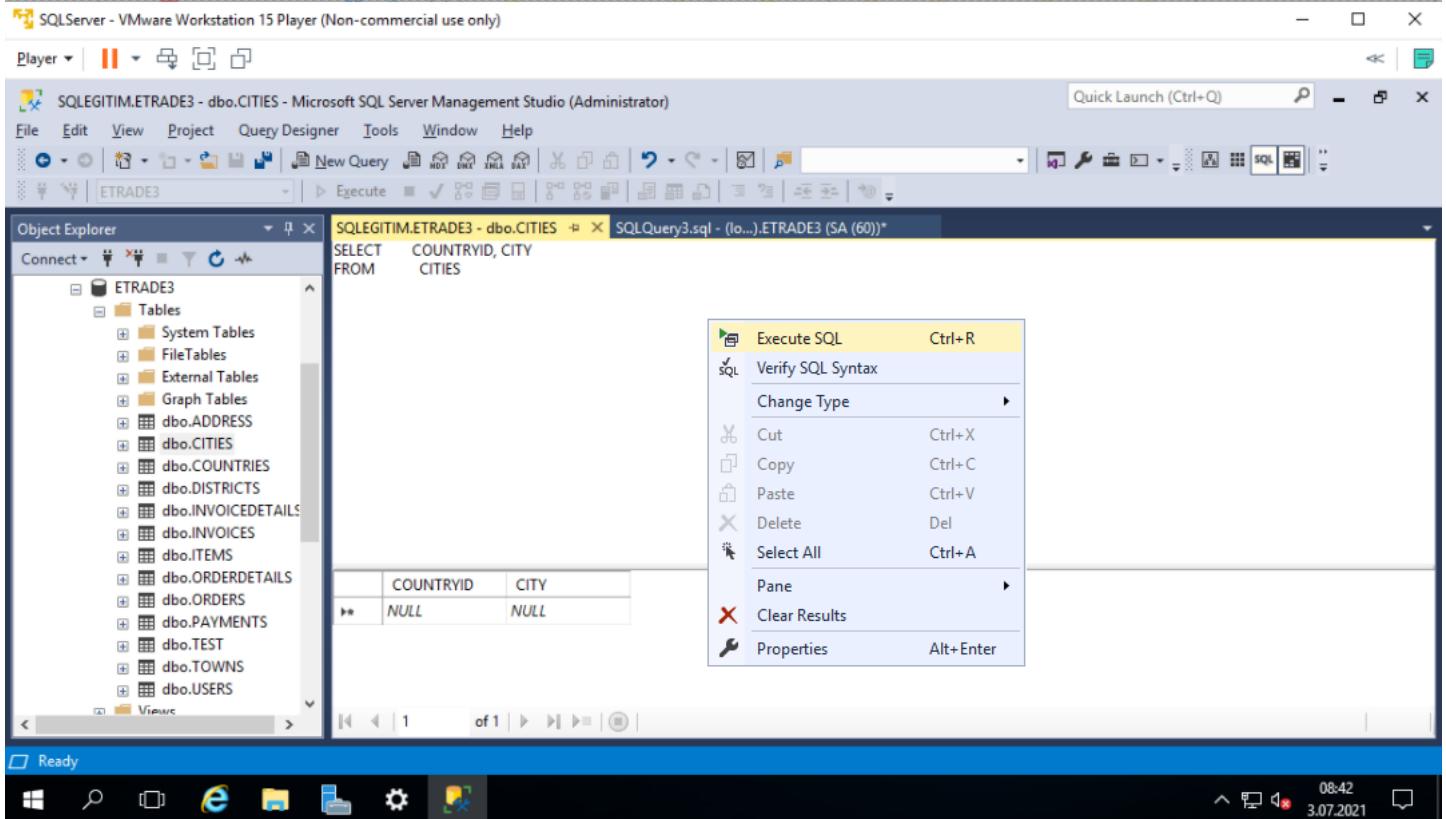
```
SELECT COUNTRYID, CITY
FROM CITIES
```

	ID	COUNTRYID	CITY
*	NULL	NULL	NULL

Ready

Query Changed

Ve sağa tıklayıp Execute yaparak komutu çalışıralım.



Silinecek:

SQLServer - VMware Workstation 15 Player (Non-commercial use only)

Player | New Query | XML DAX | Execute | SQL

Quick Launch (Ctrl+Q)

File Edit View Project Query Designer Tools Window Help

ETRADE3 | Object Explorer | Connect | Object Explorer | Tables | System Tables | FileTables | External Tables | Graph Tables | dbo.ADDRESS | dbo.CITIES | dbo.COUNTRIES | dbo.DISTRICTS | dbo.INVOICEDETAILS | dbo.INVOICES | dbo.ITEMS | dbo.ORDERDETAILS | dbo.ORDERS | dbo.PAYMENTS | dbo.TEST | dbo.TOWNS | dbo.USERS

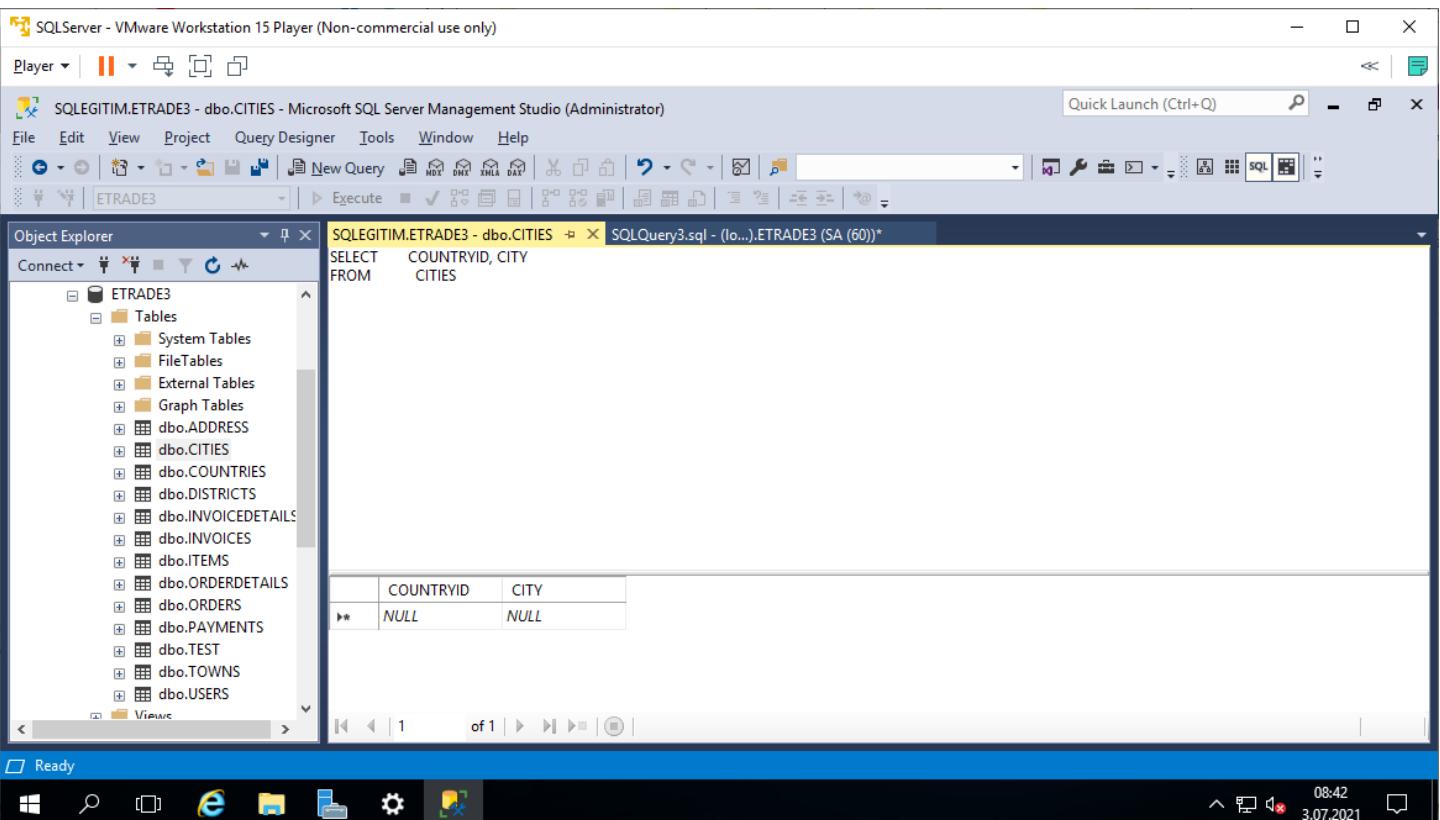
Object Explorer | SQLEGITIM.ETRADE3 - dbo.CITIES - Microsoft SQL Server Management Studio (Administrator) | SQLQuery3.sql - (Io...).ETRADE3 (SA (60))*

SELECT COUNTRYID, CITY
FROM CITIES

	COUNTRYID	CITY
*	NULL	NULL

Ready

08:42 3.07.2021



Excelden verilerimizi yapıştırmak üzere kopyalayalım(Tüm stünu CTRL+SHIFT+AşağıYön oku ile seçebiliriz.):

Uygulamalarla SQL Öğreniyorum | OrnekVeriler.xlsx - Microsoft Excel | UygulamalarlaSQLÖğreniyorumK...

onelive.live.com/edit.aspx?cid=a3472054ce1833e7&page=view&resid=A3472054CE1833E7!12929&parId=A3...

Uygulamalar RC Model Sıfır, İkin... Post Malone - Circl... Ekşi Şeyler GTAV İle Kodlama Eğitim Mobilya RC Okuma listesi

Excel OrnekVeriler - OneDrive üzerinde kaydedildi Ara (Alt + G) Premium'a geçin

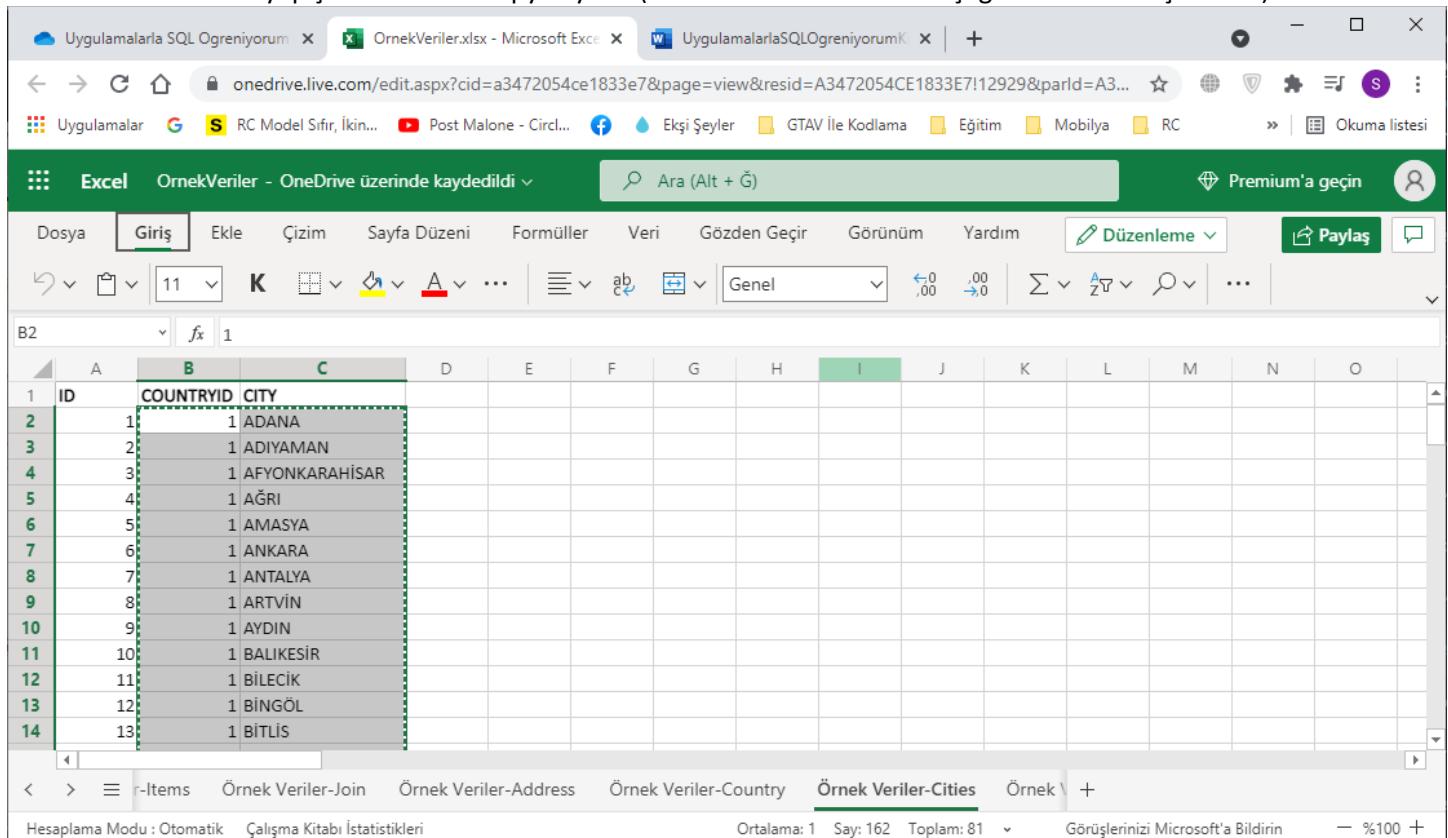
Dosya Giriş Ekle Çizim Sayfa Düzeni Formüller Veri Gözden Geçir Görünüm Yardım Düzenleme Paylaş

B2

ID	COUNTRYID	CITY
1	1	ADANA
2	1	ADIYAMAN
3	1	AFYONKARAHİSAR
4	1	AĞRI
5	1	AMASYA
6	1	ANKARA
7	1	ANTALYA
8	1	ARTVİN
9	1	AYDIN
10	1	BALIKESİR
11	1	BİLECİK
12	1	BİNGÖL
13	1	BITLİS

-Items Örnek Veriler-Join Örnek Veriler-Address Örnek Veriler-Country Örnek Veriler-Cities Örnek +

Hesaplama Modu : Otomatik Çalışma Kitabı İstatistikleri Ortalama: 1 Say: 162 Toplam: 81 Görüslərinizi Microsoft'a Bildirin - %100 +



Daha sonra ise aşağıdaki siyah ile yuvarlak içine aldığım kısma birkere tıklayalım.

SQLServer - VMware Workstation 15 Player (Non-commercial use only)

Player | New Query | XMLA DAX | Execute | Quick Launch (Ctrl+Q) |

SQLLEGITIM.ETRADE3 - dbo.CITIES - Microsoft SQL Server Management Studio (Administrator)

File Edit View Project Query Designer Tools Window Help

ETRADE3 | Execute |

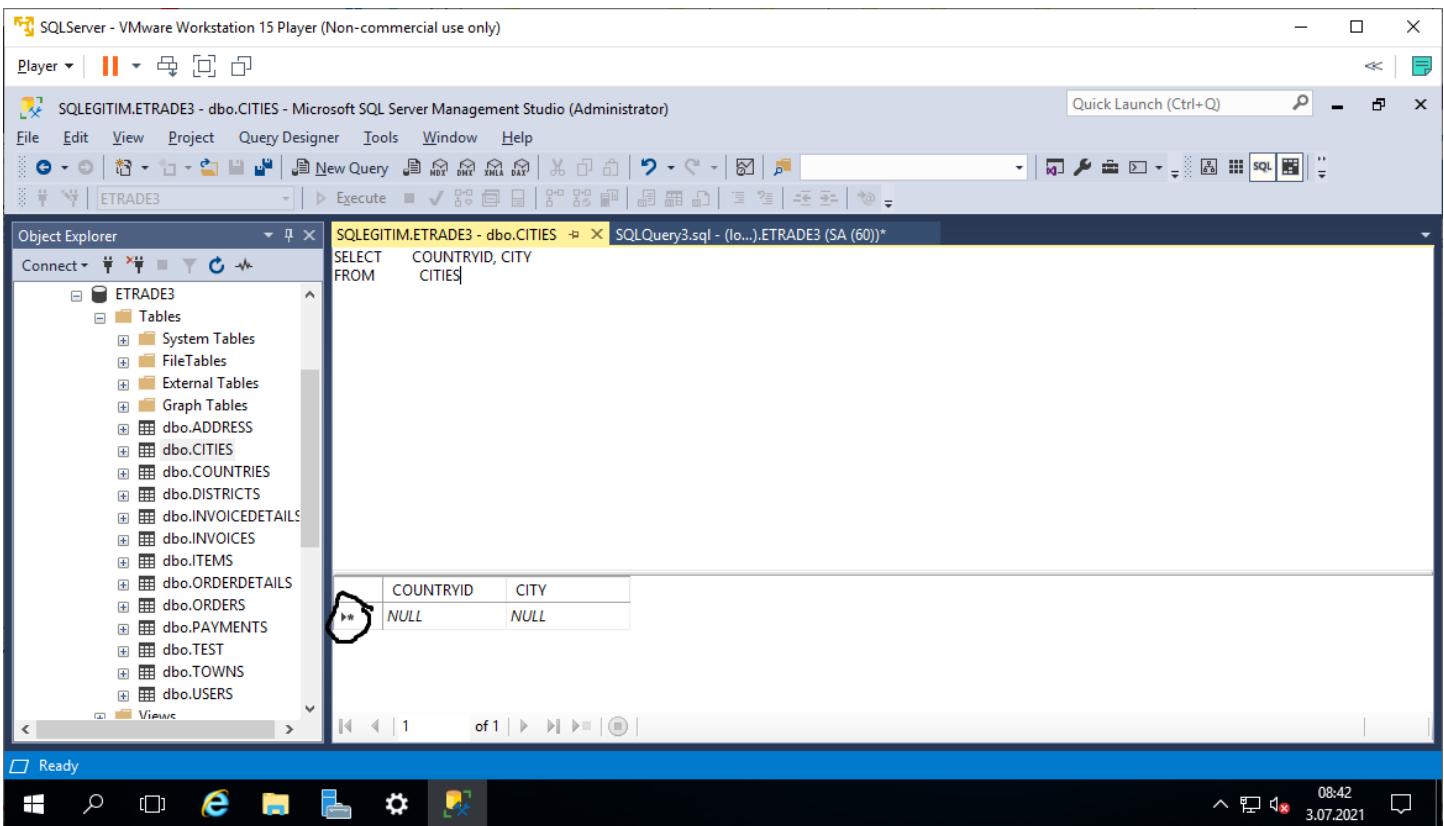
Object Explorer | Connect |

SELECT COUNTRYID, CITY
FROM CITIES

	COUNTRYID	CITY
**	NULL	NULL

Ready

08:42 3.07.2021



Tıkladıktan sonra aşağıdaki gibi her iki satırda mavi olarak seçilmiş olduğundan emin olalım

SQLServer - VMware Workstation 15 Player (Non-commercial use only)

Player | New Query | XMLA DAX | Execute | Quick Launch (Ctrl+Q) |

SQLLEGITIM.ETRADE3 - dbo.CITIES - Microsoft SQL Server Management Studio (Administrator)

