

25.06.2018 TARİHLİ SQL SERVER NOTLARI

- Matematiksel Fonksiyonlar
- Kumeleme Fonksiyonlar
- Sıralama

Model 2010 olan turu devlet olan vasitaları getirin

```
select * from tbl_vasita
where model='2010' and turu='Devlet'
```

| Results | | Messages | | | | | | | | |
|---------|-----------|----------|-------------|-------|--------|--------------|---------|-----------|-----------------|--|
| | vasita_id | plaka | marka | model | turu | arac_kapi_no | ha_t_id | surucu_id | ulasim_sekli_id | |
| 1 | 1 | 34AB001 | temsa | 2010 | devlet | 101-25368 | 1 | 1 | 1 | |
| 2 | 2 | 34AB002 | temsa | 2010 | devlet | 101-25369 | 2 | 2 | 1 | |
| 3 | 3 | 34AB003 | temsa | 2010 | devlet | 101-25370 | 3 | 3 | 1 | |
| 4 | 4 | 34AB004 | temsa | 2010 | devlet | 101-25371 | 4 | 4 | 1 | |
| 5 | 5 | 34AB005 | temsa | 2010 | devlet | 101-25372 | 5 | 5 | 1 | |
| 6 | 6 | 34AB006 | temsa | 2010 | devlet | 101-25373 | 6 | 6 | 1 | |
| 7 | 7 | 34AB007 | temsa | 2010 | devlet | 101-25374 | 7 | 7 | 1 | |
| 8 | 8 | 34AB008 | temsa | 2010 | devlet | 101-25375 | 8 | 8 | 1 | |
| 9 | 9 | 34AB009 | temsa | 2010 | devlet | 101-25376 | 9 | 9 | 1 | |
| 10 | 10 | 34AB010 | temsa | 2010 | devlet | 101-25377 | 10 | 10 | 1 | |
| 11 | 11 | 34AB011 | temsa | 2010 | devlet | 101-25378 | 11 | 11 | 1 | |
| 12 | 12 | 34AB012 | temsa | 2010 | devlet | 101-25379 | 12 | 12 | 1 | |
| 13 | 13 | 34AB013 | temsa | 2010 | devlet | 101-25380 | 13 | 13 | 1 | |
| 14 | 58 | 34AB141 | BMC Procity | 2010 | devlet | 101-25508 | 58 | 58 | 1 | |
| -- | -- | -- | -- | -- | -- | -- | -- | -- | -- | |

Son kullanma tarihi bugün biten yolcular

```
select * from tbl_yolcu
where skt='2018-06-25'
```

| Results | | Messages | | | | | | | | |
|---------|----------|-------------|---------------------------|--------------|----------|------------|--------|-------|---------|--|
| | yolcu_id | yolcu_tc | adi_soyadi | dogum_tarihi | fotograf | skt | bakiye | limit | kart_id | |
| | 577 | 2222222798 | Ataalp Alver | 1976-08-30 | NULL | 2018-06-25 | 20,00 | 0 | 1 | |
| | 1795 | 22222224016 | Rabia Kübra Acar | 1979-12-31 | NULL | 2018-06-25 | 0,00 | 14 | 4 | |
| | 3826 | 22222226047 | Öğünç Adanır | 1985-07-23 | NULL | 2018-06-25 | 0,00 | 64 | 1 | |
| | 4562 | 22222226793 | İrem Ceren Adıgüzel | 1987-07-29 | NULL | 2018-06-25 | 43,00 | 0 | 2 | |
| | 5780 | 22222228001 | Servet Ece naz Ağa | 1990-11-28 | NULL | 2018-06-25 | 0,00 | 37 | 3 | |
| | 7811 | 22222230032 | Gür al Ağır oğlu | 1996-06-20 | NULL | 2018-06-25 | 0,00 | 87 | 5 | |
| | 8547 | 22222230768 | Gülçe Ağın ar | 1998-06-26 | NULL | 2018-06-25 | 66,00 | 0 | 5 | |
| | 9765 | 22222231986 | Nesrin Gül ke Ahm adı Asl | 2001-10-26 | NULL | 2018-06-25 | 0,00 | 60 | 1 | |

mahalle adların da **haci** yada **koca** ifadesi gecen mahalleleri listeleyin

şikayet tarihi **2016-04-08** ile **2017-04-07** arasında olmayan şikayetleri listeleyin

```
select * from tbl_sikayet  
where sikayet_tarihi  
between '2016-04-08' and '2017-04-07'
```

