

SORGULAR

SORGULAR VE ÇEŞİTLERİ

- **Seçme Sorguları:** En sık kullanılan sorgu türüdür. Seçme sorguları, bilgileri “veri sayfası görünümü”nde gösteren veri tabanı nesnesi türüdür. Sorgu, verileri bir veya birden fazla tablodan, mevcut sorgulardan veya bunların her ikisinden alabilmektedir.
- **Parametre Sorguları:** Parametre sorguları, çalıştırıldığı zaman bir ölçüt girilmesini sağlayan iletişim kutusunu açan sorgulardır. Örneğin bir okuldaki personelin ocak ayında sevk aldığı günleri görmek istiyorsak, açılan iletişim kutusuna ölçüt olarak istenilen tarih aralıkları girilir ve bu tarihler arasındaki veriler listelenir.

SORGULAR VE ÇEŞİTLERİ

- **Çapraz Sorgular:** Bir tablodaki bilgileri analiz etmek, karşılaştırmak ve tablonun özetini hazırlamak için kullanılan sorgu türüdür. Belirtilen iki alana göre istenilen işlemi (toplama, ortalama, vs.) tablo şeklinde göstermekte kullanılır.
- **Eylem Sorguları:** Tek işlemle birçok kayıta değişiklik yapan sorgulardır. Silme sorgusu, güncelleştirme sorgusu, ekleme sorgusu ve tablo yapma sorgusu olmak üzere dört çeşit eylem sorgusu vardır.
- **SQL Sorguları:** SQL deyimlerini kullanarak veri tabanını güncelleştirme ve yönetme ile ilgili oluşturulan sorgulardır.

SQL Dilinin Yapısı

- Sorgu İle Tablo Oluşturma

Sorgu kullanarak tablo oluşturmak için CREATE TABLE ifadesi kullanılır.

SQL kodu yazılarak tablo oluşturma aşağıdaki şekilde yapılmaktadır.

Kullanımı:

```
CREATE TABLE tablo_adi
```

```
(
```

```
Sütun1 veri tipi,
```

```
Sütun2 veri tipi,
```

```
·
```

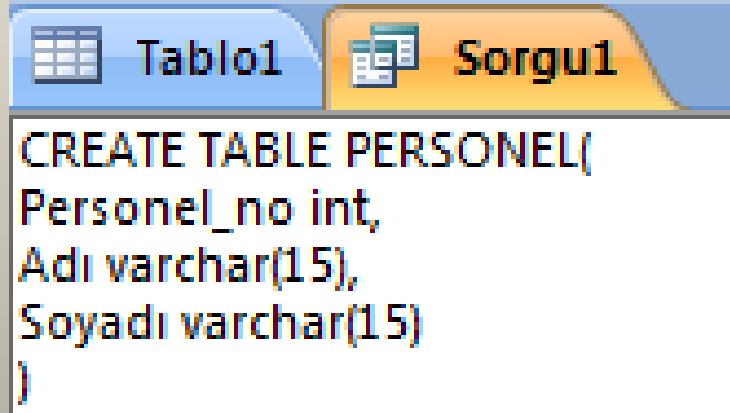
```
·
```

```
sütunN veri tipi
```

```
)
```

SQL Dilinin Yapısı

- Veri tabanını oluşturduktan sonra Oluştur sekmesi altında yer alan Diğer grubundan Sorgu Tasarımı düğmesine tıklanır. Sorgu Tasarımı seçeneği sorgu tasarımcısının açılmasını sağlar ve Tabloyu Göster iletişim kutusu görüntülenir.



The screenshot shows a software interface with two tabs at the top: 'Tablo1' (blue) and 'Sorgu1' (orange). Below the tabs, a text area contains the following SQL code:

```
CREATE TABLE PERSONEL(  
Personel_no int,  
Adı varchar(15),  
Soyadı varchar(15)  
)
```

- Çalıştır düğmesine basıldığında aşağıdaki sonuç elde edilir.

SQL Dilinin Yapısı

- **Tablo Silme**
- Daha önceden oluşturmuş olduğunuz bir tabloya ihtiyaç duymayıp veri tabanından silmek isteyebilirsiniz. Silme işlemini gerçekleştirmek için DROP deyimi kullanılmaktadır.
- **Kullanımı:**
- DROP TABLE tablo
- **Örnek:**
- DROP TABLE PERSONEL

SQL Dilinin Yapısı

- **Sütun Ekleme**
- Bir tablo oluşturduktan sonra, isteğe göre oluşturulmuş olan tabloya yeni sütunlar ekleyebilirsiniz. Tabloya yeni bir sütun eklemek için **ADD COLUMN** deyimi kullanılır.

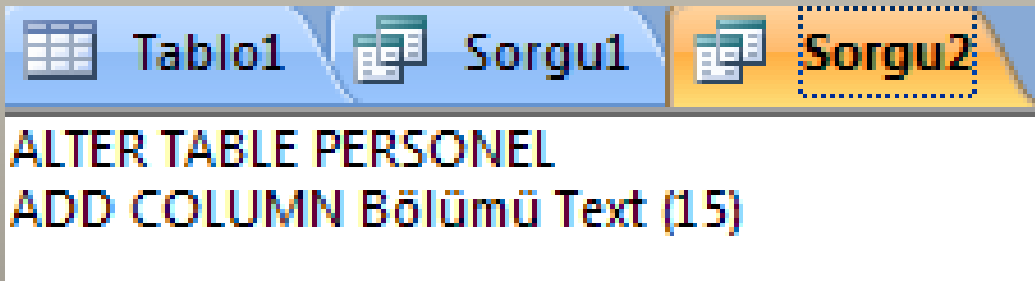
- **Kullanımı:**

```
ALTER TABLE tablo_adi
```

```
ADD sütun_adi, veri_türü
```