File Edit View Project Query Designer Tools Window Help

ETRADE3 | Execute |

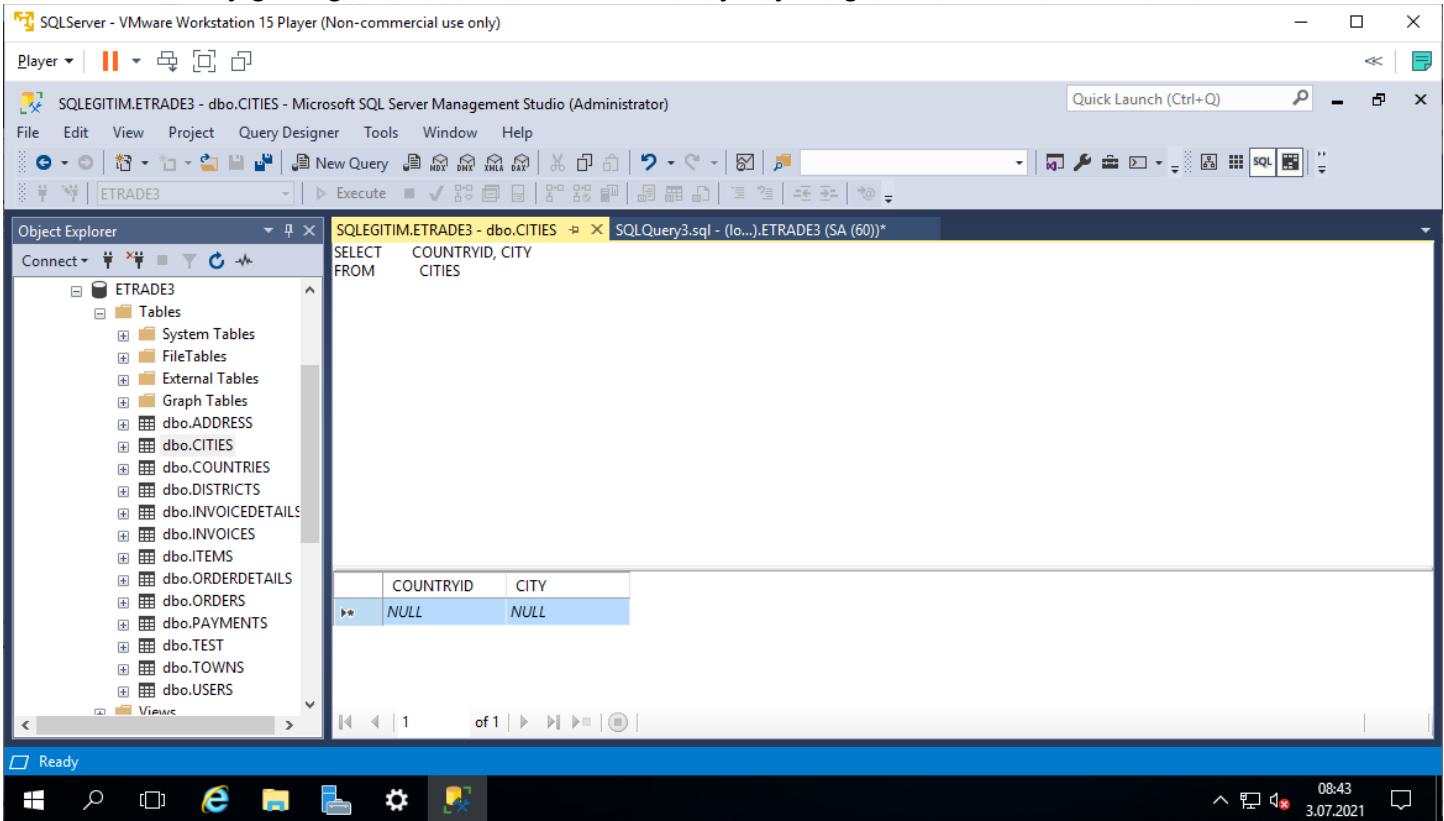
Object Explorer | Connect |

SELECT COUNTRYID, CITY
FROM CITIES

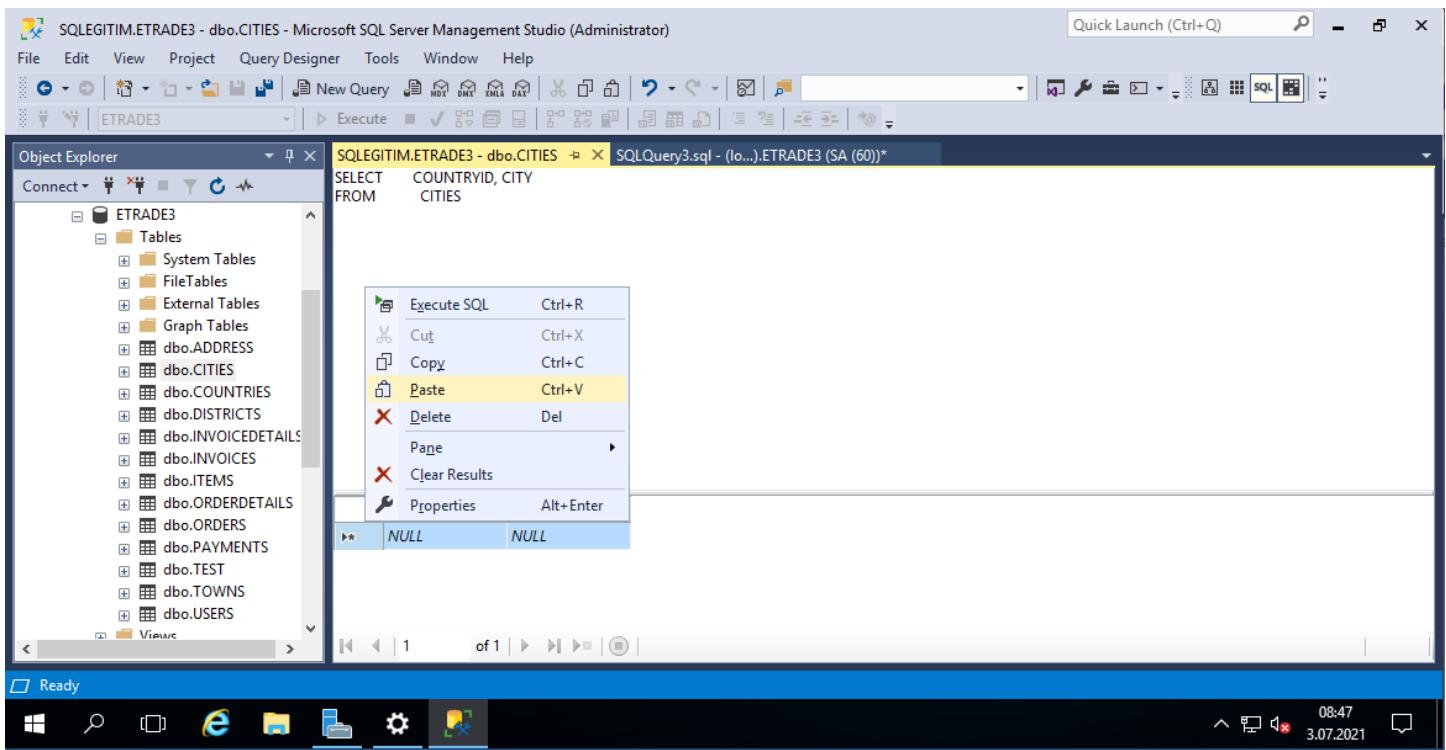
	COUNTRYID	CITY
**	NULL	NULL

Ready

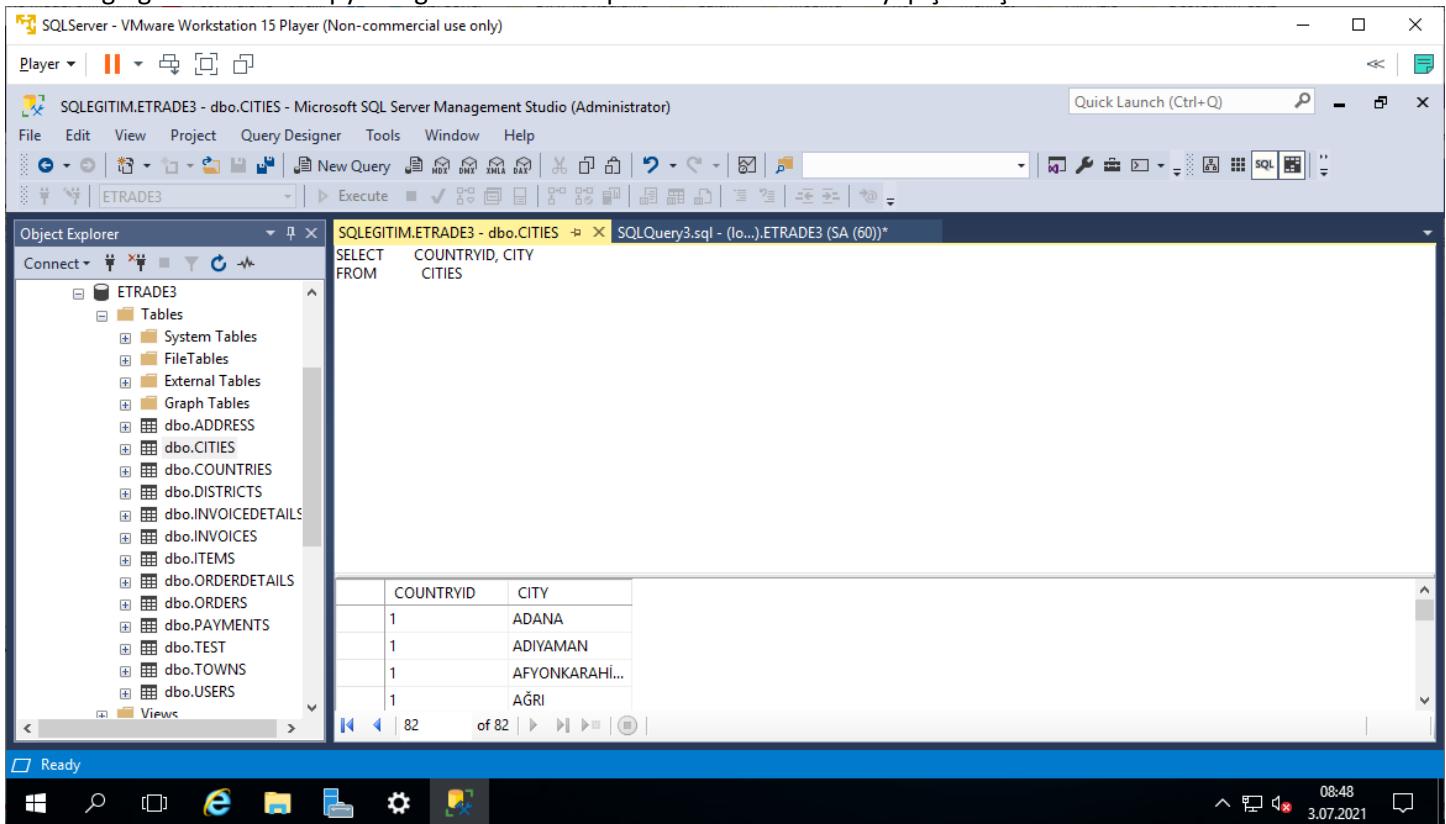
08:43 3.07.2021



Ve yine yukarıdaki siyah ile daire içine aldığım yerde sağa tıklayarak paste ile verilerimizi yapıştalım



Göründüğü gibi excelden kopyaladığım verilerin hepsini CITIES tablomuza yapıştırmış olduk:



ANCAK bu işlem pratik olsa da 50000 satırlık veriyi aktarmamız 15 Dk sürdü ! Bu yöntem pratik olduğu kadar en yavaş olanıdır. O sebeple az verileri bu şekilde yapıştırıldı ancak DISTRICTS için verilerimiz 50.000 den fazla olduğu için onu yeni bir yöntem ile SQL tablomuza ekleyelim.

8.2_IkiTabloyuBirlestirme

--Sorgular

SELECT * FROM USERS WHERE ID=1 --ID si 1 olanı listeledik

SELECT * FROM ADDRESS WHERE USERID=1 --USERID si 1 olanı listeledik.

--Uygulama 1

SELECT USERS.* , ADDRESS.ADDRESSTEXT --USERS tablosundan tüm stünlar ve ADDRESS tablosundaki ADDRESSTEXT stünler olacak şekilde

FROM

USERS, ADDRESS --USERS ve ADDRESS tablosundan

WHERE USERS.ID=ADDRESS.USERID --(ADDRESS tablosundaki USERSID---Foreign Key) Bu iki kavramla iki tabloyu USERS tablosundaki ID ile ADDRESS tablosundaki USERSID o kişiye ait address olduğundan WHERE USERS.ID=ADDRESS.USERID ile birleştirdik.

--Uygulama 2

SELECT USERS.* , ADDRESS.ADDRESSTEXT FROM USERS, ADDRESS WHERE USERS.ID=ADDRESS.USERID AND USERS.ID=3 --burda ise sadece ID si 3 olan kullanıcının adreslerini listeledik.

8.3_AliasKullanimi

--Sorgumuz

SELECT

USERS.NAMESURNAME, USERS.EMAIL, USERS.GENDER,

USERS.BIRTHDATE,

ADDRESS.ADDRESSTEXT

FROM

USERS, ADDRESS

WHERE USERS.ID = ADDRESS.USERID

AND USERS.ID=4

--Alias Kullanımı(Uygulama 1)

--USERS ve ADDRESS tablosu için sürekli bu tablo isimlerini yazmak yerine onlara kısa ve geçici isimler verebiliriz.

--Alias verdikten sonra tüm USERS ve ADDRESS isimlerini verdığımız alias ile değiştirelim

SELECT

U.NAMESURNAME, U.EMAIL, U.GENDER,

U.BIRTHDATE,

A.ADDRESSTEXT

FROM

USERS U, ADDRESS A --Burada Users için U ADDRESS için A alias veriyoruz(yani çok karmaşık sorgularda kolaylık için kısa isimler veriyoruz)

```
WHERE U.ID = A.USERID
```

```
AND U.ID=4
```

--Uygulama 2

--Burda ise stün isimlerine geçici isimler verelim:

```
SELECT
```

```
U.NAMESURNAME AS ADSOYAD, --Önemli not: AS kullanarakta yapabiliriz. (eski sürümlerde AS zorunluydu )
```

```
U.EMAIL MAIL, --Önemli not: AS kullanmayarakta yapabiliriz.
```

```
U.GENDER CINSIYET,
```

```
U.BIRTHDATE AS DOGUTARIHI,
```

```
A.ADDRESSTEXT AS ACIKADRES
```

```
FROM
```

```
USERS U, ADDRESS A
```

```
WHERE U.ID = A.USERID
```

```
AND U.ID=4
```

8.4_IkidenFazlaTabloKullanimi

--Benim uygulamam

```
SELECT USERS.NAMESURNAME,USERS.GENDER,
```

```
ADDRESS.ADDRESSTEXT,
```

```
COUNTRIES.COUNTRY,
```

```
CITIES.CITY
```

```
FROM
```

```
USERS,ADDRESS,COUNTRIES,CITIES
```

```
WHERE USERS.ID=ADDRESS.USERID AND COUNTRIES.ID=ADDRESS.COUNTYID AND  
CITIES.ID=ADDRESS.CITYID
```

--Dersteki uygulama

--Uygulama 1

```
SELECT
```

```
U.NAMESURNAME AS ADSOYAD,
```

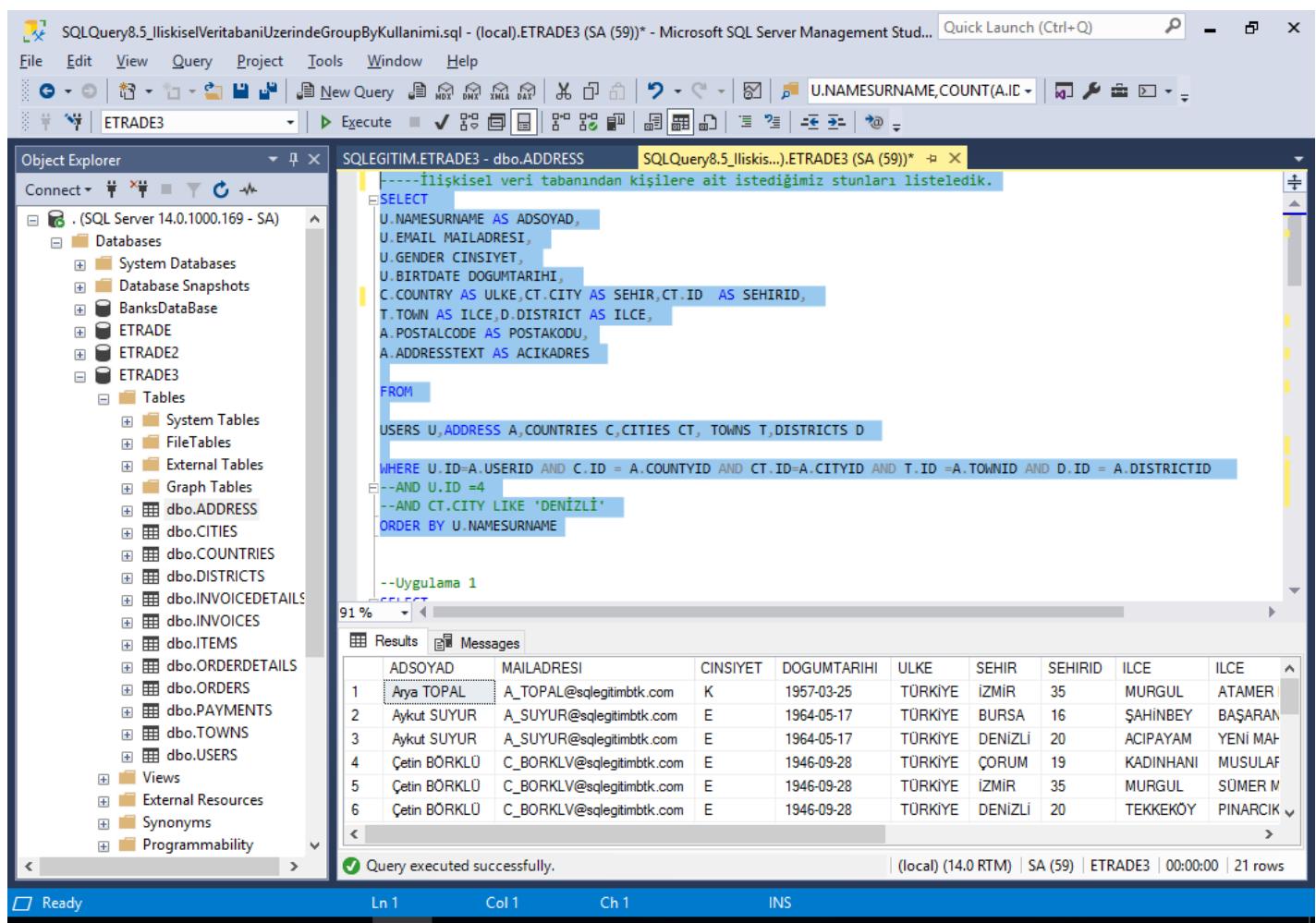
```
U.EMAIL MAILADRESI,
```

```
U.GENDER CINSIYET,
```

U.BIRTHDATE DOGUMTARIHI,
 C.COUNTRY AS ULKE,CT.CITY AS SEHIR,
 T.TOWN AS ILCE,D.DISTRICT AS ILCE,
 A.ADDRESSTEXT AS ACIKADRES
 FROM
 USERS U,ADDRESS A,COUNTRIES C,CITIES CT, TOWNS T,DISTRICTS D

 WHERE U.ID=A.USERID AND C.ID = A.COUNTYID AND CT.ID=A.CITYID AND T.ID =A.TOWNID AND D.ID =
 A.DISTRICTID
 --AND U.ID =4
 AND CT.CITY LIKE 'DENİZLİ'
 ORDER BY U.NAMESURNAME

8.5_IlliskiselVeritabaniUzerindeGroupByKullanimi



The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left lists the database structure, including tables like ADDRESS, CITIES, COUNTRIES, DISTRICTS, INVOICEDETAILS, INVOICES, ITEMS, ORDERDETAILS, ORDERS, PAYMENTS, TOWNS, and USERS. The central pane displays a query window titled 'SQLEGITIM.ETRADE3 - dbo.ADDRESS'. The query itself is:

```

--İlliskisel veri tabanından kişilere ait istediğimiz stünları listeledik.
SELECT
  U.NAMESURNAME AS ADSOYAD,
  U.EMAIL MAILADRESI,
  U.GENDER CINSIYET,
  U.BIRTHDATE DOGUMTARIHI,
  C.COUNTRY AS ULKE,CT.CITY AS SEHIR,CT.ID AS SEHIRID,
  T.TOWN AS ILCE,D.DISTRICT AS ILCE,
  A.POSTALCODE AS POSTAKODU,
  A.ADDRESSTEXT AS ACIKADRES

FROM
  USERS U,ADDRESS A,COUNTRIES C,CITIES CT, TOWNS T,DISTRICTS D

WHERE U.ID=A.USERID AND C.ID = A.COUNTYID AND CT.ID=A.CITYID AND T.ID =A.TOWNID AND D.ID = A.DISTRICTID
--AND U.ID =4
--AND CT.CITY LIKE 'DENİZLİ'
ORDER BY U.NAMESURNAME
  
```

The results pane shows a table with 21 rows of data, with the first few rows being:

	ADSOYAD	MAILADRESI	CINSIYET	DOGUMTARIHI	ULKE	SEHIR	SEHIRID	ILCE	ILCE
1	Arya TOPAL	A_TOPAL@sqlegitimblk.com	K	1957-03-25	TÜRKİYE	İZMİR	35	MURGUL	ATAMER
2	Ayuk SUYUR	A_SUYUR@sqlegitimblk.com	E	1964-05-17	TÜRKİYE	BURSA	16	ŞAHINBEY	BAŞARAN
3	Aykut SUYUR	A_SUYUR@sqlegitimblk.com	E	1964-05-17	TÜRKİYE	DENİZLİ	20	ACIPAYAM	YENİ MAH.
4	Çetin BÖRKLU	C_BORKLV@sqlegitimblk.com	E	1946-09-28	TÜRKİYE	ÇORUM	19	KADINHANI	MUSULAF
5	Çetin BÖRKLU	C_BORKLV@sqlegitimblk.com	E	1946-09-28	TÜRKİYE	İZMİR	35	MURGUL	SÜMER M
6	Çetin BÖRKLU	C_BORKLV@sqlegitimblk.com	E	1946-09-28	TÜRKİYE	DENİZLİ	20	TEKKEKOY	PINARCIK

At the bottom of the results pane, it says 'Query executed successfully.'

SQLQuery8.5_IlliskiselVeritabaniUzerindeGroupByKullanimi.sql - (local).ETRADE3 (SA (59)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

New Query DDL DML XLS XLSX XLSM XLSB Execute ✓

U.NAMESURNAME,COUNT(A.ID)

ETRADE3

Object Explorer

SQLLEGITIM.ETRADE3 - dbo.ADDRESS

```
WHERE U.ID=A.USERID AND C.ID = A.COUNTYID AND CT.ID=A.CITYID AND T.ID =A.TOWNID AND D.ID = A.DISTRICTID
--AND U.ID =4
--AND CT.CITY LIKE 'DENİZLİ'
ORDER BY U.NAMESURNAME

--Uygulama 1
--Adreslerin sayısını getirdik
SELECT
COUNT(U.NAMESURNAME) --isim stününün satır sayısı

FROM
USERS U,ADDRESS A,COUNTRIES C,CITIES CT, TOWNS T,DISTRICTS D

WHERE U.ID=A.USERID AND C.ID = A.COUNTYID AND CT.ID=A.CITYID AND T.ID =A.TOWNID AND D.ID = A.DISTRICTID
--AND U.ID =4
--AND CT.CITY LIKE 'DENİZLİ'
--ORDER BY U.NAMESURNAME

--Uygulama 2
```

Results Messages

(No column name)
1 21

Query executed successfully.

(local) (14.0 RTM) | SA (59) | ETRADE3 | 00:00:00 | 1 rows

Ready Ln 22 Col 1 Ch 1 INS

SQLQuery8.5_IlliskiselVeritabaniUzerindeGroupByKullanimi.sql - (local).ETRADE3 (SA (59))* - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

New Query Indexes DML XML DAX

ETRADE3 Execute

Object Explorer

SQLLEGITIM.ETRADE3 - dbo.ADDRESS

```

FROM
USERS U,ADDRESS A,COUNTRIES C,CITIES CT, TOWNS T,DISTRICTS D

WHERE U.ID=A.USERID AND C.ID = A.COUNTYID AND CT.ID=A.CITYID AND T.ID =A.TOWNID AND D.ID = A.DISTRICTID
--AND U.ID =4
--AND CT.CITY LIKE 'DENİZLİ'
--ORDER BY U.NAMESURNAME

--Uygulama 2
--Adresleri değilde Adresleri olan kişileri listeledik !
SELECT
U.NAMESURNAME

FROM
USERS U,ADDRESS A,COUNTRIES C,CITIES CT, TOWNS T,DISTRICTS D

WHERE U.ID=A.USERID AND C.ID = A.COUNTYID AND CT.ID=A.CITYID AND T.ID =A.TOWNID AND D.ID = A.DISTRICTID

--Uygulama 3

```

Results

NAMESURNAME
14 Soner ULGEN
15 Emirhan SELİM
16 Emirhan SELİM
17 Arya TOPAL
18 Nazlican ÖZSİM...
19 Nazlican ÖZSİM...
20 Nazlican ÖZSİM...
21 Nazlican ÖZSİM...