10 yıl önce bugün doğan yolcuların limitini 200 tl olarak güncelleyin

şikayetlerde kavga kelimesi geçen şikayetleri «surucu yolcu ile tartıştı olarak »güncelleyin

```
update tbl_sikayet  
set sikayet='surucu yolcu ile tartıştı '  
where sikayet like '%kavga%'
```

SELECT POWER(P1,P2)

NOT:P1 parametresi ile girilen sayının P2 parametresi ile girilen sayı kadar kuvvetini alır.

Örnek:





```
1  
2 Select power(3,4) -- Sonuc 81 (3*3*3*3 = 81)  
3
```

```
1  
2 select power(2,3)--cevap olarak 8 değerini döndürür.  
3
```

SELECT ABS()

NOT:Matematiksel bir fonksiyondur.Sayının mutlak değerini döndürür.

Örnek:



```
1  
2 select Abs(-3) --Ekran çıktısı 3 tür.  
3
```

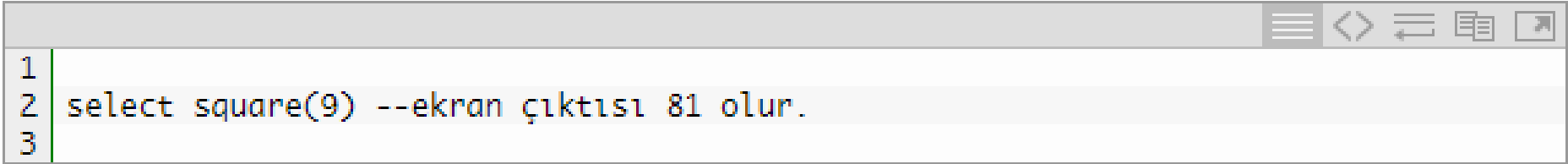


```
1  
2 select Abs(3) --Ekran çıktısı 3 tür.  
3
```

SELECT SQUARE()

NOT:Matematiksel bir fonksiyondur ve sayının karesini alır.

Örnek:

A code editor window with a light gray header bar containing icons for menu, code, undo, redo, and zoom. The editor area has a light gray background with line numbers 1, 2, and 3 on the left. Line 2 contains the SQL query: `select square(9) --ekran çıktısı 81 olur.`

```
1  
2 select square(9) --ekran çıktısı 81 olur.  
3
```

SQRT fonksiyonu matematiksel olarak karekök alma işlemini yapmaktadır. Bu fonksiyon tek parametreye ihtiyaç duyar. Belirtilen verinin karekökünü alır. Örnekleyelim :

```
1 SELECT SQRT(3)
2 SELECT SQRT(16)
3 SELECT SQRT(21)
```

Sırası ile 3, 16 ve 21 rakamlarının karekökünü aldım.

| Results Messages | |
|------------------|------------------|
| (No column name) | |
| 1 | 1,73205080756888 |
| (No column name) | |
| 1 | 4 |
| (No column name) | |
| 1 | 4,58257569495584 |

SELECT RAND()

NOT:0 ve 1 arasında **rastgele** ondalıklı değer üretir.Örnekleri inceleyerek daha iyi kavrayacaksınız.

Örnek:0 ve 50 arasında değer üretiniz.

```
1  
2 select rand()*50 --0 ve 50 arasında rastgele değer üretir.  
3
```

0 ve 100 arasında değer üretiniz.

```
1  
2 select rand()*100 --0 ve 100 arasınnda rastgele değer üretir.  
3
```

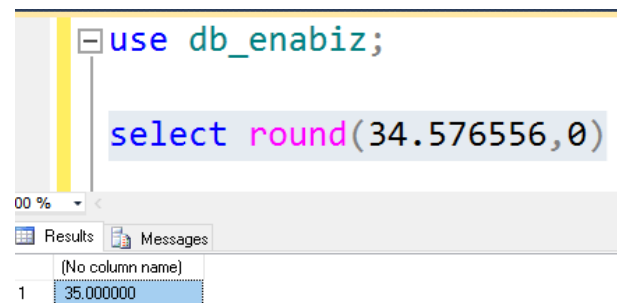
SELECT ROUND(P1,P2,[P3])

NOT:Yuvarlama fonksiyonudur.Round fonksiyonu 2 değerle iş yapar ama istenilirse 3 değer de alabilir.