Query executed successfully. | (local) (14.0 RTM) | SA (59) | ETRADE3 | 00:00:00 | 21 rows

Ln 37 Col 1 Ch 1 INS

SQLQuery8.5_IlliskiselVeritabaniUzerindeGroupByKullanimi.sql - localhost.ETRADE3 (SQLLEGITIM\Administrator (55)) - Microsoft SQL Server Management Studio (Administrator)

File Edit View Query Project Tools Window Help

New Query Indexes DML XML DAX

ETRADE3 Execute

Object Explorer

SQLQuery8.5_Illiskis... (Administrator (55))

```

USERS U,ADDRESS A,COUNTRIES C,CITIES CT, TOWNS T,DISTRICTS D

WHERE U.ID=A.USERID AND C.ID = A.COUNTYID AND CT.ID=A.CITYID AND T.ID =A.TOWNID AND D.ID = A.DISTRICTID

--Uygulama 3
SELECT
DISTINCT U.NAMESURNAME --Aynı isimleri sadece 1 tane getirecek şekilde listeledik.(DISTINCT)

FROM
USERS U,ADDRESS A,COUNTRIES C,CITIES CT, TOWNS T,DISTRICTS D

WHERE U.ID=A.USERID AND C.ID = A.COUNTYID AND CT.ID=A.CITYID AND T.ID =A.TOWNID AND D.ID = A.DISTRICTID

--Uygulama 4
--birden fazla tabloda GROUP BY kullanımı
--Gruplavarak aynı isimden sadece 1 kişiyi listelemis olduk

```

Results

NAMESURNAME
1 Arya TOPAL
2 Aykut SUYUR
3 Çelin BÖRKLO
4 Emirhan SELİM
5 Ezgi KÜRTUNCU
6 Hazal ÜREĞİL
7 Nazlican ÖZSİMİTCİ
8 Onur KIRIT
9 Soner ULGEN

Query executed successfully. | localhost (14.0 RTM) | SQLLEGITIM\Administrato... | ETRADE3 | 00:00:00 | 9 rows

SQLServer - VMware Workstation 15 Player (Non-commercial use only)

Player | SQLQuery8.5_IlliskiselVeritabaniUzerindeGroupByKullanimi.sql - (local).ETRADE3 (SA (59))* - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

ETRADE3 Execute

Object Explorer

SQLLEGITIM.ETRADE3 - dbo.ADDRESS

```
WHERE U.ID=A.USERID AND C.ID = A.COUNTYID AND CT.ID=A.CITYID AND T.ID =A.TOWNID AND D.ID = A.DISTRICTID

--Uygulama 4
--birden fazla tabloda GROUP BY kullanımı
--Gruplayarak aynı isimden sadece 1 kişiyi listelemiştir
SELECT
U.NAMESURNAME

FROM
USERS U,ADDRESS A,COUNTRIES C,CITIES CT, TOWNS T,DISTRICTS D

WHERE U.ID=A.USERID AND C.ID = A.COUNTYID AND CT.ID=A.CITYID AND T.ID =A.TOWNID AND D.ID = A.DISTRICTID

GROUP BY U.NAMESURNAME --birden fazla tabloda GROUP BY kullanımı

--Uygulama 5
--Kişilerin Adres sayısı
SELECT
U.NAMESURNAME,COUNT(A.ID) AS ADRESSAYISI --ya da COUNT(U.ID) aynı sonucu alırız
```

Results

NAMESURNAME
2 Aykut SUYUR
3 Çetin BÖRKLU
4 Emirhan SELİM
5 Ezgi KÜRTÜNCÜ
6 Hazal ÜREGİL
7 Nazlıcan ÖZSİMITÇİ
8 Onur KIRIT
9 Soner ÜLGEN

Query executed successfully.

(local) (14.0 RTM) | SA (59) | ETRADE3 | 00:00:00 | 9 rows

Ready

İura yazarı

Ln 58 Col 1 Ch 1 INS

25°C Güneşli

SQLQuery8.5_IlliskiselVeritabaniUzerindeGroupByKullanimi.sql - (local).ETRADE3 (SA (59))* - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

ETRADE3 Execute

U.NAMESURNAME,COUNT(A.ID)

Object Explorer

Connect .(SQL Server 14.0.1000.169 - SA) Databases System Databases Database Snapshots BanksDataBase ETRADE ETRADE2 ETRADE3 Tables System Tables FileTables External Tables Graph Tables dbo.ADDRESS dbo.CITIES dbo.COUNTRIES dbo.DISTRICTS dbo.INVOICEDETAILS dbo.INVOICES dbo.ITEMS dbo.ORDERDETAILS dbo.ORDERS dbo.PAYMENTS dbo.TOWNS dbo.USERS Views External Resources Synonyms Programmability

SQLLEGITIM.ETRADE3 - dbo.ADDRESS

```
WHERE U.ID=A.USERID AND C.ID = A.COUNTYID AND CT.ID=A.CITYID AND T.ID =A.TOWNID AND D.ID = A.DISTRICTID  
GROUP BY U.NAMESURNAME --birden fazla tabloda GROUP BY kullanımı  
--Uygulama 5  
--Kişilerin Adres sayısı  
SELECT U.NAMESURNAME,COUNT(A.ID) AS ADRESSAYISI --ya da COUNT(U.ID) aynı sonucu alırız  
FROM USERS U,ADDRESS A,COUNTRIES C,CITIES CT, TOWNS T,DISTRICTS D  
WHERE U.ID=A.USERID AND C.ID = A.COUNTYID AND CT.ID=A.CITYID AND T.ID =A.TOWNID AND D.ID = A.DISTRICTID  
GROUP BY U.NAMESURNAME  
--Uygulama 6  
--Kişilere ait adresler kaç farklı şehirdeymiş ona bakalım  
SELECT U.NAMESURNAME,COUNT(A.ID) AS ADRESSAYISI  
--COUNT(DISTINCT CT.ID) AS SEHIRSAYISI ---
```

Results

NAMESURNAME
2 Aykut SUYUR
3 Çetin BORKLU
4 Emirhan SELIM
5 Ezgi KÖRTÜNCÜ
6 Hazal ÜREGİL
7 Nazlıcan ÖZSİMİTÇİ
8 Onur KIRIT
9 Soner ÜLGEN

Query executed successfully. | (local) (14.0 RTM) | SA (59) | ETRADE3 | 00:00:00 | 9 rows

Ready Ln 71 Col 1 Ch 1 INS

```
FROM
USERS U,ADDRESS A,COUNTRIES C,CITIES CT, TOWNS T,DISTRICTS D

WHERE U.ID=A.USERID AND C.ID = A.COUNTYID AND CT.ID=A.CITYID AND T.ID =A.TOWNID AND D.ID = A.DISTRICTID

GROUP BY U.NAMESURNAME

--Uygulama 6
--Kişilere ait adresler kaç farklı şehirdeymiş ona bakalım
SELECT
U.NAMESURNAME,COUNT(A.ID) AS ADRESSAYISI,
COUNT(DISTINCT CT.ID) AS SEHIRSAYISI ---Şehir sayısı
FROM
USERS U,ADDRESS A,COUNTRIES C,CITIES CT, TOWNS T,DISTRICTS D

WHERE U.ID=A.USERID AND C.ID = A.COUNTYID AND CT.ID=A.CITYID AND T.ID =A.TOWNID AND D.ID = A.DISTRICTID

GROUP BY U.NAMESURNAME
```

91 %

Results Messages

	NAMESURNAME	ADRESSAYISI
1	Arya TOPAL	1
2	Aykut SUYUR	2
3	Çetin BÖRKLÜ	3
4	Emirhan SELİM	2
5	Ezgi KÜRTUNCÜ	4
6	Hazal ÜREGİL	2
7	Nazlıcan ÖZSİMITÇİ	4
8	Onur KIRIT	1

✔ Query executed successfully.

(local) (14.0 RTM) | SA (59) | ETRADE3 | 00:00:00 | 9 rows

Şuanki sorguyu anlamak için öncelikle aşağıdaki tabloyu inceleyelim:

SQLEGITIM.ETRADE3 - dbo.ADDRESS SQLQuery8.5_Illiskis...).ETRADE3 (SA (59))*

```
--İlişkisel veri tabanından kişilere ait istediğimiz stnları listeledik.
SELECT
U.NAMESURNAME AS ADSOYAD,
U.EMAIL MAILADRESI,
U.GENDER CINSIYET,
U.BIRTHDATE DOGUMTARIHI,
C.COUNTRY AS ULKE,CT.CITY AS SEHIR,CT.ID AS SEHIRID,
T.TOWN AS ILCE,D.DISTRICT AS ILCE,
A.POSTALCODE AS POSTAKODU,
A.ADDRESSTEXT AS ACIKADRES

FROM
USERS U,ADDRESS A,COUNTRIES C,CITIES CT, TOWNS T,DISTRICTS D

WHERE U.ID=A.USERID AND C.ID = A.COUNTYID AND CT.ID=A.CITYID AND T.ID =A.TOWNID AND D.ID = A.DISTRICTID
--AND U.ID =4
--AND CT.CITY LIKE 'DENİZLİ'
ORDER BY U.NAMESURNAME
```

91 %

	ADSOYAD	MAILADRESI	CINSIYET	DOGUMTARIHI	ULKE	SEHIR	SEHIRID	ILCE	ILCE
1	Arya TOPAL	A_TOPAL@sqlegitimbtk.com	K	1957-03-25	TÜRKİYE	İZMİR	35	MURGUL	ATAMER
2	Aykut SUYUR	A_SUYUR@sqlegitimbtk.com	E	1964-05-17	TÜRKİYE	BURSA	16	ŞAHİNBEY	BAŞARAI
3	Aykut SUYUR	A_SUYUR@sqlegitimbtk.com	E	1964-05-17	TÜRKİYE	DENİZLİ	20	ACIPAYAM	YENİ MA
4	Çetin BÖRKÜLÜ	C_BORKLV@sqlegitimbtk.com	E	1946-09-28	TÜRKİYE	ÇORUM	19	KADINHANI	MUSULA
5	Çetin BÖRKÜLÜ	C_BORKLV@sqlegitimbtk.com	E	1946-09-28	TÜRKİYE	İZMİR	35	MURGUL	SÜMER İ
6	Çetin BÖRKÜLÜ	C_BORKLV@sqlegitimbtk.com	E	1946-09-28	TÜRKİYE	DENİZLİ	20	TEKKEKÖY	PINARCII
7	Emirhan SELİM	E_SELIM@sqlegitimbtk.com	E	1993-01-21	TÜRKİYE	BURSA	16	NAZILLİ	YALINTA
8	Emirhan SELİM	E_SELIM@sqlegitimbtk.com	F	1993-01-21	TÜRKİYE	KOCAELİ	22	İZMİR	FAHRI

Query executed successfully. | (local) (14.0 RTM) | SA (59) | ETRADE3 | 00:00:00 | 21 rows

Tablodan görüleceği üzere burada eğer biz CITY deki ID lerini sayıdırsak kişilere(Mesela Çetin Börkülü) ait sayıyı listelemiş oluruz. O zaman hemen deneyelim:

```
FROM
USERS U,ADDRESS A,COUNTRIES C,CITIES CT, TOWNS T,DISTRICTS D

WHERE U.ID=A.USERID AND C.ID = A.COUNTYID AND CT.ID=A.CITYID AND T.ID =A.TOWNID AND D.ID = A.DISTRICTID

GROUP BY U.NAMESURNAME

--Uygulama 6
--Kişilere ait adresler kaç farklı şehirdeymiş ona bakalım
SELECT
U.NAMESURNAME,COUNT(A.ID) AS ADRESSAYISI,
COUNT(DISTINCT CT.ID) AS SEHIRSAYISI ---
FROM
USERS U,ADDRESS A,COUNTRIES C,CITIES CT, TOWNS T,DISTRICTS D

WHERE U.ID=A.USERID AND C.ID = A.COUNTYID AND CT.ID=A.CITYID AND T.ID =A.TOWNID AND D.ID = A.DISTRICTID

GROUP BY U.NAMESURNAME
```

91 %

Results Messages

	NAMESURNAME	ADRESSAYISI	SEHIRSAYISI
2	Aykut SUYUR	2	2
3	Çetin BORKLU	3	3
4	Emihan SELİM	2	2
5	Ezgi KÜRTÜNCÜ	4	4
6	Hazal ÖREGİL	2	2
7	Nazlican ÖZSİMITÇİ	4	4
8	Onur KIRIT	1	1
9	Soner ÜLGEN	2	2

Query executed successfully.

(local) (14.0 RTM) | SA (59) | ETRADE3 | 00:00:00 | 9 rows

SQLEGITIM.ETRADE3 - dbo.ADDRESS SQLQuery8.5_İliskis...).ETRADE3 (SA (59))*

```
--Uygulama 7
--Kişilere ait adresler kaç farklı ülkede
SELECT
    U.NAMESURNAME, COUNT(A.ID) AS ADRESSAYISI,
    COUNT(DISTINCT CT.ID) SEHIRSAYISI, --Sehir sayısı
    COUNT(DISTINCT C.ID) AS ULKESAYISI --Ülke sayısı
FROM
    USERS U, ADDRESS A,COUNTRIES C,CITIES CT, TOWNS T,DISTRICTS D
WHERE U.ID=A.USERID AND C.ID = A.COUNTYID AND CT.ID=A.CITYID AND T.ID =A.TOWNID AND D.ID = A.DISTRICTID
GROUP BY U.NAMESURNAME
```

91 %

Results Messages

	NAMESURNAME	ADRESSAYISI	SEHIRSAYISI	ULKESAYISI
1	Aya TOPAL	1	1	1
2	Aykut SUYUR	2	2	1
3	Çetin BÖRKÜ	3	3	1
4	Emirhan SELİM	2	2	1
5	Ezgi KÜRTÜNCÜ	4	4	1
6	Hazal ÜREGİL	2	2	1
7	Nazlican ÖZSİMİTÇİ	4	4	1
8	Oya KIRIT	1	1	1

Query executed successfully. | (local) (14.0 RTM) | SA (59) | ETRADE3 | 00:00:00 | 9 rows

```
--Uygulama
--ikiden fazla adresi olanı listeleyelim
SELECT
U.NAMESURNAME,COUNT(A.ID) AS ADRESSAYISI,
COUNT(DISTINCT CT.ID) SEHIRSAYISI,
COUNT(DISTINCT C.ID) AS ULKESAYISI ---Sehir sayısı
FROM
USERS U,ADDRESS A,COUNTRIES C,CITIES CT, TOWNS T,DISTRICTS D
WHERE U.ID=A.USERID AND C.ID = A.COUNTYID AND CT.ID=A.CITYID AND T.ID =A.TOWNID AND D.ID = A.DISTRICTID
GROUP BY U.NAMESURNAME
HAVING COUNT(A.ID)>2 -- ikiden fazla adresi olanı listeledik
```

91 %

Results Messages

	NAMESURNAME	ADRESSAYISI	SEHIRSAYISI	ULKESAYISI
1	Çetin BÖRKLU	3	3	1
2	Ezgi KÜRTUNCÜ	4	4	1
3	Nazlican ÖZSİMITÇİ	4	4	1

Query executed successfully.

(local) (14.0 RTM) | SA (59) | ETRADE3 | 00:00:00 | 3 rows

```
SQLLEGITIM.ETRADE3 - dbo.ADDRESS SQLQuery8.5_İliskis...).ETRADE3 (SA (59))* ↵ ×
GROUP BY U.NAMESURNAME
HAVING COUNT(A.ID)>2 -- ikiden fazla adresi olanı listeledik

--Uygulama 9
--Kullanıcıların adresleri hangi şehirlerde ? bunu listeleyelim
SELECT
CT.CITY
FROM
USERS U,ADDRESS A,COUNTRIES C,CITIES CT, TOWNS T,DISTRICTS D
WHERE U.ID=A.USERID AND C.ID = A.COUNTYID AND CT.ID=A.CITYID AND T.ID =A.TOWNID AND D.ID = A.DISTRICTID
GROUP BY CT.CITY

91 % ▶
Results Messages


|    | CITY      |
|----|-----------|
| 7  | KIRIKKALE |
| 8  | KOCAELİ   |
| 9  | KONYA     |
| 10 | MERSİN    |
| 11 | NIĞDE     |
| 12 | TEKİRDAĞ  |
| 13 | TRABZON   |
| 14 | Yozgat    |

✔ Query executed successfully. | (local) (14.0 RTM) | SA (59) | ETRADE3 | 00:00:00 | 14 rows
```

```
--Uygulama 10
--Şehirlerinde kullanıcı adres sayısı
SELECT
CT.CITY,COUNT(U.ID) AS KULLANICISAYISI
FROM
USERS U,ADDRESS A,COUNTRIES C,CITIES CT, TOWNS T,DISTRICTS D
WHERE U.ID=A.USERID AND C.ID = A.COUNTYID AND CT.ID=A.CITYID AND T.ID =A.TOWNID AND D.ID = A.DISTRICTID
GROUP BY CT.CITY
```

91 %

Results Messages

	CITY	KULLANICISAYISI
1	BURDUR	1
2	BURSA	2
3	ÇORUM	1
4	DENİZLİ	2
5	ISPARTA	2
6	İZMİR	3
7	KIRIKKALE	1
8	KOCAELİ	1

Query executed successfully.

(local) (14.0 RTM) | SA (59) | ETRADE3 | 00:00:00 | 14 rows

```
--Uygulama 11  
--Şehirlerinde kullanıcı ilçe sayısı  
SELECT  
CT.CITY,COUNT(U.ID) AS KULLANICISAYISI,COUNT(D.ID) AS ILCESAYISI  
FROM  
USERS U,ADDRESS A,COUNTRIES C,CITIES CT, TOWNS T,DISTRICTS D  
WHERE U.ID=A.USERID AND C.ID = A.COUNTYID AND CT.ID=A.CITYID AND T.ID =A.TOWNID AND D.ID = A.DISTRICTID  
GROUP BY CT.CITY
```

91 %

Results Messages

	CITY	KULLANICISAYISI	ILCESAYISI
1	BURDUR	1	1
2	BURSA	2	2
3	ÇORUM	1	1
4	DENİZLİ	2	2
5	ISPARTA	2	2
6	İZMİR	3	3
7	KIRIKKALE	1	1
8	KOCAELİ	1	1

Query executed successfully.

(local) (14.0 RTM) | SA (59) | ETRADE3 | 00:00:00 | 14 rows

8.6_JoinİslemleriGiris

Veri tabanını birleştirmek için WHERE ile kullandığımız teknik eski bir teknik JOIN ise bunun yeni bir tekniğidir.

Bir tabloda olup başka tabloda olmayanları bağlayabiliyoruz ve kayıt varsa getirip yoksa getirmeyebiliyoruz. İşte biz bu yüzden JOIN i kullanıyoruz.

İLİŞKİSEL TABLOLARDAN VERİ SORGULAMA

SELECT

A.KOLON1,A.KOLON2,B.KOLON3,B.KOLON4
FROM TABLO1 A,TABLO2 B
WHERE A.PK_KOLON=B.FK_KOLON

8.7_JoinUygulamaGiris

İlk olarak Where ile tabloları birlestirelim:

--WHERE ile bir tablo yapalim daha sonra ayınısını join ile yapalim

```

SELECT
U.NAMESURNAME, U.EMAIL,
A.ADDRESSTEXT,
C.COUNTRY,
CT.CITY,
T.TOWN,
D.DISTRICT

FROM
USERS U, ADDRESS A,COUNTRIES C,CITIES CT, TOWNS T,DISTRICTS D

WHERE
U.ID=A.USERID AND C.ID = A.COUNTYID AND CT.ID=A.CITYID AND T.ID = A.TOWNID AND D.ID = A.DISTRICTID

```

110 %

Results Messages

	NAMESURNAME	EMAIL	ADDRESSTEXT	COUNTRY	CITY	TOWN	DISTRICT
1	Onur KIRIT	O_KIRIT@sqllegitimbtk.com	EVCILER-GÜLTEPE MAH.PAPATYA SOKAK NO:363 114571 ÇAYIL...	TÜRKİYE	YOZGAT	ÇAYIRLI	EVCILER-GÜLTEPE MAH.
2	Ayku SUYUR	A_SUYUR@sqllegitimbtk.com	BAŞARAN MAH.1.GÜZ SOKAK NO:314 85995 OSMANGAZİ/BUR...	TÜRKİYE	BURSA	ŞAHİNBEY	BAŞARAN MAH.
3	Ayku SUYUR	A_SUYUR@sqllegitimbtk.com	YENİ MAH.BADEMKENT KÜME EVLERİ SOKAK NO:662 88689 A...	TÜRKİYE	DENİZLİ	ACİPAYAM	YENİ MAH.
4	Hazal ÜREGİL	H_VREGİL@sqllegitimbtk.com	ŞİHBEDRETTİN MAH.396. SOKAK NO:737 101280 ILGIN/KONYA	TÜRKİYE	KONYA	İLİÇ	ŞİHBEDRETTİN MAH.
5	Hazal ÜREGİL	H_VREGİL@sqllegitimbtk.com	CAMINUR MAH.0802. SOKAK NO:161 95504 TARSUS/MERSİN	TÜRKİYE	MERSİN	TEKMAN	CAMINUR MAH.
6	Ezgi KÜRTUNCU	E_IBUKVRTVNCV@sqllegitimbtk.com	MİSİNLI-KALE MAH.İSTİKLAL CADDESİ SOKAK NO:649 110954 Ç...	TÜRKİYE	TEKİRDAĞ	ÇORUM MERKEZ	MİSİNLI-KALE MAH.
7	Ezgi KÜRTUNCU	E_IBUKVRTVNCV@sqllegitimbtk.com	ULUAĞAÇ KÖYÜ MAH.MERKEZ C MEVKİ SOKAK NO:209 112450...	TÜRKİYE	TRABZON	ORTA	ULUAĞAÇ KÖYÜ
8	Ezgi KÜRTUNCU	E_IBUKVRTVNCV@sqllegitimbtk.com	DOĞANKENT-AKSARAY MAH.ATATÜRK CADDESİ SOKAK NO:7...	TÜRKİYE	YOZGAT	SUSURLUK	DOĞANKENT-AKSARAY MAH.
9	Ezgi KÜRTUNCU	E_IBUKVRTVNCV@sqllegitimbtk.com	SEVGİ MAH.87. SOKAK NO:377 96596 GAZİEMİR/İZMİR	TÜRKİYE	İZMİR	GÜZELBAHÇE	SEVGİ MAH.
10	Çetin BÖRKLU	C_BORKLV@sqllegitimbtk.com	MUSULAR KÖYÜ MAH.MERKEZ MEVKİ SOKAK NO:44 88131 İSK...	TÜRKİYE	ÇORUM	KADINHANI	MUSULAR KÖYÜ
11	Çetin BÖRKLU	C_BORKLV@sqllegitimbtk.com	SÜMER MAH.440. SOKAK NO:506 96717 KONAK/İZMİR	TÜRKİYE	İZMİR	MURGUL	SÜMER MAH.
12	Çetin BÖRKLU	C_BORKLV@sqllegitimbtk.com	PINARCİK MAH.BALTALAR SOKAK NO:609 89159 SERİNHİSAR/...	TÜRKİYE	DENİZLİ	TEKKEKÖY	PINARCİK MAH.

Query executed successfully. | (local) (14.0 RTM) | SA (54) | ETRADE3 | 00:00:00 | 21 rows

Aynı tabloyu join ile yapalim:

--Aynı tabloyu join ile birleştirip listeleyelim

```

SELECT
U.NAMESURNAME,U.EMAIL,
A.ADDRESSTEXT,
C.COUNTRY,
CT.CITY,
T.TOWN,
D.DISTRICT

FROM
USERS U
JOIN ADDRESS A ON A.USERID=U.ID
JOIN COUNTRIES C ON C.ID=A.COUNTYID
JOIN CITIES CT ON CT.ID=A.CITYID
JOIN TOWNS T ON T.ID=A.TOWNID
JOIN DISTRICTS D ON D.ID=A.DISTRICTID

```

100 %

Results Messages

	NAMESURNAME	EMAIL	ADDRESSTEXT	COUNTRY	CITY	TOWN	DISTRICT
1	Onur KIRIT	O_KIRIT@sqllegitimbtk.com	EVCILER-GÜLTEPE MAH.PAPATYA SOKAK NO:363 114571 ÇAYIL...	TÜRKİYE	YOZGAT	ÇAYIRLI	EVCILER-GÜLTEPE MAH.
2	Ayku SUYUR	A_SUYUR@sqllegitimbtk.com	BAŞARAN MAH.1.GÜZ SOKAK NO:314 85995 OSMANGAZİ/BUR...	TÜRKİYE	BURSA	ŞAHİNBEY	BAŞARAN MAH.
3	Ayku SUYUR	A_SUYUR@sqllegitimbtk.com	YENİ MAH.BADEMKENT KÜME EVLERİ SOKAK NO:662 88689 A...	TÜRKİYE	DENİZLİ	ACİPAYAM	YENİ MAH.
4	Hazal ÜREGİL	H_VREGİL@sqllegitimbtk.com	ŞİHBEDRETTİN MAH.396. SOKAK NO:737 101280 ILGIN/KONYA	TÜRKİYE	KONYA	İLİÇ	ŞİHBEDRETTİN MAH.
5	Hazal ÜREGİL	H_VREGİL@sqllegitimbtk.com	CAMINUR MAH.0802. SOKAK NO:161 95504 TARSUS/MERSİN	TÜRKİYE	MERSİN	TEKMAN	CAMINUR MAH.
6	Ezgi KÜRTUNCU	E_IBUKVRTVNCV@sqllegitimbtk.com	MİSİNLI-KALE MAH.İSTİKLAL CADDESİ SOKAK NO:649 110954 Ç...	TÜRKİYE	TEKİRDAĞ	ÇORUM MERKEZ	MİSİNLI-KALE MAH.
7	Ezgi KÜRTUNCU	E_IBUKVRTVNCV@sqllegitimbtk.com	ULUAĞAÇ KÖYÜ MAH.MERKEZ C MEVKİ SOKAK NO:209 112450...	TÜRKİYE	TRABZON	ORTA	ULUAĞAÇ KÖYÜ
8	Ezgi KÜRTUNCU	E_IBUKVRTVNCV@sqllegitimbtk.com	DOĞANKENT-AKSARAY MAH.ATATÜRK CADDESİ SOKAK NO:7...	TÜRKİYE	YOZGAT	SUSURLUK	DOĞANKENT-AKSARAY MAH.
9	Ezgi KÜRTUNCU	E_IBUKVRTVNCV@sqllegitimbtk.com	SEVGİ MAH.87. SOKAK NO:377 96596 GAZİEMİR/İZMİR	TÜRKİYE	İZMİR	GÜZELBAHÇE	SEVGİ MAH.
10	Çetin BÖRKLU	C_BORKLV@sqllegitimbtk.com	MUSULAR KÖYÜ MAH.MERKEZ MEVKİ SOKAK NO:44 88131 İSK...	TÜRKİYE	ÇORUM	KADINHANI	MUSULAR KÖYÜ
11	Çetin BÖRKLU	C_BORKLV@sqllegitimbtk.com	SÜMER MAH.440. SOKAK NO:506 96717 KONAK/İZMİR	TÜRKİYE	İZMİR	MURGUL	SÜMER MAH.
12	Çetin BÖRKLU	C_BORKLV@sqllegitimbtk.com	PINARCİK MAH.BALTALAR SOKAK NO:609 89159 SERİNHİSAR/...	TÜRKİYE	DENİZLİ	TEKKEKÖY	PINARCİK MAH.

Query executed successfully. | (local) (14.0 RTM) | SA (54) | ETRADE3 | 00:00:00 | 21 rows

SQLQuery8.5_Iliskis...).ETRADE3 (SA (55))

SQLQuery8.7_JoinU...ETRADE3 (SA (54))*

```
--Hangi şehirde kaç kullanıcı var onu listeleyelim
SELECT
CT.CITY,
COUNT(U.ID) AS KULLANICISAYISI
FROM
USERS U
JOIN ADDRESS A ON A.USERID=U.ID
JOIN COUNTRIES C ON C.ID=A.COUNTYID
JOIN CITIES CT ON CT.ID=A.CITYID
JOIN TOWNS T ON T.ID=A.TOWNID
JOIN DISTRICTS D ON D.ID=A.DISTRICTID
GROUP BY CT.CITY
```

110 %

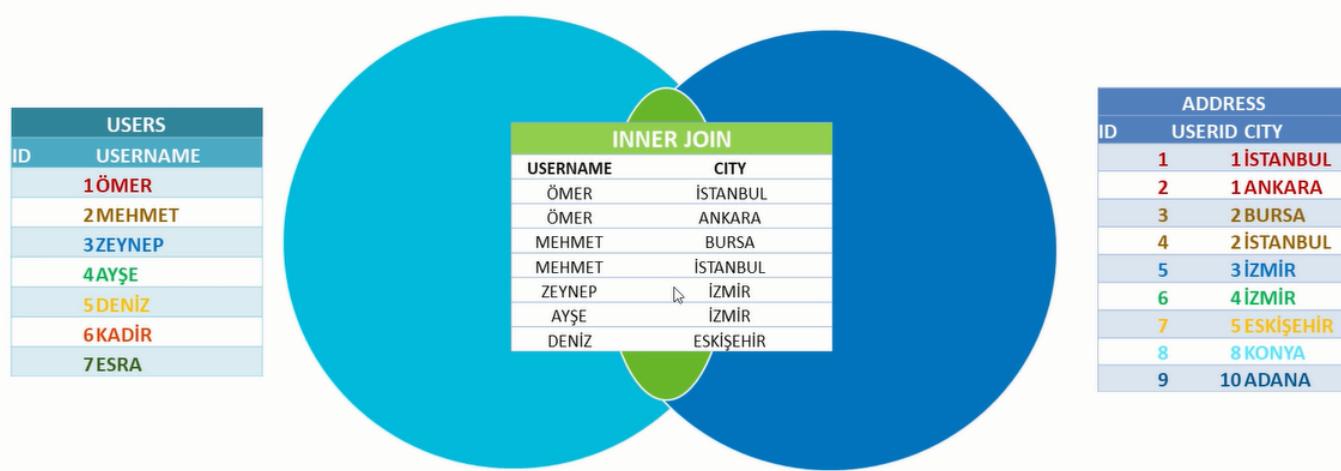
Results Messages

	CITY	KULLANICISAYISI
4	DENİZLİ	2
5	İSPARTA	2
6	İZMİR	3
7	KIRIKKALE	1
8	KOCAELİ	1
9	KONYA	1
10	MERSİN	1
11	NİĞDE	1
12	TEKİRDAĞ	2
13	TRABZON	1
14	YOZGAT	2

Query executed successfully.

(local) (14.0 RTM) | SA (54) | ETRADE3 | 00:00:00 | 14 rows

8.8_Innerjoin



8.9_LefJoin

Sol tarafda o kayıttan yoksa bile boş olarak getirir !

JOIN TÜRLERİ

LEFT (OUTER) JOIN

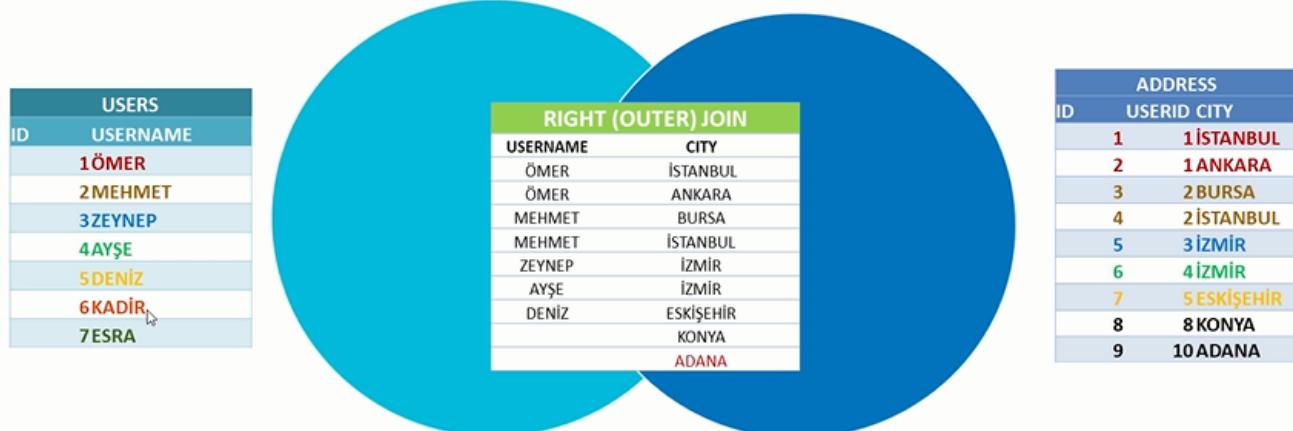


8.10_RightJoin

Sağ tarafda o kayıttan yoksa bile boş olarak getirir !

JOIN TÜRLERİ

RIGHT (OUTER) JOIN

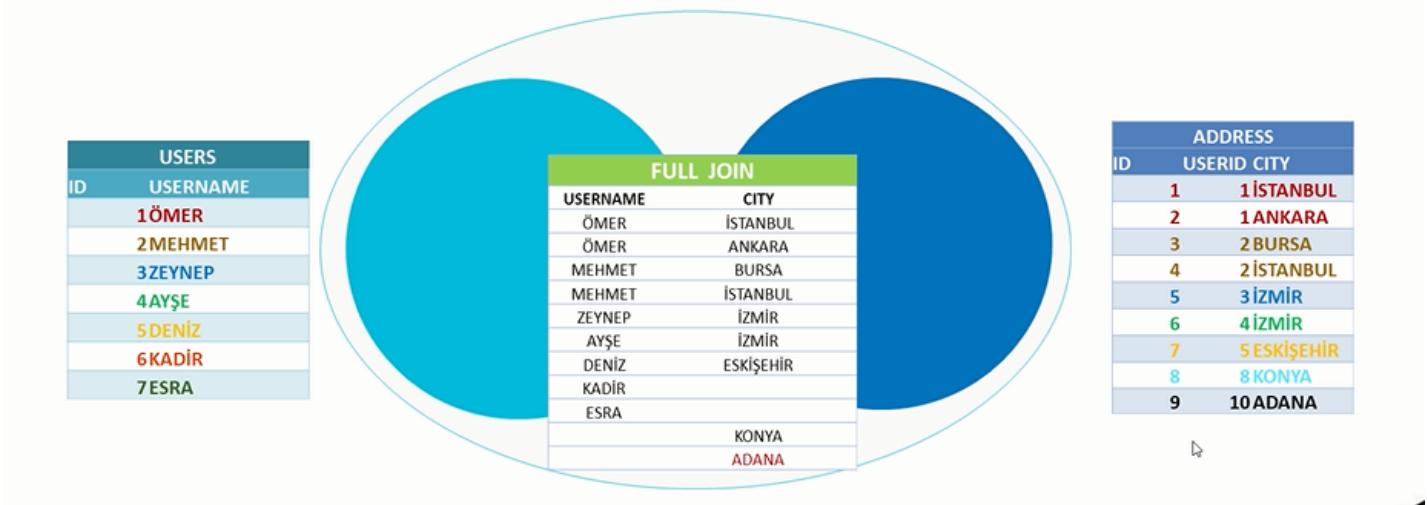


8.11_FullJoin

Adındananda anlaşılacağı üzere ilişki olsun olmasın her iki tablodan o kayıtları listeler !

Aşağıda görüldüğü gibi USERS tablosunda Kadir varken ADRESS tablosundan Kadire ait Adress olmadığı için boş listeledi.Aynı şekilde ADDRESS tablosunda Konya ve Adana varken bu işehirlere ait USERS tablosundan herhangi bir kayıtlı olan kullanıcı yok ancak Konya adresi ve Adana adresi listelendi Username de doğal olarak boş listelendi !

JOIN TÜRLERİ FULL JOIN



8.12_JoinTurleriUygulama

TEST adında bir database oluşturduk ve bu database içinde USERS ve ADDRESS adında basit iki tane tablo oluşturduk.

SQLLEGITIM.TEST - dbo.ADDRESS SQLLEGITIM.TEST - dbo.USERS

	ID	USERNAME_
▶	1	ÖMER
	2	MEHMET
	3	ZEYNEP
	4	AYŞE
	5	DENİZ
	6	KADİR
	7	ESRA
*	NULL	NULL

◀ ▶ | 1 of 7 | ▶ ▶ ⏪ | Cell is Read Only.

SQLLEGITIM.TEST - dbo.ADDRESS SQLLEGITIM.TEST - dbo.USERS

	ID	USERID	ADRESSTEXT
	1	1	İSTANBUL
	2	1	ANKARA
	3	2	BURSA
	4	2	İSTANBUL
	5	3	İZMİR
	6	4	İZMİR
	7	5	ESKİŞEHİR
▶	8	8	KONYA
	9	10	ADANA
*	NULL	NULL	NULL

◀ ▶ | 8 of 9 | ▶ ▶ ⏪ |

Yukarda da tablolara ait verileri görmekteyiz. Şimdi ise bu tabloları uygulamalarla birleştirelim:

SELECT * FROM USERS

SELECT * FROM ADDRESS

SQLEGITIM.TEST - dbo.USERS SQLQuery8.12_JoinT...cal).TEST (SA (56))*

```
--Uygulama1
--Join
SELECT
U.ID,U.USERNAME_,A.ADRESSTEXT
FROM USERS U JOIN ADDRESS A ON A.USERID=U.ID

--Uygulama2
```

110 %

Results Messages

ID	USERNAME_	ADRESSTEXT
1	ÖMER	İSTANBUL
2	ÖMER	ANKARA
3	MEHMET	BURSA
4	MEHMET	İSTANBUL
5	ZEYNEP	İZMİR
6	AYŞE	İZMİR
7	DENİZ	ESKİŞEHİR

Query executed successfully. | (local) (14.0 RTM) | SA (56) | TEST | 00:00:00 | 7 rows

SOLEGITIM.TEST - dbo.USERS SQLQuery8.12_JoinT...cal).TEST (SA (56))*

```
--Uygulama2
--Join ile aynıdır! hiçbir sonuç değişmez !
--InnerJoin
SELECT
U.ID,U.USERNAME_,A.ADRESSTEXT
FROM USERS U JOIN ADDRESS A ON A.USERID=U.ID
```

110 %

Results Messages

ID	USERNAME_	ADRESSTEXT
1	ÖMER	İSTANBUL
2	ÖMER	ANKARA
3	MEHMET	BURSA
4	MEHMET	İSTANBUL
5	ZEYNEP	İZMİR
6	AYŞE	İZMİR
7	DENİZ	ESKİŞEHİR

Query executed successfully. | (local) (14.0 RTM) | SA (56) | TEST | 00:00:00 | 7 rows

LEFT JOIN de ise USERS (soldaki) da olup ADDRESS (sağdaki) te olmayanda gelecektir.
Diğer bir deyişle (Soldaki)USERS tablonun tüm elemanlarını listeler. ADDRESS(Sağdaki) tablosundan sadece USERS (solda) da bulunanları listeler.

The screenshot shows a SQL Server Management Studio window with the following details:

- Query Editor Title: SQLEGITIM.TEST - dbo.ADDRESS
- Query Editor Content:

```
--Uygulama3
--Leftjoin
--LEFT JOIN de ise USERS da olup ADDRESS te olmayanda gelecektir
SELECT
U.ID,U.USERNAME_,A.ADRESSTEXT
FROM USERS U LEFT JOIN ADDRESS A ON A.USERID=U.ID
```
- Results Grid:

ID	USERNAME_	ADRESSTEXT
3	2	MEHMET
4	2	MEHMET
5	3	ZEYNEP
6	4	AYŞE
7	5	DENİZ
8	6	KADİR
9	7	ESRA
- Status Bar: Query executed successfully. | (local) (14.0 RTM) | SA (56) | TEST | 00:00:00 | 9 rows

Peki sorguda sağ ve solun hangi tablo olduğunu nasıl anlarız ?

The screenshot shows a SQL Server Management Studio window with the following details:

- Query Editor Title: SQLEGITIM.TEST - dbo.ADDRESS
- Query Editor Content:

```
--Uygulama3
--Leftjoin
--LEFT JOIN de ise USERS da olup ADDRESS te olmayanda gelecektir
SELECT
U.ID,U.USERNAME_,A.ADRESSTEXT
FROM USERS U LEFT JOIN ADDRESS A ON A.USERID=U.ID
```
- Results Grid:

ID	USERNAME_	ADRESSTEXT
3	2	MEHMET
4	2	MEHMET
5	3	ZEYNEP
6	4	AYŞE
7	5	DENİZ
8	6	KADİR
9	7	ESRA
- Annotations: The word "sol" is written over the word "LEFT JOIN" in the query, and the word "sağ" is written over the word "ADDRESS" in the query.
- Status Bar: Query executed successfully. | (local) (14.0 RTM) | SA (56) | TEST | 00:00:00 | 9 rows

LEFT JOIN Sağında kalan sağ tablo, solunda kalan ise sol tablo olarak düşünebiliriz !

SQLLEGITIM.TEST - dbo.ADDRESS SQLQuery8.12_JoinT...cal).TEST (SA (56))*

```
--Uygulama3
--Leftjoin
--LEFT JOIN de ise USERS da olup ADDRESS te olmayanda gelecektir
SELECT
U.ID,U.USERNAME_,A.ADRESSTEXT
FROM USERS U LEFT JOIN ADDRESS A ON A.USERID=U.ID
```

110 %

Results Messages

	ID	USERNAME_	ADRESSTEXT
3	2	MEHMET	BURSA
4	2	MEHMET	İSTANBUL
5	3	ZEYNEP	İZMİR
6	4	AYŞE	İZMİR
7	5	DENİZ	ESKİSEHIR
8	6	KADİR	NULL
9	7	ESRA	NULL

Query executed successfully. | (local) (14.0 RTM) | SA (56) | TEST | 00:00:00 | 9 rows

SQLLEGITIM.TEST - dbo.ADDRESS SQLQuery8.12_JoinT...cal).TEST (SA (56))*

```
--Uygulama4
--RIGHTJOIN
--RIGHT JOIN de ise ADDRESS da olup USERS te olmayanda gelecektir
SELECT
U.ID,U.USERNAME_,A.ADRESSTEXT
FROM USERS U RIGHT JOIN ADDRESS A ON A.USERID=U.ID
```

110 %

Results Messages

	ID	USERNAME_	ADRESSTEXT
3	2	MEHMET	BURSA
4	2	MEHMET	İSTANBUL
5	3	ZEYNEP	İZMİR
6	4	AYŞE	İZMİR
7	5	DENİZ	ESKİSEHIR
8	NULL	NULL	KONYA
9	NULL	NULL	ADANA

Query executed successfully. | (local) (14.0 RTM) | SA (56) | TEST | 00:00:00 | 9 rows

The screenshot shows a SQL Server Management Studio interface with three tabs at the top: 'SQLEGITIM.TEST - dbo.ADDRESS', 'SQLQuery8.12_JoinT...cal).TEST (SA (56))' (active), and 'SQLEGITIM.TEST - dbo.USERS'. The query window contains the following code:

```
--Uygulama5
--FULL JOIN
--FULL JOIN de ise her iki tabloda arasında ilişki olsada olmasada hepsini listeler !
SELECT
U.ID,U.USERNAME_,A.ADRESSTEXT
FROM USERS U FULL JOIN ADDRESS A ON A.USERID=U.ID
```

The results pane shows a table with 11 rows of data:

ID	USERNAME_	ADRESSTEXT
3	2	MEHMET BURSA
4	2	MEHMET İSTANBUL
5	3	ZEYNEP İZMİR
6	4	AYŞE İZMİR
7	5	DENİZ ESKİSEHIR
8	6	KADIR NULL
9	7	ESRA NULL
10	N...	KONYA
11	N...	ADANA

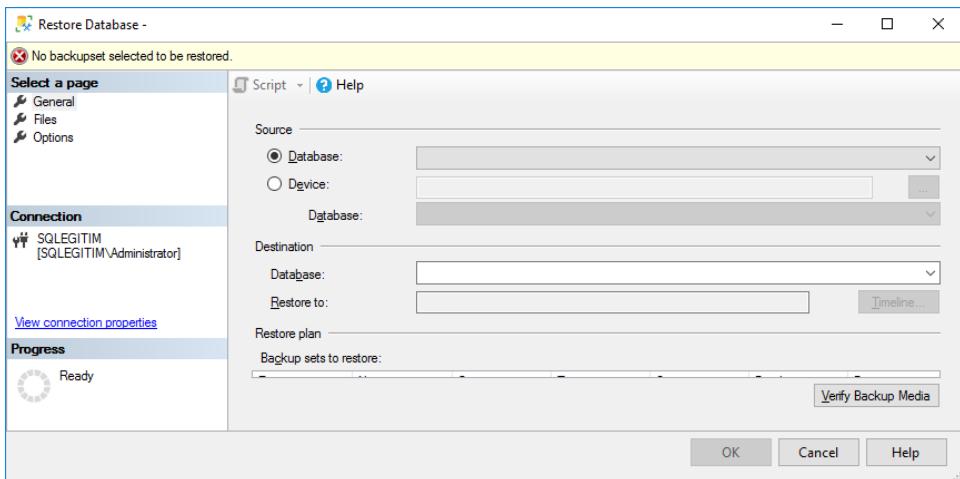
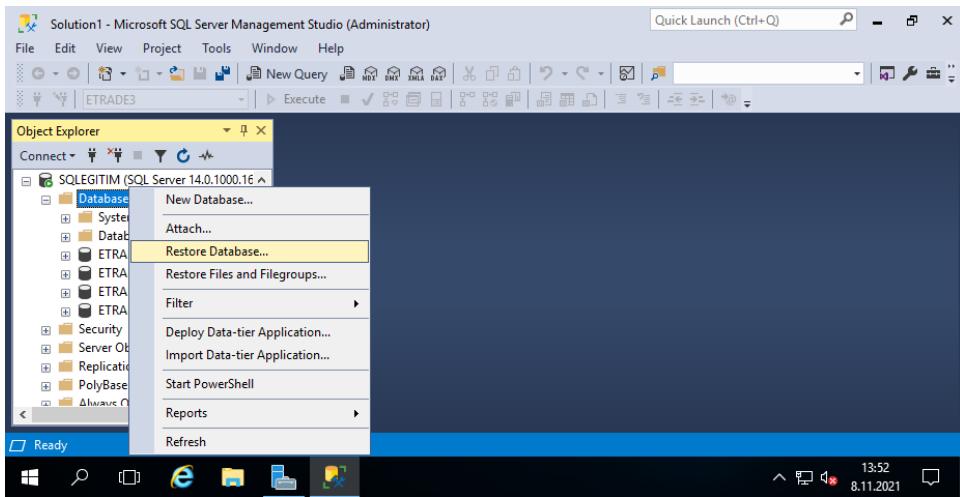
At the bottom of the results pane, it says 'Query executed successfully.' and '(local) (14.0 RTM) | SA (56) | TEST | 00:00:00 | 11 rows'.