P1 = Her hangi bir tam veya ondalıklı sayı yazılır.

P2 = Sayının ondalıklı kısmının kaç basamaklı olmasını istiyorsak o sayı girilir. Sayı negatif girilirse tam sayı kısmını da yuvarlar

```
1  
2 Select round(25.3453453,2) -- Sonuç 25,35  
3
```

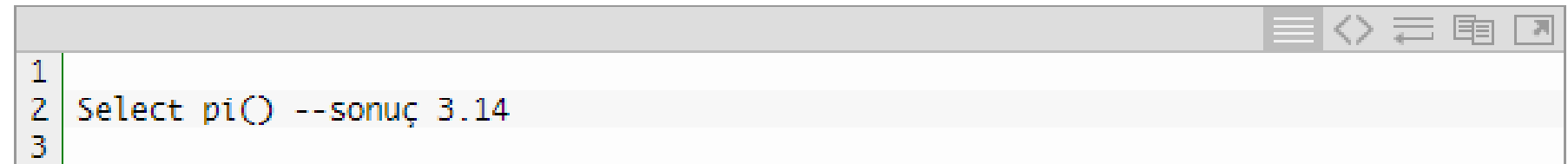


```
select * from tbl_surucu  
where surucu_id=round(  
rand()*1000,0)
```

SELECT PI()

NOT:Pi sayısını listeler

ÖRNEK

A terminal window with a light gray title bar containing icons for menu, navigation, and window management. The terminal has a light gray background with a vertical green cursor line at the start of the first line. The text '1' is on the first line, '2 select pi() --sonuç 3.14' is on the second line, and '3' is on the third line.

```
1  
2 select pi() --sonuç 3.14  
3
```

SELECT SIGN()

NOT:P1 parametresi ile girilen sayının işaretini öğrenmemizi sağlar

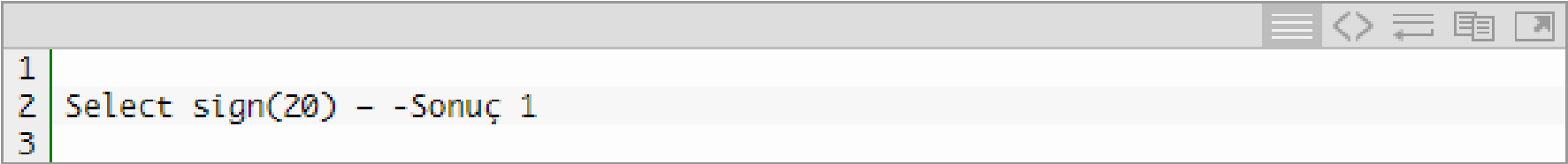
P1 parametresi ile girilen sayı pozitif ise 1

P1 parametresi ile girilen sayı negatif ise -1

P1 parametresi ile girilen sayı sıfır ise 0

Sonucunu görürüz.

ÖRNEK



```
1  
2 Select sign(20) - -Sonuç 1  
3
```

SELECT İSNUMERIC()

NOT:P1 parametresi ile girilen ifadenin veri türünün sayısal olup olmadığını verir. Sayısal ise 1 değilse 0 değeri döndürür.

A code editor window with a light gray background and a dark gray title bar. The title bar contains icons for a menu, a back/forward arrow, a refresh/cancel icon, a print icon, and a window icon. The editor area has a line number column on the left with numbers 1, 2, and 3. The text in the editor is "Select isnumeric('100') --- 1".

```
1  
2 Select isnumeric('100') --- 1  
3
```

A code editor window with a light gray background and a dark gray title bar. The title bar contains icons for a menu, a back/forward arrow, a refresh/cancel icon, a print icon, and a window icon. The editor area has a line number column on the left with numbers 1, 2, and 3. The text in the editor is "Select isnumeric('a') --- 0".