8.13_Bolumsonu_Join

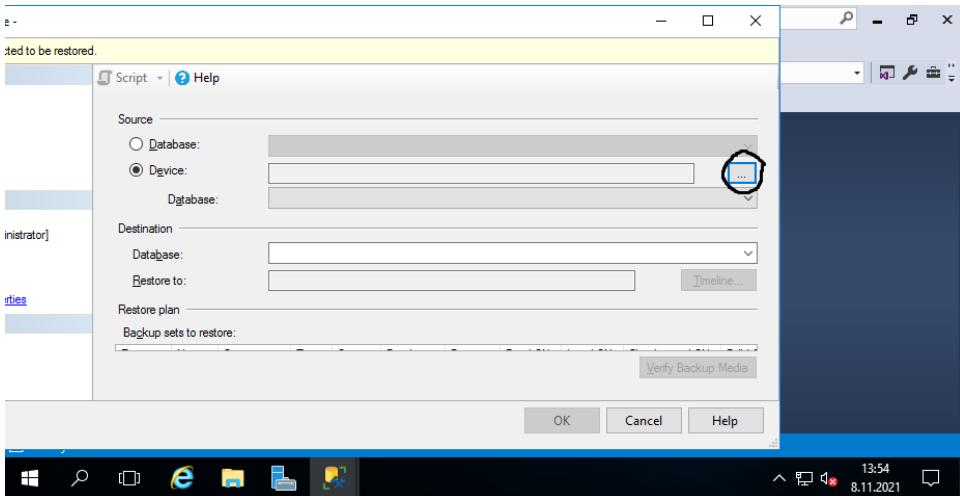
- İlişkisel veritabanında birden fazla tablo nasıl birleştirilir?
- JOIN kavramı ne demektir?
- INNER JOIN, LEFT JOIN, RIGHT JOIN, FULL JOIN Kavramları arasında ne farklar vardır?
- Master Detay özelliğindeki tablolar nasıl bağlanarak soru çekilir?
- ALIAS nedir?
- JOIN kullandığımız tablolarda ALIAS kullanımı nasıl yapılır?
- JOIN kullandığımız tablolarda GROUP BY nasıl kullanılır?

9.1_VeriTabaniRestoreEtme

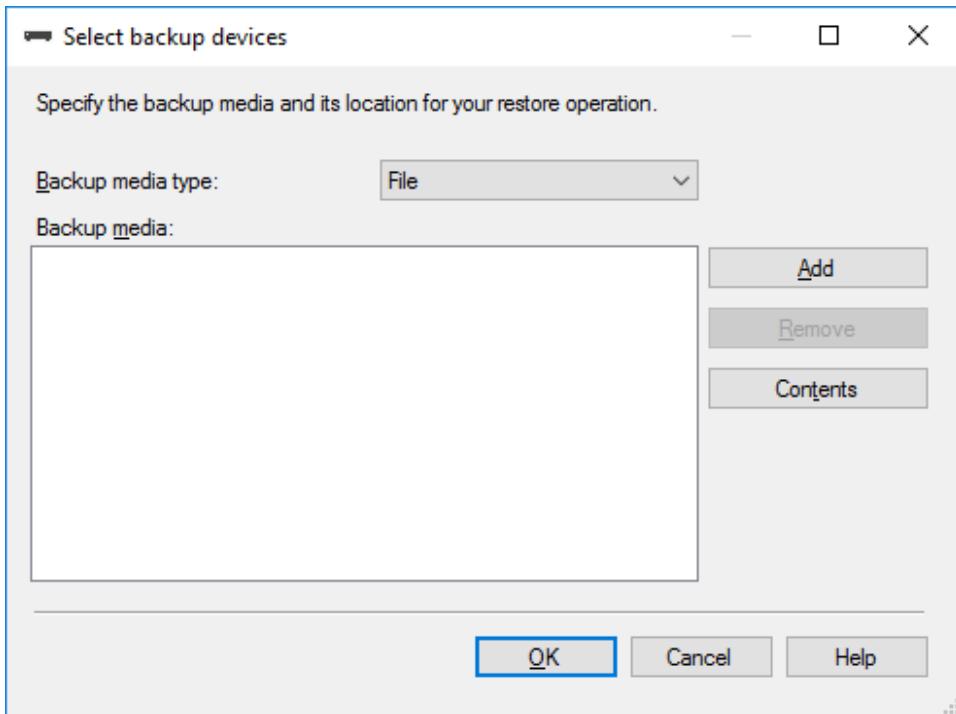
Örnek olarak burada gerçek verileri barındıran ETRADE4.bak dosyasını Restore Edelim.(DataBaseye eklemek).



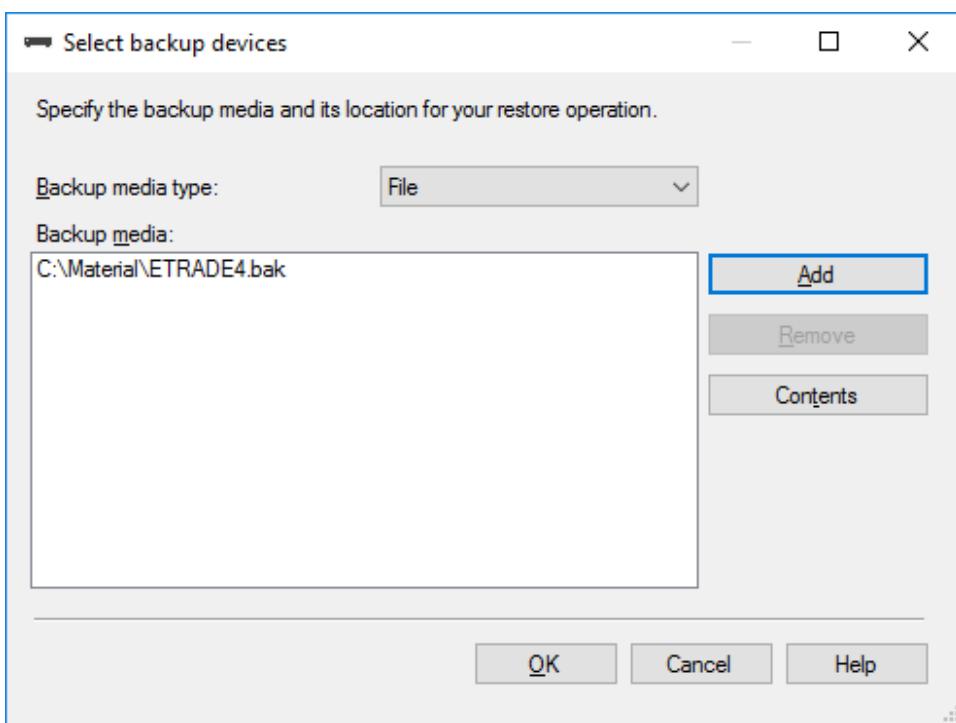
Burada Device seçip "..."(üç nokta) ya basalım.



Gelen pencereden ETRADE4.bak databasemizi add ile ekleyelim



Göründüğü gibi ekledik.



Daha sonra “OK” tuşu basarak ETRADE4’ü DATABASE’ye eklemiş oluyoruz.

9.2_GercekUygulamalarOrnek1

```

--Uygulama1
--Bu uygulamada örneğin Talha CANIKARA adlı müşteriye ait sipariş bilgisini getirelim:
--Kullanıcı Adı Ad Soyad il İlçe Semt Açık Adres Sipariş ID Tarih Toplam Tutar Ödeme Tarihi
--Banka Onay Kodu Fatura Tarihi Kargo Fiş No
--SELECT
    U.USERNAME_ AS KULLANICIADI, U.NAMESURNAME AS ADSOYAD,
    CT.CITY AS IL,T.TOWN AS ILCE,D.DISTRICT AS SEMT, A.ADDRESSTEXT AS ACIKADRES,
    O.ID SIPARISID,O.DATE_ AS TARİH,O.TOTALPRICE AS TOPLAMTUTAR,
    P.DATE_ AS ODEMETARIHI, P.APPROVECODE AS BANKAONAYKODU,
    I.DATE_ AS FATURATARİHİ, I.CARGOFICHENO AS KARGOFISNO
FROM ORDERS O
INNER JOIN USERS U ON U.ID=O.USERID
INNER JOIN ADDRESS A ON A.ID=O.ADDRESSID
INNER JOIN CITIES CT ON CT.ID=A.CITYID
INNER JOIN TOWNS T ON T.ID=A.TOWNID
INNER JOIN DISTRICTS D ON D.ID=A.DISTRICTID
INNER JOIN PAYMENTS P ON P.ORDERID=O.ID
INNER JOIN INVOICES I ON I.ORDERID=O.ID
WHERE U.NAMESURNAME='Talha CANIKARA'

```

Results Messages

KULLANICIADI	ADSOYAD	IL	ILCE	SEMT	ACIKADRES	SIPARISID	TARIH	TOPLAMTUTAR	ODEMETA
T_CANIKARA	Talha CANIKARA	MANİSA	SALİHLİ	ADALA-ATATÜRK MAH.	ADALA-ATATÜRK MAH.DAR SOKAK NO:612 71244 SALİHLİ/...	13162	2019-12-06 17:10:08.000	106.7563	2019-12-06
T_CANIKARA	Talha CANIKARA	MANİSA	SALİHLİ	ADALA-ATATÜRK MAH.	ADALA-ATATÜRK MAH.DAR SOKAK NO:612 71244 SALİHLİ/...	13330	2019-11-01 11:16:42.000	497.5150	2019-11-01
T_CANIKARA	Talha CANIKARA	MANİSA	SALİHLİ	ADALA-ATATÜRK MAH.	ADALA-ATATÜRK MAH.DAR SOKAK NO:612 71244 SALİHLİ/...	19065	2019-08-26 06:08:06.000	389.1587	2019-08-26
T_CANIKARA	Talha CANIKARA	MANİSA	SALİHLİ	ADALA-ATATÜRK MAH.	ADALA-ATATÜRK MAH.DAR SOKAK NO:612 71244 SALİHLİ/...	19503	2019-04-01 23:32:40.000	196.7921	2019-04-01
T_CANIKARA	Talha CANIKARA	MANİSA	SALİHLİ	ADALA-ATATÜRK MAH.	ADALA-ATATÜRK MAH.DAR SOKAK NO:612 71244 SALİHLİ/...	29595	2020-01-14 22:43:50.000	357.5202	2020-01-14

Query executed successfully. | SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrat... | ETRADE4 | 00:00:00 | 5 rows

Gerçek senaryoya yakın oluşturmuş bu tablodaki kişilere rastgele random olarak sipariş atama işlemini RANDOM_ORDER_GENERATOR.sql adlı dosyayı SSMS(SQL Server Management Studio) ile Open>File ile açalım:

9_2_GerçekUygulamalarOrnek1.sql - SQLEGITIM.ETRADE4 (SQLEGITIM\Administrator (54)) - Microsoft SQL Server Management Studio (Administrator)

File Edit View Query Project Tools Window Help

File... Ctrl+O

KULLANICIADI	ADSOYAD	IL	ILCE	SEMT	ACIKADRES	SIPARISID	TARIH	TOPLAMTUTAR	ODEMETA
T_CANIKARA	Talha CANIKARA	MANİSA	SALİHLİ	ADALA-ATATÜRK MAH.	ADALA-ATATÜRK MAH.DAR SOKAK NO:612 71244 SALİHLİ/...	13162	2019-12-06 17:10:08.000	106.7563	2019-12-06
T_CANIKARA	Talha CANIKARA	MANİSA	SALİHLİ	ADALA-ATATÜRK MAH.	ADALA-ATATÜRK MAH.DAR SOKAK NO:612 71244 SALİHLİ/...	13330	2019-11-01 11:16:42.000	497.5150	2019-11-01
T_CANIKARA	Talha CANIKARA	MANİSA	SALİHLİ	ADALA-ATATÜRK MAH.	ADALA-ATATÜRK MAH.DAR SOKAK NO:612 71244 SALİHLİ/...	19065	2019-08-26 06:08:06.000	389.1587	2019-08-26
T_CANIKARA	Talha CANIKARA	MANİSA	SALİHLİ	ADALA-ATATÜRK MAH.	ADALA-ATATÜRK MAH.DAR SOKAK NO:612 71244 SALİHLİ/...	19503	2019-04-01 23:32:40.000	196.7921	2019-04-01
T_CANIKARA	Talha CANIKARA	MANİSA	SALİHLİ	ADALA-ATATÜRK MAH.	ADALA-ATATÜRK MAH.DAR SOKAK NO:612 71244 SALİHLİ/...	29595	2020-01-14 22:43:50.000	357.5202	2020-01-14

Query executed successfully. | SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrat... | ETRADE4 | 00:00:00 | 5 rows

Göründüğü gibi:

The screenshot shows the Microsoft SQL Server Management Studio (SSMS) interface. The title bar reads "RANDOM_ORDER_GENERATOR.sql - SQLEGITIM.ETRADE4 (SQLEGITIM\Administrator (56)) - Microsoft SQL...". The menu bar includes File, Edit, View, Query, Project, Tools, Window, and Help. The toolbar has various icons for file operations like New Query, Save, Print, and Database objects like Indexes, DMVs, XML, and Data. The Object Explorer on the left shows the database structure for "ETRADE4", including tables like ADDRESS, CITIES, COUNTRIES, DISTRICTS, INVOICEDETAILS, INVOICES, ITEMS, ORDERDETAILS, ORDERS, PAYMENTS, TOWNS, and USERS. The main pane displays a T-SQL script for generating random orders:

```
DECLARE @USERID AS INT
DECLARE @DATE AS DATETIME
DECLARE @TOTALPRICE AS FLOAT
DECLARE @PRICE AS FLOAT
DECLARE @AMOUNT AS INT
DECLARE @INVOICEID AS INT

WHILE @I<10000
BEGIN
    SELECT TOP 1 @USERID=ID FROM ETRADE4.DBO.USERS ORDER BY NEWID()
    SELECT TOP 1 @ADDRESSID=ID FROM ADDRESS WHERE USERID=@USERID ORDER BY NEWID()
WHILE @ADDRESSID IS NULL
BEGIN
    SELECT TOP 1 @USERID=ID FROM ETRADE4.DBO.USERS ORDER BY NEWID()
    SELECT TOP 1 @ADDRESSID=ID FROM ADDRESS WHERE USERID=@USERID ORDER BY NEWID()
```

Burdaki 10000 rakamı ile aslında rastgele müsteriye rastgele zamanda,tarihte,mekanda,miktarda sipariş üretebiliriz.

9.3_GercekUygulamalarOrnek2

```
SQLLEGITIM.ETRAD.dbo.ORDERDETAILS 9.3_GercekUygulama...dministrator (52)* SQLEGITIM.ETRADE4 - dbo.ITEMS
--Uygulama1
--Bu örnekte ise itemlere(Ürünlere) ait Ürün kodu,brim miktarı,fiyatı gibi bilgileri listeleyelim.

--Kullanıcı Adı Ad Soyad İl İlçe Semt Açık Adres Sipariş ID Tarih Toplam Tutar Ödeme Tarihi
--Banka Onay Kodu Fatura Tarihini Kargo Fiş No
SELECT
    U.USERNAME_ AS KULLANICIADI, U.NAMESURNAME AS ADSOYAD,
    CT.CITY AS IL,T.TOWN AS ILCE,D.DISTRICT AS SEMT, A.ADDRESSTEXT AS ACIKADRES,
    O.ID SIPARISID,O.DATE_ AS TARIH,O.TOTALPRICE AS TOPLAMTUTAR,
    P.DATE_ AS ODEMETARIHI, P.APPROVECODE AS BANKAONAYKODU,
    I.DATE_ AS FATURATARİHİ, I.CARGOFICHENO AS KARGOFISNO,
    ITM.ITEMCODE AS URUNKODU,ITM.ITEMNAME AS URUNADI,OD.AMOUNT AS MIKTAR, OD.UNITPRICE AS BIRIMFIYAT,
    OD.LINETOTAL AS SATIRTOPLAMI
FROM ORDERS O
INNER JOIN USERS U ON U.ID=O.USERID
INNER JOIN ADDRESS A ON A.ID=O.ADDRESSID
INNER JOIN CITIES CT ON CT.ID=A.CITYID
INNER JOIN TOWNS T ON T.ID=A.TOWNID
INNER JOIN DISTRICTS D ON D.ID=A.DISTRICTID
INNER JOIN PAYMENTS P ON P.ORDERID=O.ID
INNER JOIN INVOICES I ON I.ORDERID=O.ID
INNER JOIN ORDERDETAILS OD ON OD.ORDERID=O.ID
INNER JOIN ITEMS ITM ON ITM.ID = OD.ITEMID
WHERE U.NAMESURNAME='Ceyda GEZGİNÇİ'

91 % ▾
Results Messages


|   | KULLANICIADI | ADSOYAD        | IL         | ILCE             | SEMT                       | ACIKADRES                                             | SIPARISID | TARIH                   |
|---|--------------|----------------|------------|------------------|----------------------------|-------------------------------------------------------|-----------|-------------------------|
| 1 | C_GEZGINCI   | Ceyda GEZGİNÇİ | BURSA      | MUSTAFAKEMALPAŞA | TATKAVAKLI-CUMHURİYET MAH. | TATKAVAKLI-CUMHURİYET MAH. 8. SOKAK NO:435 86723 M... | 3515      | 2019-10-24 23:16:05.000 |
| 2 | C_GEZGINCI   | Ceyda GEZGİNÇİ | BURSA      | MUSTAFAKEMALPAŞA | TATKAVAKLI-CUMHURİYET MAH. | TATKAVAKLI-CUMHURİYET MAH. 8. SOKAK NO:435 86723 M... | 3515      | 2019-10-24 23:16:05.000 |
| 3 | C_GEZGINCI   | Ceyda GEZGİNÇİ | DIYARBAKIR | SUR              | KÖPRÜBAŞI MAH.             | KÖPRÜBAŞI MAH.KÖPRÜBAŞI 1 SOKAK NO:320 118857 SU...   | 24601     | 2019-10-31 03:28:45.000 |
| 4 | C_GEZGINCI   | Ceyda GEZGİNÇİ | DIYARBAKIR | SUR              | KÖPRÜBAŞI MAH.             | KÖPRÜBAŞI MAH.KÖPRÜBAŞI 1 SOKAK NO:320 118857 SU...   | 24601     | 2019-10-31 03:28:45.000 |


```

9.4_GercekUygulamalarOrnek3

Uygulama 1

```

SQLLEGITIM.ETRAD...dbo.ORDERDETAILS      9.3_GercekUygulama..dministrator (55)      9.4_GercekUygulama..dministrator (52)*  ↗ X
----Uygulama: ŞEHİRLERE GÖRE TOPLAM VERİLEN SİPARİŞ MIKTARI

SELECT
CT.CITY AS SEHIRADI,
SUM(OD.LINETOTAL) AS TOPLAMSIPARIS_TUTARI,
SUM(OD.AMOUNT) AS TOPLAMSIPARIS_ADEDI,
COUNT(OD.ID) AS TOPLAMSIPARIS_SAYISI

FROM ORDERS O
INNER JOIN USERS U ON U.ID=O.USERID
INNER JOIN ADDRESS A ON A.ID=O.ADDRESSID
INNER JOIN CITIES CT ON CT.ID=A.CITYID
INNER JOIN TOWNS T ON T.ID=A.TOWNID
INNER JOIN DISTRICTS D ON D.ID=A.DISTRICTID
INNER JOIN PAYMENTS P ON P.ORDERID=O.ID
INNER JOIN INVOICES I ON I.ORDERID=O.ID
INNER JOIN ORDERDETAILS OD ON OD.ORDERID=O.ID
INNER JOIN ITEMS ITM ON ITM.ID = OD.ITEMID

--WHERE O.ID=26121 --BALIKESİRE ŞEHİRİNÉ AİT ID

GROUP BY CITY

```

91 %

Results Messages

	SEHIRADI	TOPLAMSIPARIS_TUTARI	TOPLAMSIPARIS_ADEDI	TOPLAMSIPARIS_SAYISI
1	İSTANBUL	1306114.7096	90125	16384
2	YALOVA	67499.5495	4551	842
3	ARTVIN	33254.3655	2649	500
4	ELAZİĞ	80710.9651	5823	1050
5	RİZE	47474.4273	3382	610
6	TRABZON	161163.7223	10819	1951
7	ÇEDİZCAN	67715.6272	4790	860

Query executed successfully. | SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 81 rows

Uygulama 2

```

SQLLEGITIM.ETRAD...dbo.ORDERDETAILS      9.3_GercekUygulama..dministrator (55)      9.4_GercekUygulama..dministrator (52)*  ↗ X
----Uygulama2: ŞEHİRLERE GÖRE TOPLAM VERİLEN SİPARİŞ MIKTARI

SELECT
CT.CITY AS SEHIRADI,
SUM(OD.LINETOTAL) AS TOPLAMSIPARIS_TUTARI,
SUM(OD.AMOUNT) AS TOPLAMSIPARIS_ADEDI,
COUNT(OD.ID) AS TOPLAMSIPARIS_SAYISI

FROM ORDERS O
INNER JOIN USERS U ON U.ID=O.USERID
INNER JOIN ADDRESS A ON A.ID=O.ADDRESSID
INNER JOIN CITIES CT ON CT.ID=A.CITYID
INNER JOIN TOWNS T ON T.ID=A.TOWNID
INNER JOIN DISTRICTS D ON D.ID=A.DISTRICTID
INNER JOIN PAYMENTS P ON P.ORDERID=O.ID
INNER JOIN INVOICES I ON I.ORDERID=O.ID
INNER JOIN ORDERDETAILS OD ON OD.ORDERID=O.ID
INNER JOIN ITEMS ITM ON ITM.ID = OD.ITEMID

--WHERE O.ID=26121 --BALIKESİRE ŞEHİRİNÉ AİT ID

GROUP BY CITY
--ORDER BY CITY --Şehirleri A dan Z ye sıraladık.
--ORDER BY SUM(OD.LINETOTAL) DESC --Toplam sipariş tutarı en yüksekten en düşüğe göre sıraladık


```

91 %

Results Messages

	SEHIRADI	TOPLAMSIPARIS_TUTARI	TOPLAMSIPARIS_ADEDI	TOPLAMSIPARIS_SAYISI
1	İSTANBUL	1306114.7096	90125	16384
2	İZMİR	831469.4527	60375	11006
3	ANKARA	557886.1088	39040	7042
4	BURSA	506566.8119	34673	6306
5	ADANA	463018.7600	31322	5738
6	KONYA	450798.5604	31593	5724
7	ANTALYA	472212.6272	20649	5569

Query executed successfully. | SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 00 rows

Uygulama 3

```

SQLLEGITIM.ETRAD..dbo.ORDERDETAILS      9.3_GercekUygulama..dministrator (55)      9.4_GercekUygulama..dministrator (52)*  ✎ ✖
----Uygulama2: ŞEHİRLERE GÖRE TOPLAM VERİLEN SİPARİŞ MIKTARI

SELECT
CT.CITY AS SEHIRADI,
SUM(OD.LINETOTAL) AS TOPLAMSIPARIS_TUTARI,
SUM(OD.AMOUNT) AS TOPLAMSIPARIS_ADEDI, --Siparişteki ürün sayısı
COUNT(OD.ID) AS TOPLAMSIPARIS_SAYISI

FROM ORDERS O
INNER JOIN USERS U ON U.ID=O.USERID
INNER JOIN ADDRESS A ON A.ID=O.ADDRESSID
INNER JOIN CITIES CT ON CT.ID=A.CITYID
INNER JOIN TOWNS T ON T.ID=A.TOWNID
INNER JOIN DISTRICTS D ON D.ID=A.DISTRICTID
INNER JOIN PAYMENTS P ON P.ORDERID=O.ID
INNER JOIN INVOICES I ON I.ORDERID=O.ID
INNER JOIN ORDERDETAILS OD ON OD.ORDERID=O.ID
INNER JOIN ITEMS ITM ON ITM.ID = OD.ITEMID

--WHERE O.ID=26121 --BALIKESİRE ŞEHİRNE AİT ID
GROUP BY CT.CITY
--ORDER BY CITY --Şehirleri A dan Z ye sıraladık.
--ORDER BY SUM(OD.LINETOTAL) DESC --Toplam sipariş tutarı en yüksekten en düşüğe göre sıraladık
--Yada stün ismini yazarak sıralayabiliriz:
ORDER BY 2 DESC

```

91 %

Results Messages

	SEHIRADI	TOPLAMSIPARIS_SAYISI
1	İSTANBUL	16384
2	YALOVA	842
3	ARTVIN	500
4	ELAZİĞ	1050
5	RİZE	610
6	TRABZON	1951
7	ŞEHİRLER	000

Query executed successfully. | SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 81 row

```

SQLLEGITIM.ETRAD..dbo.ORDERDETAILS      9.3_GercekUygulama..dministrator (55)      9.4_GercekUygulama..dministrator (52)*  ✎ ✖
----Uygulama2: ŞEHİRLERE GÖRE TOPLAM VERİLEN SİPARİŞ MIKTARI

SELECT
CT.CITY AS SEHIRADI,
SUM(OD.LINETOTAL) AS TOPLAMSIPARIS_TUTARI,
SUM(OD.AMOUNT) AS TOPLAMSIPARIS_ADEDI,
COUNT(OD.ID) AS TOPLAMSIPARIS_SAYISI

FROM ORDERS O
INNER JOIN USERS U ON U.ID=O.USERID
INNER JOIN ADDRESS A ON A.ID=O.ADDRESSID
INNER JOIN CITIES CT ON CT.ID=A.CITYID
INNER JOIN TOWNS T ON T.ID=A.TOWNID
INNER JOIN DISTRICTS D ON D.ID=A.DISTRICTID
INNER JOIN PAYMENTS P ON P.ORDERID=O.ID
INNER JOIN INVOICES I ON I.ORDERID=O.ID
INNER JOIN ORDERDETAILS OD ON OD.ORDERID=O.ID
INNER JOIN ITEMS ITM ON ITM.ID = OD.ITEMID

--WHERE O.ID=26121 --BALIKESİRE ŞEHİRNE AİT ID
GROUP BY CITY
--ORDER BY CITY --Şehirleri A dan Z ye sıraladık.
--ORDER BY SUM(OD.LINETOTAL) DESC --Toplam sipariş tutarı en yüksekten en düşüğe göre sıraladık
--Yada stün ismini yazarak sıralayabiliriz:
ORDER BY 2 DESC

```

91 %

Results Messages

	SEHIRADI	TOPLAMSIPARIS_TUTARI	TOPLAMSIPARIS_ADEDI	TOPLAMSIPARIS_SAYISI
1	İSTANBUL	1306114.7096	90125	16384
2	İZMİR	831469.4527	60375	11006
3	ANKARA	557886.1088	39040	7042
4	BURSA	50566.8119	34673	6306
5	ADANA	463018.7600	31322	5738
6	KONYA	450798.5604	31593	5724
7	ANTALYA	402210.6272	29649	5620

Query executed successfully. | SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 81 row

9.5_GercekUygulamalarOrnek4

Uygulama

```

9.5_GercekUygulama...dministrator (56) 9.5_GercekUygulama...dministrator (56) SOLEGITIM.ETRADE4 - dbo.ITEMS SOLEGITIM.ETRAD...dbo.ORDERDETAILS SOLEGITIM.ETRADE4 - dbo.ORDERS
----Uygulama: ÜRÜN KATEGORİLERİNE GÖRE SİPARİŞ DAĞILIMI
--Örneğin Ev kategorisine göre Toplam sipariş tutarı,adedi,Sayısı ' listeleyelim.
SELECT
    ITM.CATEGORY1,
    SUM(OD.LINETOTAL) AS TOPLAMSIPARIS_TUTARI,
    SUM(OD.AMOUNT) AS TOPLAMSIPARIS_ADEDI, --Siparişteki ürün sayısı
    COUNT(OD.ID) AS TOPLAMSIPARIS_SAYISI

    FROM ORDERS O
        INNER JOIN USERS U ON U.ID=O.USERID
        INNER JOIN ADDRESS A ON A.ID=O.ADDRESSID
        INNER JOIN CITIES CT ON CT.ID=A.CITYID
        INNER JOIN TOWNS T ON T.ID=A.TOWNID
        INNER JOIN DISTRICTS D ON D.ID=A.DISTRICTID
        INNER JOIN PAYMENTS P ON P.ORDERID=O.ID
        INNER JOIN INVOICES I ON I.ORDERID=O.ID
        INNER JOIN ORDERDETAILS OD ON OD.ORDERID=O.ID
        INNER JOIN ITEMS ITM ON ITM.ID = OD.ITEMID
WHERE ITM.CATEGORY1='EV' --Ev kategorisi için sorguladık
GROUP BY ITM.CATEGORY1 --Ürün katetorilerine göre grupladık
ORDER BY 2 DESC

107 %
Results Messages
CATEGORY1 TOPLAMSIPARIS_TUTARI TOPLAMSIPARIS_ADEDI TOPLAMSIPARIS_SAYISI
1 EV 3413224.6555 338911 61576

```

Query executed successfully. | localhost (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 1 rows

Uygulama 2

```

SQLEGITIM.ETRAD...dbo.ORDERDETAILS 9.5_GercekUygulama...dministrator (59)* 9.5_GercekUygulama...dministrator (59)* SOLEGITIM.ETRADE4 - dbo.ITEMS SOLEGITIM.ETRAD...dbo.ORDERDETAILS SOLEGITIM.ETRADE4 - dbo.ORDERS
----Uygulama 2: ÜRÜN KATEGORİLERİNE GÖRE SİPARİŞ DAĞILIMI
--Örneğin KOZMETİK kategorisine göre Toplam sipariş tutarı,adedi,Sayısı ve Ortalama brim fiyatı ' listeleyelim.
SELECT
    ITM.CATEGORY1,ITM.CATEGORY2,ITM.CATEGORY3,ITM.CATEGORY4,
    SUM(OD.LINETOTAL) AS TOPLAMSIPARIS_TUTARI,
    SUM(OD.AMOUNT) AS TOPLAMSIPARIS_ADEDI, --Siparişteki ürün sayısı
    COUNT(OD.ID) AS TOPLAMSIPARIS_SAYISI,
    SUM(OD.LINETOTAL)/SUM(OD.AMOUNT) AS ORTALAMABIRIMFIYAT --ORTALAMA BİRİM FİYATI

    FROM ORDERS O
        INNER JOIN USERS U ON U.ID=O.USERID
        INNER JOIN ADDRESS A ON A.ID=O.ADDRESSID
        INNER JOIN CITIES CT ON CT.ID=A.CITYID
        INNER JOIN TOWNS T ON T.ID=A.TOWNID
        INNER JOIN DISTRICTS D ON D.ID=A.DISTRICTID
        INNER JOIN PAYMENTS P ON P.ORDERID=O.ID
        INNER JOIN INVOICES I ON I.ORDERID=O.ID
        INNER JOIN ORDERDETAILS OD ON OD.ORDERID=O.ID
        INNER JOIN ITEMS ITM ON ITM.ID = OD.ITEMID
WHERE ITM.CATEGORY1='KOZMETIK'
GROUP BY ITM.CATEGORY1,ITM.CATEGORY2,ITM.CATEGORY3,ITM.CATEGORY4
ORDER BY 2 DESC

109 %
Results Messages
CATEGORY1 CATEGORY2 CATEGORY3 CATEGORY4 TOPLAMSIPARIS_TUTARI TOPLAMSIPARIS_ADEDI TOPLAMSIPARIS_SAYISI ORTALAMABIRIMFIYAT
7 KOZMETIK YUZ BAKIM TIRAS MAL... TIRAS BIC... 71735.1227 2390 436 30.014695
8 KOZMETIK YUZ BAKIM TIRAS MAL... TIRAS SAB... 426.6177 129 25 3.307113
9 KOZMETIK YUZ BAKIM TIRAS MAL... TIRAS LOS... 581.8428 25 6 23.273712
10 KOZMETIK YUZ BAKIM TIRAS MAL... TIRAS KOP... 15588.5687 1443 262 10.802888
11 KOZMETIK YUZ BAKIM DUDAK KO... DUDAK KR... 7778.3139 418 73 18.608406

```

Query executed successfully. | SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 99 rows

9.6_GercekUygulamalarOrnek5

Uygulama 1

--Uygulama: TARİHE GÖRE SİPARİŞ DAĞILIMI
--Örneğin tarihe göre Toplam sipariş tutarı,adedi,Sayısını listeleyelim.

```

SELECT
    CONVERT(DATE,O.DATE_) TARIH, --tarih-saat bilgisinin sadece tarih kısmını aldık.
    SUM(OD.LINETOTAL) AS TOPLAMSIPARIS_TUTARI,
    SUM(OD.AMOUNT) AS TOPLAMSIPARIS_MIKTAR,
    COUNT(OD.ID) AS TOPLAMSIPARIS_SAYISI

FROM ORDERS O
    INNER JOIN USERS U ON U.ID=O.USERID
    INNER JOIN ADDRESS A ON A.ID=O.ADDRESSID
    INNER JOIN CITIES CT ON CT.ID=A.CITYID
    INNER JOIN TOWNS T ON T.ID=A.TOWNID
    INNER JOIN DISTRICTS D ON D.ID=A.DISTRICTID
    INNER JOIN PAYMENTS P ON P.ORDERID=O.ID
    INNER JOIN INVOICES I ON I.ORDERID=O.ID
    INNER JOIN ORDERDETAILS OD ON OD.ORDERID=O.ID
    INNER JOIN ITEMS ITM ON ITM.ID = OD.ITEMID
GROUP BY CONVERT(DATE,O.DATE_)
ORDER BY CONVERT(DATE,O.DATE_)
```

107 %

Results Messages

	TARIH	TOPLAMSIPARIS_TUTARI	TOPLAMSIPARIS_MIKTAR	TOPLAMSIPARIS_SAYISI
1	2019-01-01	37471.8724	1810	347
2	2019-01-02	29414.3349	2054	371
3	2019-01-03	24784.6718	1772	342
4	2019-01-04	27935.3684	1982	369
5	2019-01-05	32337.8971	2460	457
6	2019-01-06	33079.0468	2100	372

Query executed successfully. | localhost (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 450 rows

Uygulama 2

--Uygulama 2: AYLARA GÖRE SİPARİŞ DAĞILIMI
--AYLARA göre Toplam sipariş tutarı,adedi,Sayısını listeleyelim.

```

SELECT
    DATEPART(MONTH,O.DATE_) AY, --tarih-saat bilgisinin sadece "AY" kısmını aldık.
    SUM(OD.LINETOTAL) AS TOPLAMSIPARIS_TUTARI,
    SUM(OD.AMOUNT) AS TOPLAMSIPARIS_MIKTAR,
    COUNT(OD.ID) AS TOPLAMSIPARIS_SAYISI

FROM ORDERS O
    INNER JOIN USERS U ON U.ID=O.USERID
    INNER JOIN ADDRESS A ON A.ID=O.ADDRESSID
    INNER JOIN CITIES CT ON CT.ID=A.CITYID
    INNER JOIN TOWNS T ON T.ID=A.TOWNID
    INNER JOIN DISTRICTS D ON D.ID=A.DISTRICTID
    INNER JOIN PAYMENTS P ON P.ORDERID=O.ID
    INNER JOIN INVOICES I ON I.ORDERID=O.ID
    INNER JOIN ORDERDETAILS OD ON OD.ORDERID=O.ID
    INNER JOIN ITEMS ITM ON ITM.ID = OD.ITEMID
GROUP BY DATEPART(MONTH,O.DATE_)
ORDER BY DATEPART(MONTH,O.DATE_)
```

107 %

Results Messages

	AY	TOPLAMSIPARIS_TUTARI	TOPLAMSIPARIS_MIKTAR	TOPLAMSIPARIS_SAYISI
1	1	1811589.3330	125346	22695
2	2	1750204.9713	119220	21627
3	3	1665039.5341	114330	20821
4	4	860202.6280	60010	10907
5	5	885191.3542	63491	11526
6	6	869006.0401	62022	11267

Query executed successfully. | localhost (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 12 rows

Uygulama 3

```

9.6_GercekUygulama..dministrator (56)  ➤ X
----Uygulama 3: AYLARA GÖRE SİPARİŞ DAĞILIMI
--Her yıla ait AYLARA göre Toplam sipariş tutarı,adedi,Sayısını ' listeleyelim.

SELECT
    DATEPART(YEAR,O.DATE_) YIL, --tarih-saat bilgisinin sadece "YIL" kısmını aldık.
    DATEPART(MONTH,O.DATE_) AY, --tarih-saat bilgisinin sadece "AY" kısmını aldık.
    SUM(OD.AMOUNT) AS TOPLAMSIPARIS_MIKTAR,
    COUNT(OD.ID) AS TOPLAMSIPARIS_SAYISI

FROM ORDERS O
    INNER JOIN USERS U ON U.ID=O.USERID
    INNER JOIN ADDRESS A ON A.ID=O.ADDRESSID
    INNER JOIN CITIES CT ON CT.ID=A.CITYID
    INNER JOIN TOWNS T ON T.ID=A.TOWNID
    INNER JOIN DISTRICTS D ON D.ID=A.DISTRICTID
    INNER JOIN PAYMENTS P ON P.ORDERID=O.ID
    INNER JOIN INVOICES I ON I.ORDERID=O.ID
    INNER JOIN ORDERDETAILS OD ON OD.ORDERID=O.ID
    INNER JOIN ITEMS ITM ON ITM.ID = OD.ITEMID
GROUP BY DATEPART(YEAR,O.DATE_),DATEPART(MONTH,O.DATE_)
ORDER BY DATEPART(YEAR,O.DATE_),DATEPART(MONTH,O.DATE_)

109 %

```

Results Messages

	YIL	AY	TOPLAMSIPARIS_MIKTAR	TOPLAMSIPARIS_SAYISI
1	2019	1	63529	11485
2	2019	2	60144	10864
3	2019	3	62984	11441
4	2019	4	60010	10907
5	2019	5	63491	11526
6	2019	6	62022	11267
7	2019	7	63410	11501

Query executed successfully. | SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 15 rows

Uygulama 4

```
9.6_GercekUygulama...dministrator (56)* X
----Uygulama 4: AYLARA GÖRE SİPARİŞ DAĞILIMI
--Simdide ayların sayıları yerine isimlerini getirelim
SELECT
    DATEPART(YEAR,O.DATE_) YIL, --tarih-saat bilgisinin sadece "YIL" kısmını aldık.
    DATEPART(MONTH,O.DATE_) AY, --tarih-saat bilgisinin sadece "AY" kısmını aldık.

    -->>>Aşağıda görüldüğü gibi sayılar yerine ayların isimlerini bu şekilde listeleyebiliriz.
    CASE
        WHEN DATEPART(MONTH,O.DATE_) =1 THEN 'OCAK'
        WHEN DATEPART(MONTH,O.DATE_) =2 THEN 'ŞUBAT'
        WHEN DATEPART(MONTH,O.DATE_) =3 THEN 'MART'
        WHEN DATEPART(MONTH,O.DATE_) =4 THEN 'NİSAN'
        WHEN DATEPART(MONTH,O.DATE_) =5 THEN 'MAYIS'
        WHEN DATEPART(MONTH,O.DATE_) =6 THEN 'HAZİRAN'
        WHEN DATEPART(MONTH,O.DATE_) =7 THEN 'TEMMUZ'
        WHEN DATEPART(MONTH,O.DATE_) =8 THEN 'AĞUSTOS'
        WHEN DATEPART(MONTH,O.DATE_) =9 THEN 'EYLÜL'
        WHEN DATEPART(MONTH,O.DATE_) =10 THEN 'EKİM'
        WHEN DATEPART(MONTH,O.DATE_) =11 THEN 'KASIM'
        WHEN DATEPART(MONTH,O.DATE_) =12 THEN 'ARALIK'
    END AS AYADI, --yeni bir kolon ile ay isimlerini listeledik !
    SUM(OD.AMOUNT) AS TOPLAMSIPARIS_MIKTAR,
    COUNT(OD.ID) AS TOPLAMSIPARIS_SAYISI

    FROM ORDERS O
    INNER JOIN USERS U ON U.ID=O.USERID
    INNER JOIN ADDRESS A ON A.ID=O.ADDRESSID
    INNER JOIN CITIES CT ON CT.ID=A.CITYID
    INNER JOIN TOWNS T ON T.ID=A.TOWNID
    INNER JOIN DISTRICTS D ON D.ID=A.DISTRICTID
    INNER JOIN PAYMENTS P ON P.ORDERID=O.ID
    INNER JOIN INVOICES I ON I.ORDERID=O.ID
    INNER JOIN ORDERDETAILS OD ON OD.ORDERID=O.ID
    INNER JOIN ITEMS ITM ON ITM.ID = OD.ITEMID
    GROUP BY DATEPART(YEAR,O.DATE_),DATEPART(MONTH,O.DATE_)
    ORDER BY DATEPART(MONTH,O.DATE_),DATEPART(YEAR,O.DATE_)

74 % < >
Results Messages


|   | YIL  | AY | AYADI | TOPLAMSIPARIS_MIKTAR | TOPLAMSIPARIS_SAYISI |
|---|------|----|-------|----------------------|----------------------|
| 4 | 2020 | 2  | ŞUBAT | 59076                | 10763                |
| 5 | 2019 | 3  | MART  | 62984                | 11441                |
| 6 | 2020 | 3  | MART  | 51346                | 9380                 |

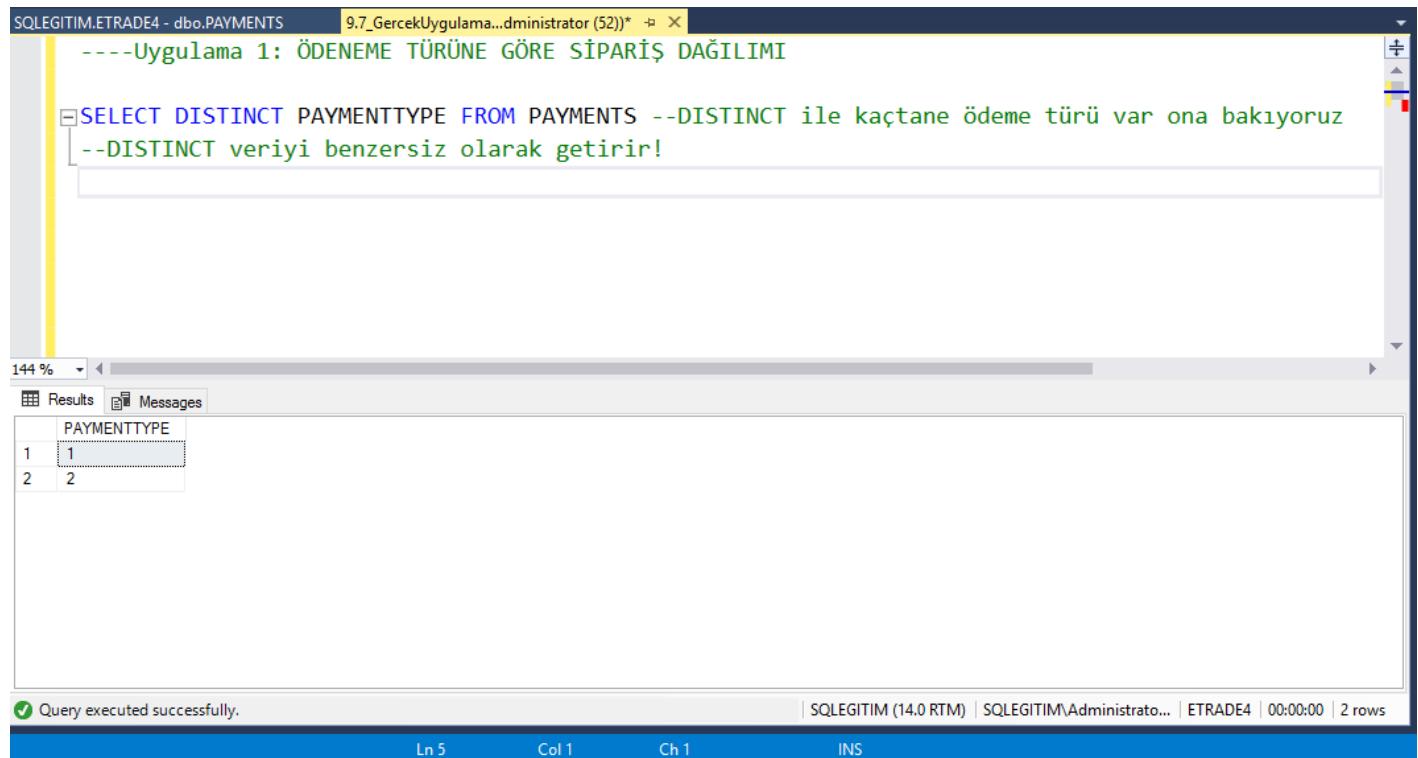
✓ Query executed successfully.
```

9.7_GercekUygulamalarOrnek6

Uygulama 1

ÖDEME türüne göre sipariş dağılımı için ilk başta kaçtane ödeme türü var ona bakalım.