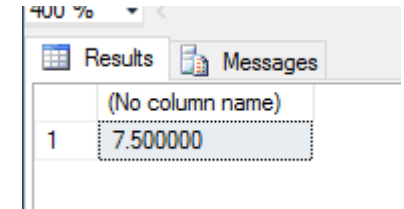
```
1  
2 Select isnumeric('a') --- 0  
3
```

Kenar uzunlukları 3,4 olan dikdörtgenin alanını hesaplayan sql

```
select 3*4
```

Dik uzunlukları 3,5 olan üçgenin alanını hesaplayın

```
select 3.0*5.0/2
```

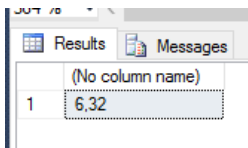
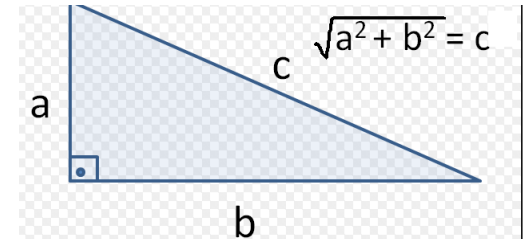


Results Messages

| | (No column name) |
|---|------------------|
| 1 | 7.500000 |

A=2 , b=6 için c=?

```
select round( sqrt(  
square(2)+square(6)),2)
```



Results Messages

| | (No column name) |
|---|------------------|
| 1 | 6.32 |

KÜMELEME FONKSİYONLARI

AVG() Fonksiyonu

Ortalama hesaplar. Örneğin çalışanların ortalama ne kadar maaş aldığını hesaplamak istersek:

```
SELECT AVG(maaş) FROM kisiler
```

Sorgunun çıktısı aşağıdaki gibi olacaktır:

AVG(maaş)

2872.7273

Maasları 2000 ile 3000 arasında olan surucuların maas ortalamasını getiren sql

```
select avg(maas) as iyi from tbl_surucu_ssk  
where maas between 1000 and 3000
```



```
select sum(maas)/count(maas) as kotu from  
tbl_surucu_ssk  
where maas between 1000 and 3000
```



364 % <

Results Messages

| | iyi |
|---|-----------|
| 1 | 2810,3305 |

| | kotu |
|---|-----------|
| 1 | 2810,3305 |

Eğer çıktının biraz daha okunur olmasını arzu edersek sorgu üzerinde ufak bir değişiklik yapabiliriz:

```
SELECT AVG(maaş) AS OrtalamaMaaş FROM kisiler
```

Sorgunun çıktısı aşağıdaki gibi olacaktır:

| OrtalamaMaaş |
|--------------|
|--------------|

| |
|-----------|
| 2872.7273 |
|-----------|

```
select round(avg(ucret)/10,0) as  
ortalama from tbl_ucret
```

Eğer yalnızca Türkiye’de çalışanların maaş ortalamasını hesaplamak istersek:

```
SELECT AVG(maaş) AS OrtalamaMaaş FROM kisiler WHERE ülke='Türkiye'
```

Sorgunun çıktısı aşağıdaki gibi olacaktır:

| OrtalamaMaaş |
|--------------|
|--------------|

| |
|-----------|
| 2414.2857 |
|-----------|

```
select avg(ucret) as ortalama from tbl_ucret  
where ucret between 20 and 40
```

COUNT() Fonksiyonu

Bir alandaki değerlerin kaç adet olduğunu yani sayısını hesaplar. Örneğin kaç çalışanın olduğunu hesaplamak istersek:

```
SELECT COUNT(maaş) AS Sayı FROM kisiler
```

Sorgunun çıktısı aşağıdaki gibi olacaktır:

| Sayı |
|------|
|------|

| |
|----|
| 11 |
|----|

```
select count(*) as kayıt from  
tbl_ucret  
where ucret between 20 and 40
```

COUNT(*) Kullanımı

Eğer kesin olarak kayıt sayısını istiyorsak her zaman ifadeyi COUNT(*) şeklinde kullanmalıyız. Bu ifade her zaman tablodaki ya da sonuç kümesindeki kayıt sayısını (satır sayısını) döndürür.

```
SELECT COUNT(*) AS Sayı FROM kisiler
```