(Ödeme türü bilgisi PAYMENTS tablosundaki PAYMENTTYPE stünunda)



```
-----Uygulama 1: ÖDENEME TÜRÜNE GÖRE SİPARİŞ DAĞILIMI

SELECT DISTINCT PAYMENTTYPE FROM PAYMENTS --DISTINCT ile kaçtane ödeme türü var ona bakiyoruz
--DISTINCT veriyi benzersiz olarak getirir!
```

PAYMENTTYPE
1
2

Query executed successfully. | SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 2 rows

Uygulama 2

Ödeme türüne göre toplam tutar:

SQLEGITIM.ETRADE4 - dbo.PAYMENTS 9.7_GercekUygulama..dministrator (52)*

```
--Uygulama 2:

SELECT
    PAYMENTTYPE AS ODEMETURU,
    SUM(PAYMENTTOTAL) AS TOPLAMTUTAR

FROM PAYMENTS --DISTINCT ile kaçtane ödeme türü var ona bakıyoruz
--DISTINCT veriyi benzersiz olarak getirir!
GROUP BY PAYMENTTYPE
```

144 %

Results Messages

	ODEMETURU	TOPLAMTUTAR
1	1	2628495.4500
2	2	2565533.7800

Query executed successfully. | SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 2 rows

Ancak biz bunu 1 ise kredi kartı 2 ise banka havalesi ile değiştirelim.

Uygulama 3

SQLEGITIM.ETRADE4 - dbo.PAYMENTS 9.7_GercekUygulama..dministrator (52)*

```
--Uygulama 3:
--Ödeme türü 1 ise kredi kartı 2 ise banka havalesi olarak gösterelim
SELECT
    DATEPART(YEAR,DATE_) AS YIL,
    DATEPART(MONTH,DATE_) AS AY,
    CASE
        WHEN PAYMENTTYPE=1 THEN 'KREDİ KARTI'
        WHEN PAYMENTTYPE=2 THEN 'BANKA HVALESİ'
    END AS ODEMETURU_ACIKLAMA, --geçici stünun ismini verdik
    SUM(PAYMENTTOTAL) AS TOPLAMTUTAR

FROM PAYMENTS --DISTINCT ile kaçtane ödeme türü var ona bakıyoruz
--DISTINCT veriyi benzersiz olarak getirir!
GROUP BY DATEPART(YEAR,DATE_), PAYMENTTYPE, DATEPART(MONTH,DATE_)
ORDER BY DATEPART(YEAR,DATE_), DATEPART(MONTH,DATE_)
```

108 %

Results Messages

	YIL	AY	ODEMETURU_ACIKLAMA	TOPLAMTUTAR
1	2019	1	KREDİ KARTI	171617.4900
2	2019	1	BANKA HVALESİ	178093.8300
3	2019	2	KREDİ KARTI	175212.8500
4	2019	2	BANKA HVALESİ	180059.2200
5	2019	3	BANKA HVALESİ	160597.1900
6	2019	3	KREDİ KARTI	195826.7500
7	2019	4	BANKA HVALESİ	163764.4500
8	2019	4	KREDİ KARTI	164764.0400
9	2019	5	BANKA HVALESİ	186611.7700
10	2019	5	KREDİ KARTI	195261.4600
11	2019	6	BANKA HVALESİ	178048.3800
12	2019	6	KREDİ KARTI	159962.6100

Query executed successfully. | SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 30 rows

9.8_GercekUygulamalarOrnek7

Uygulama 1

Ortalama teslimat süresi için ilk başta iki zaman arasındaki Ay,yıl gün, saat, dakika farkını hesaplamamız gereklidir. Bunun içinde DATEDIFF komutunu kullanacağımız:

The screenshot shows a SQL query window titled '9.8_GercekUygulamalarOrnek7' with the title bar indicating 'dministrator (57)*'. The status bar at the bottom right shows 'SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 1 rows'. The main area contains the following SQL code and its results:

```
--DATAIFF İLE Ay,yıl gün, saat, dakika farkı dahi alabiliriz !
SELECT DATEDIFF(HOUR,'2019-02-05 23:56:58.000','2019-02-06 10:54:58.000') -- iki zaman arasındaki SAAT farkı
```

Results	Messages
(No column name)	
1	11

A green checkmark icon in the status bar indicates the query was executed successfully.

The screenshot shows another SQL query window with the same title and connection details. The main area contains the following SQL code and its results:

```
SELECT DATEDIFF(DAY,'2019-02-05 23:56:58.000','2019-02-06 10:54:58.000') -- GÜN FARKI
```

Results	Messages
(No column name)	
1	1

A green checkmark icon in the status bar indicates the query was executed successfully.

SELECT DATEDIFF(HOUR,'2019-02-05 23:56:58.000','2019-02-06 10:54:58.000') --HOUR yerine MINUTE,DAY gibi zamanların farkını bulabiliyoruz.

Uygulama 2

9.8_GercekUygulama...dministrator (57) ━ ×

```
--Uygulama 2:
SELECT
    O.ID AS SIPARISID,
    O.DATE_ AS SIRAPISTARIHI,
    I.DATE_ AS FATURATARIHI,
    --Sipariş tarihinden fatura tarihini çıkarırsak TESLIMATSURESI_SAAT elde etmiş oluruz.
    DATEDIFF(HOUR,O.DATE_,I.DATE_) AS TESLIMATSURESI_SAAT -- iki zaman arasındaki SAAT farkı
FROM ORDERS O
    INNER JOIN USERS U ON U.ID=O.USERID
    INNER JOIN ADDRESS A ON A.ID=O.ADDRESSID
    INNER JOIN CITIES CT ON CT.ID=A.CITYID
    INNER JOIN TOWNS T ON T.ID=A.TOWNID
    INNER JOIN DISTRICTS D ON D.ID=A.DISTRICTID
    INNER JOIN PAYMENTS P ON P.ORDERID=O.ID
    INNER JOIN INVOICES I ON I.ORDERID=O.ID
```

119 %

Results Messages

	SIPARISID	SIRAPISTARIHI	FATURATARIHI	TESLIMATSURESI_SAAT
1	1	2019-02-05 23:56:58.000	2019-02-06 10:54:58.000	11
2	2	2019-04-26 14:05:37.000	2019-04-27 08:47:37.000	18
3	3	2019-01-19 10:30:30.000	2019-01-20 09:11:30.000	23
4	4	2019-12-23 20:46:51.000	2019-12-24 09:02:51.000	13
5	5	2019-08-26 20:02:52.000	2019-08-27 08:59:52.000	12
6	6	2019-08-14 00:01:28.000	2019-08-14 15:05:28.000	15
7	7	2020-03-19 02:29:31.000	2020-03-20 09:55:31.000	31

Query executed successfully. | SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 30.394 rows

Uygulama 3

9.8_GercekUygulama...dministrator (57) ━ ×

```
Decrease Indent
----Uygulama 3:
--Şimdide ortalama teslimat süresi için AVG komutunu kullanarak ortalama teslimat sürelerini görelim
SELECT
    --Sipariş tarihinden fatura tarihini çıkarırsak TESLIMATSURESI_SAAT elde etmiş oluruz.
    AVG(DATEDIFF(HOUR,O.DATE_,I.DATE_)) AS ORTLAMA_TESLIMATSURESI_SAAT -- iki zaman arasındaki SAAT farkı
FROM ORDERS O
    INNER JOIN USERS U ON U.ID=O.USERID
    INNER JOIN ADDRESS A ON A.ID=O.ADDRESSID
    INNER JOIN CITIES CT ON CT.ID=A.CITYID
    INNER JOIN TOWNS T ON T.ID=A.TOWNID
    INNER JOIN DISTRICTS D ON D.ID=A.DISTRICTID
    INNER JOIN PAYMENTS P ON P.ORDERID=O.ID
    INNER JOIN INVOICES I ON I.ORDERID=O.ID
```

119 %

Results Messages

	ORTLAMA_TESLIMATSURESI_SAAT
1	22

Query executed successfully. | SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 1 rows

Uygulama 4

```

9.8_GercekUygulama...dministrator (55)* -p X

----Uygulama 4:
--ENKISA_TESLIMATSURESİ_SAAT ve ENUZUN_TESLIMATSURESİ_SAAT i listeledik
SELECT
MIN(DATEDIFF(HOUR,O.DATE_,I.DATE_)) AS ENKISA_TESLIMATSURESİ_SAAT, -- MIN komutu ile minimum yani en kısa teslimat süresi listeledik
MAX(DATEDIFF(HOUR,O.DATE_,I.DATE_)) AS ENUZUN_TESLIMATSURESİ_SAAT, -- MAX komutu ile minimum yani en uzun teslimat süresi listeledik
AVG(DATEDIFF(HOUR,O.DATE_,I.DATE_)) AS ORTLAMA_TESLIMATSURESİ_SAAT -- iki zaman arasındaki SAAT farkı

FROM ORDERS O
INNER JOIN USERS U ON U.ID=O.USERID
INNER JOIN ADDRESS A ON A.ID=O.ADDRESSID
INNER JOIN CITIES CT ON CT.ID=A.CITYID
INNER JOIN TOWNS T ON T.ID=A.TOWNID
INNER JOIN DISTRICTS D ON D.ID=A.DISTRICTID
INNER JOIN PAYMENTS P ON P.ORDERID=O.ID
INNER JOIN INVOICES I ON I.ORDERID=O.ID

98 % < > Results Messages
ENKISA_TESLIMATSURESİ_SAAT ENUZUN_TESLIMATSURESİ_SAAT ORTLAMA_TESLIMATSURESİ_SAAT
1 10 34 22

```

Query executed successfully.

| SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 1 rows

Uygulama 5

```

9.8_GercekUygulama...dministrator (55)* -p X

----Uygulama 5:
--Bu uygulamada ise teslimat süresi çok uzun olan müşterilerin gönlünü almak kampanya yapmak üzere onları tespit edelim

SELECT
U.ID,U.NAMESURNAME,
AVG(DATEDIFF(HOUR,O.DATE_,I.DATE_)) AS ORTLAMA_TESLIMATSURESİ_SAAT

FROM ORDERS O
INNER JOIN USERS U ON U.ID=O.USERID
INNER JOIN ADDRESS A ON A.ID=O.ADDRESSID
INNER JOIN CITIES CT ON CT.ID=A.CITYID
INNER JOIN TOWNS T ON T.ID=A.TOWNID
INNER JOIN DISTRICTS D ON D.ID=A.DISTRICTID
INNER JOIN PAYMENTS P ON P.ORDERID=O.ID
INNER JOIN INVOICES I ON I.ORDERID=O.ID

GROUP BY U.ID,U.NAMESURNAME
--HAVING AVG(DATEDIFF(HOUR,O.DATE_,I.DATE_)) > 96 -- teslimat süresi 96 saat ten yüksek olanları listeledik
ORDER BY 3 DESC

98 % < > Results Messages
ID NAMESURNAME ORTLAMA_TESLIMATSURESİ_SAAT
1 3209 Rumeysa ÖZAMARATLI 34
2 8350 Ayten ŞENSAN 34
3 8765 Mücahit OZTÜRKÇİ 34
4 9275 Pınar ELBİL 34
5 4698 Bekir BOZTUG 34
6 6679 Muhammet Ali LOKANTASI 34
7 3456 Demir HASKILIÇ 34
8 144 Deniz BENDER 34
9 3765 Bekir DOĞANTEKİN 34
10 9069 Rumeysa HISİM 34

```

Query executed successfully.

| SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 9.274 rows

Uygulama 6

9.8_GerçekUygulamalarOrnek7.sql - localhost:ETRADE4 (SQLLEGITIM\Administrator (54)) - Microsoft SQL Server Management Studio (Administrator)

```
--Uygulama 6:
SELECT
    U.ID, U.NAMESURNAME,
    AVG(DATEDIFF(HOUR,O.DATE_,I.DATE_)) AS ORTLAMA_TESLIMATSURESISAAAT,
    MIN(DATEDIFF(HOUR,O.DATE_,I.DATE_)) AS ENKISA_TESLIMAT_SURESI,
    MAX(DATEDIFF(HOUR,O.DATE_,I.DATE_)) AS ENUZUN_TESLIMAT_SURESI,
    SUM(O.TOTALPRICE) AS TOPLAM_SIPARIS_TUTARI, --fazla alış-veriş yapanlara kampanyada öncelik yapmak için
    COUNT(O.ID) AS SIPARISSAYISI
FROM ORDERS O
    INNER JOIN USERS U ON U.ID=O.USERID
    INNER JOIN ADDRESS A ON A.ID=O.ADDRESSID
    INNER JOIN CITIES CT ON CT.ID=A.CITYID
    INNER JOIN TOWNS T ON T.ID=A.TOWNID
    INNER JOIN DISTRICTS D ON D.ID=A.DISTRICTID
    INNER JOIN PAYMENTS P ON P.ORDERID=O.ID
    INNER JOIN INVOICES I ON I.ORDERID=O.ID
GROUP BY U.ID,U.NAMESURNAME
--HAVING AVG(DATEDIFF(HOUR,O.DATE_,I.DATE_)) > 96 -- teslimat süresi 96 saat ten yüksek olanları listeledik
ORDER BY 6 DESC
```

Results

ID	NAMESURNAME	ORTLAMA_TESLIMATSURESISAAAT	ENKISA_TESLIMAT_SURESI	ENUZUN_TESLIMAT_SURESI	TOPLAM_SIPARIS_TUTARI	SIPARISSAYISI
1	Ekin AKBUL	21	15	25	16919.4682	11
2	Mucahit GÖKKÜL	28	24	32	15708.1168	4
3	Berfin YAGCI	24	15	31	15482.8466	8
4	Tank ATEŞLİER	24	13	31	15370.4830	4
5	Duran BAŞGERMEZ	20	10	26	14256.5963	8
6	Omer Asaf GÜNDÜC	26	18	34	13686.7072	4

Query executed successfully.

Uygulama 7

VM_New - VMware Workstation 15 Player (Non-commercial use only)

File Edit View Query Project Tools Window Help

localhost:ETRADE4 (SQLLEGITIM\Administrator (54)) - Microsoft SQL Server Management Studio (Administrator)

```
--Uygulama 7:
SELECT
    DATEPART(YEAR,O.DATE_) AS YIL, --Tarihteki sadece yılı listeliyoruz
    DATEPART(MONTH,O.DATE_) AS AY, --Tarihteki sadece ayı listeliyoruz
    AVG(DATEDIFF(HOUR,O.DATE_,I.DATE_)) AS ORTLAMA_TESLIMATSURESISAAAT,
    MIN(DATEDIFF(HOUR,O.DATE_,I.DATE_)) AS ENKISA_TESLIMAT_SURESI,
    MAX(DATEDIFF(HOUR,O.DATE_,I.DATE_)) AS ENUZUN_TESLIMAT_SURESI,
    SUM(O.TOTALPRICE) AS TOPLAM_SIPARIS_TUTARI, --fazla alış-veriş yapanlara kampanyada öncelik yapmak için
    COUNT(O.ID) AS SIPARISSAYISI
FROM ORDERS O
    INNER JOIN USERS U ON U.ID=O.USERID
    INNER JOIN ADDRESS A ON A.ID=O.ADDRESSID
    INNER JOIN CITIES CT ON CT.ID=A.CITYID
    INNER JOIN TOWNS T ON T.ID=A.TOWNID
    INNER JOIN DISTRICTS D ON D.ID=A.DISTRICTID
    INNER JOIN PAYMENTS P ON P.ORDERID=O.ID
    INNER JOIN INVOICES I ON I.ORDERID=O.ID
GROUP BY DATEPART(YEAR,O.DATE_), DATEPART(MONTH,O.DATE_)
ORDER BY 1,2 DESC
```

Results

YIL	AY	ORTLAMA_TESLIMATSURESISAAAT	ENKISA_TESLIMAT_SURESI	ENUZUN_TESLIMAT_SURESI	TOPLAM_SIPARIS_TUTARI	SIPARISSAYISI
1	2019	12	10	34	887864.6859	2055
2	2019	11	22	10	869167.9962	2014
3	2019	10	21	10	931300.8409	2141
4	2019	9	21	10	853261.5710	1966
5	2019	8	22	10	924736.9311	2123
6	2019	7	22	10	857835.0134	2085

Query executed successfully.

Uygulama 8

```

9.8_GercekUygulamalarOrnek7.sql - localhost.ETRADE4 [SQLEGITIM\Administrator (54)*] - Microsoft SQL Server Management Studio (Administrator)
File Edit View Query Project Tools Window Help
Object Explorer
Connect > ETRADE4 Execute New Query File Database Tools Window Help
SQLLEGITIM.ETRADE4 - dbo.ORDERS 9.8_GercekUygulam...ministrator (54)*
--Yugulama 8:
--Sehirlere göre gruplayalim
SELECT
CT.CITY,
AVG(DATEDIFF(HOUR,O.DATE_,I.DATE_)) AS ORTLAMA_TESLIMATSURESI_SAAT,
MIN(DATEDIFF(HOUR,O.DATE_,I.DATE_)) AS ENKISA_TESLIMAT_SURESI,
MAX(DATEDIFF(HOUR,O.DATE_,I.DATE_)) AS ENUZUN_TESLIMAT_SURESİ,
SUM(O.TOTALPRICE) AS TOPLAM_SIPARIS_TUTARI, --fazla alış-veriş yapanlara kampanyada öncelik yapmak için
COUNT(O.ID) AS SIPARISSAYISI
FROM ORDERS O
INNER JOIN USERS U ON O.USERID = U.ID
INNER JOIN ADDRESS A ON A.ID = O.ADDRESSID
INNER JOIN CITIES CT ON CT.ID = A.CITYID
INNER JOIN TOWNS T ON T.ID = A.TOWNID
INNER JOIN DISTRICTS D ON D.ID = A.DISTRICTID
INNER JOIN PAYMENTS P ON P.ORDERID = O.ID
INNER JOIN INVOICES I ON I.ORDERID = O.ID
GROUP BY CT.CITY
ORDER BY 2 DESC

```

Results

CITY	ORTLAMA_TESLIMATSURESI_SAAT	ENKISA_TESLIMAT_SURESİ	ENUZUN_TESLIMAT_SURESİ	TOPLAM_SIPARIS_TUTARI	SIPARISSAYISI
1 BILECIK	24	10	34	38361.1231	104
2 ARDAHAN	24	12	34	10603.8959	20
3 BARTIN	23	11	34	29129.8057	65
4 SINOP	23	10	34	44002.4450	114
5 SIVAS	22	10	34	174028.4961	399
6 ŞANLIURFA	22	10	34	212144.9369	493

Query executed successfully.

10.1_Subquery

Veritabanı sistemleri esasen veri okuma-yazma yapar. Yani hdd den rama alıp kullanıcıya gösterir.

Ne kadar okuma, yazma vs yapmış bunun istatistiğine bakabiliriz.

SET STATISTICS IO ON ile yapabiliriz. İlk başka STATISTICS IO yu ON yapalım. Daha sonra hangi sorgunun istatistiğini görmek istiyorsak onu Execute(F5) leyelim

```

10.1_Subquery.sql - ...Administrator (55)*
SET STATISTICS IO ON

(28066 rows affected)
Table 'ORDERDETAILS'. Scan count 20, logical reads 3748, physical reads 0
Table 'ITEMS'. Scan count 5, logical reads 1294, physical reads 0,
Table 'Workfile'. Scan count 0, logical reads 0, physical reads 0,
Table 'Worktable'. Scan count 0, logical reads 0, physical reads 0,

```

Results

Query executed s... | SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 28.066 rows

Ancak JOIN, Subquery'ye göre daha performanslıdır !

10.2_Subquery2

Subquery ile ilgili başka bir kullanım örneği yapalım. Ürünün hangi ay en çok satıldığı bilgisini getirelim !

The screenshot shows a SQL Server Management Studio window with two tabs: '10.2_Subquery2.sql...' and '10.3_Subquery3.sql...'. The '10.3_Subquery3.sql...' tab is active, displaying a query to find products sold most in a month. The results grid shows 28 rows of product details and their sales statistics.

```
--Subquery(İç içe Select)
--Ürünlerin en çok hangi ay satıldığı bilgisini listeleyelim.

SELECT ITM.ITEMCODE AS URUNKODU,
ITM.ITEMNAME AS URUNADI,
(SELECT MIN(UNITPRICE) FROM ORDERDETAILS WHERE ITEMID=ITM.ID) AS ENDUSUKFIYAT,
(SELECT MAX(UNITPRICE) FROM ORDERDETAILS WHERE ITEMID=ITM.ID) AS ENYUKSEKFIYAT,
(SELECT AVG(UNITPRICE) FROM ORDERDETAILS WHERE ITEMID=ITM.ID) AS ORTALAMAFIYAT,
(SELECT SUM(AMOUNT) FROM ORDERDETAILS WHERE ITEMID=ITM.ID) AS TOPLAMFIYAT,
(
SELECT TOP 1 DATEPART(MONTH,O.DATE_) AS AY FROM ORDERDETAILS OD
INNER JOIN ORDERS O ON OD.ORDERID=O.ID
WHERE OD.ITEMID=ITM.ID
GROUP BY DATEPART(MONTH,O.DATE_)
ORDER BY SUM(AMOUNT) DESC
) AS ENCOKSATILANAY

FROM ITEMS ITM
ORDER BY ITM.ITEMNAME
```

URUNKODU	URUNADI	ENDUSUKFIYAT	ENYUKSEKFIYAT	ORTALAMAFIYAT	TOPLAMFIYAT	ENCOKSATILANAY
1	"ACE ELDE YIKAMA BAHAR 700 GR *22"	1.4190	1.5183	1.481050	24	4
2	"BINGO DINAMIC B.MAK.TAB.8 IN 1 30 LU *5"	12.9520	13.8327	13.335400	28	1
3	"CALGONIT 2 KG EKO LIMON *8"	4.1047	4.8952	4.632188	51	8
4	"CALGONIT FINISH COMBO PACK *9"	10.4449	11.1367	10.790942	48	2
5	"CANTA BOY KURAN I KERIM	9.2714	10.7886	9.980440	31	3
6	"CIF DW 3 KG LIMON BUL DET.LIM.*4"	8.7479	9.4258	9.085000	41	9

10.3_Subquery3

Bunların mutlaka sorgu kodlarını ve sonuçlarının ss i ni al

10.4_Bolumsonu_SubQuery

- **SUBQUERY kavramı nedir?**
- **SUBQUERY nasıl kullanılır?**
- **Join'e göre avantaj ve dezavantajları nelerdir?**
- **Nerede Join, Nerede SubQuery kullanılması gereklidir?**

11.1_stringislemeleriGiris

ASCII	RIGHT
CHAR	RTRIM
CHARINDEX	SOUNDEX
CONCAT	SPACE
CONCAT_WS	STR
DIFFERENCE	STRING_AGG
FORMAT	STRING_ESCAPE
LEFT	STRING_SPLIT
LEN	STUFF
LOWER	SUBSTRING
LTRIM	TRANSLATE
NCHAR	TRIM
PATINDEX	UNICODE
QUOTENAME	UPPER
REPLACE	
REPLICATE	
REVERSE	

11.2_stringislemeleri_Ascii_Char

```
11.1_stringislemeleri...\Administrator (55)* - X
--ASCII
SELECT ASCII ('A') --A'nın ASCII karşılığı
118 %
Results Messages
(No column name)
1 65
Query executed successfully. | SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | master | 00:00:00 | 1 rows
```

Char da Ascii nin ters fonksiyonu: Yani A'nın Ascii deki karşılığı 65. 65 in Ascii Tablosundaki kişi ise "A"

The screenshot shows a SQL Server Management Studio (SSMS) window. The title bar reads "11.1_stringislemeleri...\\Administrator (55)". The query pane contains the following code:

```
--CHAR  
SELECT CHAR (65) --A nın karşılığı
```

The results pane shows a single row with the value "A". The status bar at the bottom indicates "Query executed successfully." and "1 rows".