Sorgu kisiler tablosunda kaç kayıt olduğunu hesaplar:

| Sayı |
|------|
|------|

| |
|----|
| 11 |
|----|

DISTINCT

Bazen bir tablonun bazı kolonlarında **tekrarlanan değerler (veriler)** olabilir. Ancak tekrarlanan verileri eleyerek **her farklı veriden yalnız bir adet** bulunmasını istiyorsak **DISTINCT** anahtar sözcüğünü kullanırız

Örnek-1: Eğer yukarıdaki tabloda yalnızca farklı şehirleri seçmek istiyorsak:

```
SELECT DISTINCT Şehir FROM Kişiler
```

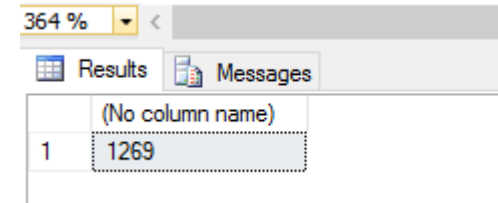
Sonuç kümesi (result-set) aşağıdaki gibi olacaktır:

| Şehir |
|----------|
| İstanbul |
| Ankara |

```
select distinct tur from  
tbl_hastane
```

Kac tane marka var?

```
select count( marka) from  
tbl_vasita
```



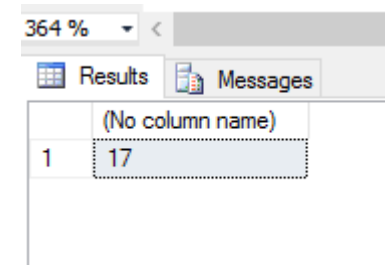
364 % <

Results Messages

| | (No column name) |
|---|------------------|
| 1 | 1269 |

Kac farklı marka var?

```
select count( distinct marka) from  
tbl_vasita
```



364 % <

Results Messages

| | (No column name) |
|---|------------------|
| 1 | 17 |

Farklı markaları getiren sql

```
select distinct marka from  
tbl_vasita
```



364 % <

Results Messages

| | marka |
|----|----------------------|
| 1 | Karsan BM Avancity S |
| 2 | Mercedes Conecto (Kö |
| 3 | BMC Procity |
| 4 | Breda |
| 5 | Ustaoglu |
| 6 | temsa |
| 7 | Meredes Citaro (Körü |
| 8 | Karsan BM Avancity+C |
| 9 | Mercedes Conecto G |
| 10 | Otokar Kent 290 LF |
| 11 | Phileas (Körükü) |
| 12 | Mercedes Citaro (Sol |
| 13 | Mercedes Capacity (K |
| 14 | Dumazlar |
| 15 | Gotha |
| 16 | Rotem |

COUNT (DISTINCT kolon_adi) Kullanımı

Eğer bir kolondaki farklı değerlerin sayısını öğrenmek istiyorsak COUNT ifadesini COUNT (DISTINCT kolon_adi) şeklinde kullanırız. Örneğin kaç farklı ülkeden çalışan olduğunu hesaplamak istersek:

```
SELECT COUNT(DISTINCT ülke) AS ÜlkeSayısı FROM kisiler
```

Tabloda ülke alanında 4 farklı değer (Türkiye, USA, Fransa, Almanya) olduğundan çıktı aşağıdaki gibi olacaktır.

Ülke Sayısı

4

```
select count(distinct tur) from  
tbl_hastane
```


MAX() ve MIN() Fonksiyonları

Bir alandaki en büyük ve en küçük değerleri döndürürler. En yaşlı ve en genç çalışanları belirlemek istersek:

```
SELECT MAX(yas) AS EnYaşlı FROM kisiler
```

Çıktı aşağıdaki gibi olacaktır:

EnYaşlı

45

```
SELECT MIN(yas) AS EnGenç FROM kisiler
```

Çıktı aşağıdaki gibi olacaktır:

EnGenç

20

Dikkat: Eğer en yaşlı çalışanı ismi ile birlikte sorgulamak isteseydik aşağıdaki gibi bir sorgunun işimizi görebileğini düşünebilirsiniz. Ancak bu sorgu hatalıdır.

```
SELECT ad, MAX(yas) AS EnYaşlı FROM kisiler
```