Bu fonksiyonları iç içe kullanabiliriz :

--Fonksiyonların iç içe kullanımı

```
SELECT ASCII(CHAR(49))
SELECT CHAR(ASCII('Ö'))
```

	(No column name)
1	Ö

Query executed successfully.

SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | master | 00:00:00 | 1 rows

11.3_stringislemeleriGiris_Substring

11.3_stringislemeleri...\\Administrator (54)* ×

```
--SUBSTRING  
--Bir string içerisindeki belirli bir harften belirli bir harfe kadar almayı sağlamaktadır.  
  
SELECT SUBSTRING('ÖMER FARUK ÇOLAKOĞLU',1,4) --1 den başla 4 tane harf al
```

118 %

Results Messages

	(No column name)
1	ÖMER

Query executed successfully. | SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | master | 00:00:00 | 1 rows

11.4_stringislemeleriGiris_Substring2

11.3_stringislemeleri...\Administrator (54)*

```
--SUBSTRING2
--İsmi O ile başlayanları listeleyelim:
SELECT * FROM USERS WHERE USERNAME_ LIKE 'O%'
```

118 %

	ID	USERNAME_	PASSWORD_	NAMESURNAME	EMAIL	GENDER	BIRTHDATE	CREAT
1	8	O_KIRIT	sugarandspice39	Onur KIRIT	O_KIRIT@sqlegitimbtk.com	E	1972-02-25	2018-0
2	61	O_FARUK	@80r+lve1y	Ömer Faruk KOCANOĞLU	O_FARUK@sqlegitimbtk.com	E	1952-08-11	2018-0
3	62	O_KVCVKIRAVUL	blondie1436	Özcan KÜÇÜKIRAVUL	O_KVCVKIRAVUL@sqlegitimbtk.com	E	1971-10-08	2018-0
4	68	O_EYYVBOGLU	gregorychan	Onur EYYÜBOĞLU	O_EYYVBOGLU@sqlegitimbtk.com	E	1945-09-29	2018-1
5	85	O_DERILIOGLU	decafhd	Özkan DERİLIOĞLU	O_DERILIOGLU@sqlegitimbtk.com	E	1996-06-02	2018-1
6	117	O_ISIS	iluvfarout1999	Orhan ISIS	O_ISIS@sqlegitimbtk.com	E	1984-04-20	2018-1
7	130	O_ELEKTIRIK	ex-solent	Özcan ELEKTİRİK	O_ELEKTIRIK@sqlegitimbtk.com	E	1960-02-26	2018-1
8	169	O_CAYIR	xalwepad	Okan CAYIR	O_CAYIR@sqlegitimbtk.com	E	1986-04-19	2018-0
9	182	O_KOKAN	blessedwith1	Oğuzhan KÖKAN	O_KOKAN@sqlegitimbtk.com	E	1941-07-09	2018-0
10	191	O_HALIS	janet9525	Ömer Halis DÖŞEME	O_HALIS@sqlegitimbtk.com	E	1977-09-06	2018-0
11	215	O_CVRE	simpsonakahottie	Oğuz CÜRE	O_CVRE@sqlegitimbtk.com	E	1973-02-11	2018-0
12	264	O_HAFIZOGLU	what_is_a_my...	Özgür HAFIZOĞLU	O_HAFIZOGLU@sqlegitimbtk.com	E	1949-11-19	2018-0
13	381	O_ELVAN	supplic@t3	Özcan ELVAN	O_ELVAN@sqlegitimbtk.com	E	1963-11-07	2018-0
14	392	O_KOSKA	billuh	Ömer KOSKA	O_KOSKA@sqlegitimbtk.com	E	1956-04-26	2018-0

Query executed successfully. | SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 296 rows

11.3_stringislemeleri...\Administrator (54)* X

Toggle pin status

--İsmi O ile başlayanları SUBSTRING ile listeleyelim:
 SELECT * FROM USERS WHERE SUBSTRING(USERNAME_,1,1)='O'

118 %

	ID	USERNAME_	PASSWORD_	NAMESURNAME	EMAIL	GENDER	BIRTHDATE	C
1	8	O_KIRIT	sugarandspice39	Onur KIRIT	O_KIRIT@sqlegitimbtk.com	E	1972-02-25	...
2	61	O_FARUK	@80r+lve1y	Ömer Faruk KOCAANOĞLU	O_FARUK@sqlegitimbtk.com	E	1952-08-11	...
3	62	O_KVCVKIRAVUL	blondie1436	Özcan KÜÇÜKIRAVUL	O_KVCVKIRAVUL@sqlegitimbtk.com	E	1971-10-08	...
4	68	O_EYYVBOGLU	gregorychan	Onur EYYÜBOĞLU	O_EYYVBOGLU@sqlegitimbtk.com	E	1945-09-29	...
5	85	O_DERILIOGLU	decafhd	Özkan DERİLIOĞLU	O_DERILIOGLU@sqlegitimbtk.com	E	1996-06-02	...
6	117	O_ISIS	iluvfarout1999	Orhan ISIŞ	O_ISIS@sqlegitimbtk.com	E	1984-04-20	...
7	130	O_ELEKTIRIK	ex-solent	Özcan ELEKTİRİK	O_ELEKTIRIK@sqlegitimbtk.com	E	1960-02-26	...
8	169	O_CAYIR	xalwepad	Okan CAYIR	O_CAYIR@sqlegitimbtk.com	E	1986-04-19	...
9	182	O_KOKAN	blessedwith1	Oğuzhan KÖKAN	O_KOKAN@sqlegitimbtk.com	E	1941-07-09	...
10	191	O_HALIS	janet9525	Ömer Halis DÖŞEME	O_HALIS@sqlegitimbtk.com	E	1977-09-06	...
11	215	O_CVRE	simpsonakahottie	Oğuz CÜRE	O_CVRE@sqlegitimbtk.com	E	1973-02-11	...
12	264	O_HAFIZOGLU	what_is_a_myspace_url	Özgür HAFIZOĞLU	O_HAFIZOGLU@sqlegitimbtk.com	E	1949-11-19	...
13	381	O_ELVAN	supplic@t3	Özcan ELVAN	O_ELVAN@sqlegitimbtk.com	E	1963-11-07	...
14	392	O_KOSKA	billuh	Ömer KOSKA	O_KOSKA@sqlegitimbtk.com	E	1956-04-26	...

Query executed successfully. | SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 296 rows

```
--A_KALKMAZO'nın Uzunluğuna bakalım  
SELECT LEN('A_KALKMAZO') --A_KALKMAZO stringinin uzunluğu
```

118 %

Results Messages

	(No column name)
1	10

Query executed successfully.

| SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 1 rows

11.4_stringislemeleri...\\Administrator (57)*

```
--Son harfi o olanlar
--Son harfinin 0 olduğunu anlamak için stringin uzunluğunu bilmek lazım.
--Bu sebeple LEN fonksiyonunu kullanırsak son harfi 0 olanları sıralayabiliriz
SELECT * FROM USERS WHERE SUBSTRING(USERNAME_,LEN(USERNAME_),1)='0'
```

118 %

	ID	USERNAME_	PASSWORD_	NAMESURNAME	EMAIL	GENDER	BIRTHDATE	CREATED
1	15	M_KALO	chocolatefireguardmusic	Mina KALO	M_KALO@sqlegitimbtk.com	K	1987-04-01	2018-08-11 10:10:10
2	438	A_KALKMAZO	ultraapte8	Aysel KALKMAZO	A_KALKMAZO@sqlegitimbtk.com	K	1967-06-19	2018-10-11 10:10:10
3	788	A_DADAGLIO	morgenstundhatgoldimmund	Asiya DADAĞLIO	A_DADAGLIO@sqlegitimbtk.com	K	1960-11-23	2018-09-11 10:10:10
4	1821	H_SISKO	onedayiamgoingtogrowwings	Hamdi ŞİŞKO	H_SISKO@sqlegitimbtk.com	E	1953-08-13	2018-08-11 10:10:10
5	1986	F_IZMAYLO	29may1471	Filiz İZMAYLO	F_IZMAYLO@sqlegitimbtk.com	K	1979-05-26	2018-01-11 10:10:10
6	2389	A_BERBERO	55432s971	Aleyna BERBERO	A_BERBERO@sqlegitimbtk.com	K	1971-05-17	2018-02-11 10:10:10
7	2782	A_IZMAYLO	nhess61	Ali İZMAYLO	A_IZMAYLO@sqlegitimbtk.com	E	1992-11-01	2018-03-11 10:10:10
8	2863	S_GOLO	kogusfamily	Seher GÖLO	S_GOLO@sqlegitimbtk.com	K	1951-01-25	2018-06-11 10:10:10
9	3013	K_MOTO	pppd23075	Kazim MOTO	K_MOTO@sqlegitimbtk.com	E	1949-01-22	2018-09-11 10:10:10
10	3175	S_CANKO	25-04-1289	Sabriye CANKO	S_CANKO@sqlegitimbtk.com	K	1965-09-17	2018-02-11 10:10:10
11	2252	M_SEZGEKO	asusztamashina	Mehmet SEZGEKO	M_SEZGEKO@sqlegitimbtk.com	K	1950-10-10	2017-11-11 10:10:10

Query executed successfully. | SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 30 rows

11.5_stringislemeleriGiris_Charindex

The screenshot shows the Microsoft SQL Server Management Studio interface. The title bar reads "SQLServer_New - VMware Workstation 15 Player (Non-commercial use only)". The main window displays a query in the Query Editor:

```
--CHARINDEX  
--(NE ARAMAK İSTİYORSUN, NEREDE ARAMAK İSTİYORSUN, NERDEN BAŞLAYARAK ARA(hangi indexten)  
SELECT CHARINDEX('F','ÖMER FARUK COLAKOĞLU',1)  
--sonuç olarak 6 tane F harfi olduğunu verir.|
```

The results pane shows a single row of data:

(No column name)
1
6

At the bottom of the results pane, a message indicates: "Query executed successfully." The status bar at the bottom right shows: "SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 1 rows".

```
--Uygulama2
-- 7. harften sonra arama yaptıralım
SELECT CHARINDEX('F','ÖMER FARUK ÇOLAKOĞLU',7) -- 7. harften sonra aramaya başla
--Yani 'ARUK ÇOLAKOĞLU' dan sonra arama yapalım demektir.F harfi yok.
```

118 %

Results Messages

	(No column name)
1	0

Query executed successfully. | SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 1 rows

11.6_stringislemeleriGiris_Concat_ConcatWs

İki ya da daha fazla stringi birleştirilmesini sağlar.

```
11.6_stringislemeleri...\\Administrator (56)* ×
--CONCAT
--3 kelimeyi birleştirelim:
SELECT 'OMER'+'FARUK'+'COLAKOGLU'

118 %
```

Results Messages

	(No column name)
1	OMERFARUKCOLAKOGLU

Query executed successfully. | SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | master | 00:00:00 | 1 rows

11.6_stringislemeleri...\\Administrator (56))* X

```
--Aynı işlemi birde CONCAT ile deneyelim:  
--Aralarına boşluk bırakalım  
SELECT CONCAT('OMER', ' ', 'FARUK', ' ', 'COLAKOGLU')
```

118 %

Results Messages

	(No column name)
1	OMER FARUK COLAKOGLU

Query executed successfully. | SQLLEGITIM (14.0 RTM) | SQLLEGITIM\Administrato... | master | 00:00:00 | 1 rows

11.6_stringislemeleri..\Administrator (56)* X

```
--CONCAT_WS  
--Aralarına konulacak boşluk ya da karakteri belirtmemiz lazım  
SELECT CONCAT_WS(' ', 'OMER', 'FARUK', 'COLAKOGLU')
```

118 %

Results Messages

	(No column name)
1	OMER FARUK COLAKOGLU

Query executed successfully. | SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | master | 00:00:00 | 1 rows

11.6_stringislemeleri...\Administrator (56)* ✎ X

```
--Arasına nokta koymalı:
SELECT CONCAT_WS('.','OMER','FARUK','COLAKOGLU')
```

118 %

Results Messages

(No column name)
1 OMER.FARUK.COLAKOGLU

Query executed successfully.

| SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | master | 00:00:00 | 1 rows

11.6_stringislemeleri...\\Administrator (56) X

```

--CONCAT, CONCAT_WS i tabloda kullanalım:
--Mesela USERNAME,PASSWORD ve NAMESURNAME'i birlikte tek sütunda getirelim
SELECT
    USERNAME_+' '+PASSWORD_+' '+NAMESURNAME, --+ ile birleştirdik
    CONCAT(USERNAME_, ' ',PASSWORD_, ' ',NAMESURNAME), --CONCAT ile birleştirdik
    CONCAT_WS(' ',USERNAME_,PASSWORD_,NAMESURNAME), --CONCAT_WS ile birleştirdik
* FROM USERS

```

118 %

	(No column name)	(No column name)	(No column name)
1	N_OZSIMITCI doropipa Nazlıcan ÖZSİMITÇİ	N_OZSIMITCI doropipa Nazlıcan ÖZSİMITÇİ	N_OZSIMITCI doropipa Nazlıcan ÖZSİMITÇİ
2	E_SELIM xpais Emirhan SELİM	E_SELIM xpais Emirhan SELİM	E_SELIM xpais Emirhan SELİM
3	S_VLGEN 4042521 Soner ÜLGEN	S_VLGEN 4042521 Soner ÜLGEN	S_VLGEN 4042521 Soner ÜLGEN
4	C_BORKLV audraia Çetin BÖRKLU	C_BORKLV audraia Çetin BÖRKLU	C_BORKLV audraia Çetin BÖRKLU
5	E_IBUKVRTVNCV translatlantic Ezgi İBUKÜRTÜNCÜ	E_IBUKVRTVNCV translatlantic Ezgi İBUKÜRTÜNCÜ	E_IBUKVRTVNCV translatlantic Ezgi İBUKÜRTÜNCÜ
6	H_VREGİL happy2shoes Hazal ÜREGİL	H_VREGİL happy2shoes Hazal ÜREGİL	H_VREGİL happy2shoes Hazal ÜREGİL
7	A_SUYUR im_nthelordsammy Aykut SUYUR	A_SUYUR im_nthelordsammy Aykut SUYUR	A_SUYUR im_nthelordsammy Aykut SUYUR
8	O_KIRIT sugarandspice39 Onur KIRIT	O_KIRIT sugarandspice39 Onur KIRIT	O_KIRIT sugarandspice39 Onur KIRIT
9	S_TVKEZİM xtremelyhot Songül TÜKEZİM	S_TVKEZİM xtremelyhot Songül TÜKEZİM	S_TVKEZİM xtremelyhot Songül TÜKEZİM
10	R_PIRINCAI swissmuite Berkav PİRİNCAI	R_PIRINCAI swissmuite Berkav PİRİNCAI	R_PIRINCAI swissmuite Berkav PİRİNCAI

Query executed successfully. | SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:01 | 9.678 rows

11.7_stringislemeleriGiris_Format

```
11.7_stringislemeleri...\\Administrator (54)* ×
--FORMAT
--sayı veya tarih türündeki değerleri istediğimiz formatta yazdırımızı sağlar

SELECT GETDATE()

--FORMATı kullanalım
SELECT FORMAT(GETDATE(), 'D', 'en-us')

--D yi d olarak değiştirelim:
SELECT FORMAT(GETDATE(), 'd', 'en-us')
--Başka bir uygulama
SELECT FORMAT(GETDATE(), 'd', 'en-gb') AS 'US English Result'
      ,FORMAT(GETDATE(), 'd', 'de-de') AS 'Great Britain English Result'
      ,FORMAT(GETDATE(), 'd', 'zh-cn') AS 'German Result'
      ,FORMAT(GETDATE(), 'd', 'en-tr') AS 'Simplified Chinese (PRC) Result';
--Başka bir uygulama
SELECT FORMAT(GETDATE(), 'D', 'en-gb')
SELECT FORMAT(GETDATE(), 'D', 'de-de')
SELECT FORMAT(GETDATE(), 'D', 'zh-cn')
SELECT FORMAT(GETDATE(), 'D', 'en-tr')

107 % ◀ ▶
Results Messages
US English Result Great Britain English Result German Result Simplified Chinese (PRC) Result
1 07/12/2021 07.12.2021 2021/12/7 12/7/2021

Query executed successfully. | SQLEGITIM (14.0 RTM) | SQLEGITIM\Administrato... | master | 00:00:00 | 1 rows
```

11.8_stringislemeleriGiris_Left_Right_Len

11.8_stringislemeleri...\Administrator (51)* X

--LEFT, RIGHT, LEN

```
SELECT LEFT('ÖMER ÇOLAKOĞLU',4) --Soldan 4 harfi alır
```

107 %

Results Messages

	(No column name)
1	ÖMER

Query executed successfully. | localhost (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 1 rows

11.8_stringislemeleri...\Administrator (51))* X

```
SELECT RIGHT('ÖMER ÇOLAKOĞLU',4) --Sağdan 4 harfi alır
```

107 %

Results Messages

	(No column name)
1	OĞLU

Query executed successfully. | localhost (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 1 rows

11.8_stringislemeleri...\Administrator (51))* - X

```
SELECT LEN('ÖMER ÇOLAKOĞLU') --Stringin uzunluğunu verir(aradaki boşluk dahil)
```

107 %

Results Messages

	(No column name)
1	14

✓ Query executed successfully. | localhost (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 1 rows

11.8_stringislemeleri...\Administrator (51)* X

```
--Uygulama
--Charindex ile bir karaktere kadar kelimeyi alabiliriz
--Örneğin Soldan başlayarak boşluğa kadar kalimeyi yazdıralım
SELECT LEFT('ÖMER ÇOLAKOĞLU',CHARINDEX(' ','ÖMER ÇOLAKOĞLU'))
```

107 % ◀ ▶

Results Messages

(No column name)
1 ÖMER

Query executed successfully. | localhost (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 1 rows

11.8_stringislemeleri...\Administrator (51)*

```
--Uygulama2  
--Şimdi ise sadece Soyismi getirelim  
--Eğer biz ÖMER ÇOLAKOĞLU'dan ÖMER'i çıkarırsak ÇOLAKOĞLU'na ulaşmış oluruz  
SELECT RIGHT('ÖMER ÇOLAKOĞLU',LEN('ÖMER ÇOLAKOĞLU')-CHARINDEX(' ','ÖMER ÇOLAKOĞLU'))
```

107 %

Results Messages

	(No column name)
1	ÇOLAKOĞLU

Query executed successfully.

| localhost (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 1 rows

```
--Uygulama3
--Şimdide tüm verilerin isim ve soy ismini ayrı yazalım
SELECT
    LEFT(NAMESURNAME,CHARINDEX(' ',NAMESURNAME)),RIGHT(NAMESURNAME,LEN(NAMESURNAME)-CHARINDEX(' ',NAMESURNAME))
FROM USERS
```

107 %

Results | Messages

	(No column name)	(No column name)
1	Nazlıcan	ÖZSİMITÇİ
2	Emirhan	SELİM
3	Soner	ÜLGEN
4	Çetin	BÖRKLU
5	Ezgi	İBUKÜRTÜNCÜ
6	Hazal	ÜREGİL
7	Aykut	SUYUR
8	Onur	KIRIT
9	Songül	TÜKEZİM
10	Berkay	PİRİNÇAL
11	Anıl	GÜLDÜ
12	Sami	OKÇE
13	Muzaffer	AĞAÇKESEN
14	Nimet	HAYDARV
15	Mina	KALO

Query executed successfully.

localhost (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 9.678 rows

11.9_stringislemeleriGiris_ LOWER,UPPER,REVERSE,REPLICATE

```
11.9_stringIslemeleri...\Administrator (51)) ➔ X
--TRIM LTRIM RTRIM
--TRIM boşlukları temizlemek anlamına geliyor
SELECT TRIM(' ÖMER ÇOLAKOĞLU ')--Baş ve sondaki boşluğu temizler
SELECT RTRIM(' ÖMER ÇOLAKOĞLU ')--Sondaki(Sağdaki) boşluğu temizler
SELECT LTRIM(' ÖMER ÇOLAKOĞLU ')--Başdaki(Soldaki) boşluğu temizler

SELECT LTRIM(RTRIM(' ÖMER ÇOLAKOĞLU '))--TRIM ile aynı
```

107 %

Results Messages

Query executed successfully. | localhost (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 1 rows

11.10_stringIslemeleri_Lower_Upper_Reverse_Replicate

```
11.10_stringisleme..\Administrator (51)* - X
--LOWER Küçük Harfe Çevirir
--UPPER Büyük Harfe Çevirir
--REVERSE
--REPLICATE

SELECT LOWER('ÖMER') --Küçük Harfe Çevirir
SELECT UPPER('ÖMER') --Büyük Harfe Çevirir
```

107 %

Results Messages

Query executed successfully. | localhost (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 1 rows

11.10_stringişlemleme...\\Administrator (51)* X

```
SELECT REVERSE('ÖMER') --Tersten yazar
```

107 %

Results Messages

	(No column name)
1	REMÖ

Query executed successfully. | localhost (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 1 rows

11.10_stringislemle..\Administrator (51)*

```
--REPLICATE  
SELECT REPLICATE('0',10)-- 0 ' 10 kere yazdırır
```

107 %

Results Messages

	(No column name)
1	0000000000

Query executed successfully. | localhost (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 1 rows

11.11_stringislemleriGiris_Replace

11.11_stringislemle...\Administrator (51))* X

```
--REPLACE  
--Birşeyle birşeyi değiştirmek olarak tanımlayabiliriz.Örnekle açılayalım:  
--ÖMER ÇOLAKOĞLU ' undaki ÖMER'i ÖMER FARUK ile değiştirelim  
SELECT REPLACE('ÖMER ÇOLAKOĞLU','ÖMER','ÖMER FARUK')
```

107 %

Results Messages

(No column name)
1 ÖMER FARUK ÇOLAKOĞLU

Query executed successfully. | localhost (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 1 rows

```
11.11_stringislemle...\Administrator (51) X
--Uygulama cümle içerisindeki kelimeyi değiştirelim:
DECLARE @CUMLE AS VARCHAR(MAX)
SET @CUMLE='Şiirsellik olmasa Dünyanın şiri eksik kalırdı. Her zaman eksik kalacak bir listedir b
Sıralamanın şiirlerin niteliği ile ilgisi yoktur. Hem şiirler nasıl yarıştırılabilirler ki? Kendim
SET @CUMLE=REPLACE(@CUMLE, 'Dünyanın', 'Evrenin') --Dünyanın kelimesini Evrenin olarak değiştirdik.
SELECT @CUMLE
```

107 %

Results Messages

	(No column name)
1	Şiirsellik olmasa Evrenin şiri eksik kalırdı. H...

Query executed successfully. | localhost (14.0 RTM) | SQLEGITIM\Administrato... | ETRADE4 | 00:00:00 | 1 rows