Sorgu ad alanına tablodaki ilk ismi getirirken MAX(yas) alanına maksimum yaşı getirir:

| ad | EnYaşlı | <pre>select min(ucret) from tbl_ucret where ucret between 20 and 40</pre> |
|-------|---------|---|
| Ahmet | 45 | |

Ancak Ahmet isimli çalışanın yaşı 45 değildir. Fonksiyonun sadece maksimum değer ile ilgilendiğini unutmayalım. En yaşlı çalışanı ismi ile birlikte bulabilmek için iç-içe (iki seviyeli) sorgu yazmamız gerekiyor ki bu da ileriki bir makalenin konusu olacak inş.

```
select max(ucret) from tbl_ucret
where ucret between 20 and 40
```

SUM() Fonksiyonu

Sayısal değerler içeren bir kolondaki değerlerin toplamını döndürür. Örneğin aylık ödenen toplam maaşı öğrenmek istersek:

```
SELECT SUM(maaş) AS ToplamMaaş FROM kisiler
```

Çıktı aşağıdaki gibidir:

| ToplamMaaş |
|------------|
|------------|

| |
|-------|
| 31600 |
|-------|

```
select sum(ucret)/count(ucret) as toplam from tbl_ucret  
select avg(ucret) as toplam from tbl_ucret
```

ORDER BY İfadesi

Order by komutu da sıralama yapılmak istenildiğinde kullanılır.
Artan(asc) ve **azalan(desc)** parametleri ile kullanılabilir.
Parametre almaz ise default artan şeklinde sıralama yapar.

//Aşağıdaki sorguda vatandaş tablosunda yaşı büyük olandan küçüğe olana doğru sıralar

```
select * from vatandas order by yas desc
```

//Aşağıdaki sorguda vatandaş tablosunda yaşı büyük olandan küçüğe olana doğru sıralar

```
select * from vatandas order by yas desc
```

//Aşağıdaki kod ise tam tersi şekilde küçükten büyüğe sıralama yapar.

```
select * from vatandas order by yasi asc
```

TOP

SQL sorgularında gelen kayıt sayısı bazen performans sorunu oluşturabiliyor. Çok fazla kayıt olan tablolardaki sorguların performansını artırmak için bu gelen kayıt sayılarını sınırlandırabiliyoruz. Bu şekilde performans sıkıntısı ortadan kalkıyor.

SELECT TOP 3 AD_SOYAD, NOT_ORTALAMASI FROM SONUCLAR;

```
select top 2 * from tbl_hasta
where adres like '%istanbul%'
order by hasta_adi
```

Alıştırmalar

- 1) Pi sayısının 4 üncü kuvvetinin karesini alan sql cümlesi
- 2) Maaşı 2400 olan ları listeleyin

1) Yolcuların skt si bitenleri listleyin

2) Yolculardan bakiyesi 30- 60 arası olanları isme gore alfabetik listeleyin

1) Yolculardan birini rastgele seçin

2) $\left(\sqrt{4 * \sqrt{4}}\right)^3 / |2 - \sqrt{5}|$ formülünü sql komutuyla hesaplayınız

11 nolu mahallede kaç adet durak vardır?

Sktsi gemeyer yolcuların limit toplamaları nedir

devlet turunde olan 2015 model vasıtaları markalarına
göre alfabetik listeleyin

vardiyaların sayısının karesini getirin

3 numaralı karta sahip yolcuların bakiyesi ortama kaç tl dir . Ondalıklı ise tamsayıya yuvarlayın

GROUP BY İfadesi

GROUP BY ifadesi gruplama yapar. Yani sonuç kümesini bir veya birden fazla kolona göre gruplar.

```
SELECT ülke FROM `kisiler` GROUP BY ülke
```

```
select marka, model, count(*) as vasita_sayisi  
from tbl_vasita group by marka, model
```

```
select * from tbl_vasita where marka='Karsan  
BM Avancity S'
```

```
select count(*), baslama_saati, bitis_saati from tbl_vardiya group by  
baslama_saati, bitis_saati
```

Aliřtırmalar

vasitalarda her marka turunde kac adet vasita var?

suruculer maasa gre hesaplandıėında her maař turunde kac adet surucu var

3)Güzergah tablosunda her hatta kac durak verilmiştir

3)Güzergah tablosunda her durakta kaç hat var?