- **Örnek:**

Önceden oluşturulmuş olan **PERSONEL** isimli tabloya “**Bölümü**” sütununu eklemek istersek SQL ifadesi aşağıdaki şekilde olacaktır.



```
ALTER TABLE PERSONEL  
ADD COLUMN Bölümü Text (15)
```

SQL Dilinin Yapısı

- **Tablo Güncelleme**
- Belirtilen tablodaki alanların değerlerini belirtilen ölçütlere göre değiştirmek için bir güncelleme sorgusu oluşturmak gerekir. Bunun için UPDATE deyimi kullanılır.
- **Kullanımı:**

UPDATE *tablo*

SET *sütun_adi*=*yenideğer*

WHERE *ölçütler*;

SQL Dilinin Yapısı

- **Örnek:** *Ücret tablosundaki kayıtlarda, Maaşı isimli alanda yer alan değerlerden % 0,07 kesinti yaparak sonucu yine aynı alan üzerinde kaydetmek istensin. Ücret tablosunun kesinti yapılmadan önceki hali aşağıdaki gibidir.*

Personel_no	maaşı
1	5000
2	2000
3	3000
4	1500
*	Yeni

```
UPDATE Ücret
SET maaşı=maaşı-0.007*maaşı
```

SQL Dilinin Yapısı

- **SELECT Deyiminin Yapısı**

Veri tabanında verilere erişebilmek için SELECT deyimi kullanılmaktadır. Select deyimi ile bir tabloda bulunan belli bir sütun, birden fazla sütun veya tüm sütunları çekebilirsiniz.

- **Kullanımı:**

- `SELECT [sütun_listesi] FROM [tablo_listesi]`

- `SELECT adı FROM PERSONEL`

	adı
	ali
	veli
	ayşe
	mehmet
*	

SQL Dilinin Yapısı

- **SELECT * FROM PERSONEL**

Personel_no	adı	Soyadı	Bölümü	İl_Kodu
1	ali	ak	matematik	34
2	veli	kara	matematik	34
30	ayşe	ay	ingilizce	35
44	mehmet	tek	müzik	07
*				

- **SELECT adı, Soyadı FROM PERSONEL**

adı	Soyadı
ali	ak
veli	kara
ayşe	ay
mehmet	tek
*	

SQL Dilinin Yapısı

- **SELECT adı, Soyadı, adı +” “ + Soyadı FROM PERSONEL**

adı	Soyadı	Expr1002
ali	ak	ali ak
veli	kara	veli kara
ayşe	ay	ayşe ay
mehmet	tek	mehmet tek
*		

SQL Dilinin Yapısı

- **Verilerin Sınırlandırılması**
- Veri tabanında veriyi alma işlemi sırasında satırlara birtakım sınırlamalar getirilerek tablonun tüm satırları yerine istenildiği kadarını elde etmek mümkündür. Tabloda belirli kısımları seçme işlemi gerçekleştirilmek için WHERE sözcüğü kullanılmaktadır.

- **Kullanımı:**

- SELECT [sütunlar]

FROM [tablo]

WHERE [koşul]

SQL Dilinin Yapısı

- **Örnek:**
- PERSONEL tablosunda yer alan İl_kodu 34 olan kayıtları listelemek istersek komut satırı aşağıdaki şekilde olacaktır.

```
Sorgu1 Sorgu9
SELECT *
FROM PERSONEL
WHERE İl_Kodu="34"
```

Personel_no	adı	Soyadı	Bölümü	İl_Kodu
1	ali	ak	matematik	34
2	veli	kara	matematik	34

SQL Dilinin Yapısı

- **Örnek:**
- Ücret tablosunda, Personel_no ve maaşı alanları görülmektedir.

Personel_no	maaşı	Yeni alan ekleyin
1	4965	
2	1986	
3	2979	
4	1490	

```
SELECT Personel_No, Maaşı FROM Ücret WHERE Maaşı > 2000;
```

Personel_No	Maaşı
1	4965
3	2979

SQL Dilinin Yapısı

KOŞUL	SORGU SONUCU
200	Maaşın 200 TL olduğu kayıtları döndürür.
Not 2000	Maaşın 2000 TL olmadığı kayıtları döndürür.
< 200	Maaşın 200 TL'den az (<200) olduğu kayıtları döndürür. İkinci ifade (<=200) birim fiyatın 200 TL'ye eşit veya daha az olduğu kayıtları görüntüler.
<= 200	
>100 >=100	Maaşın 100TL'den fazla olduğu kayıtları döndürür. İkinci ifade maaşın 100 TL'ye eşit veya daha fazla olduğu kayıtları görüntüler.
100 veya 125	Maaşın 100 TL veya 125 TL olduğu kayıtları döndürür.
>50 ve <100 -veya- Between 50 and 100	Maaşın 50 TL ile 100 TL arasında olduğu (ancak dahil değil) kayıtları döndürür.
<50 or >100	Maaşın 50 TL ile 100 TL arasında olmadığı kayıtları döndürür.
In(10, 22, 30)	Maaşın 10 TL, 22 TL veya 25 TL olduğu kayıtları döndürür.
Like "*4,99"	Maaşın sonunda "4,99" bulunan, 4,99 TL, 14,99 TL, 24,99 TL vb. kayıtları döndürür. NOT * ve % karakterleri bir ifadede kullanıldığı zaman, herhangi bir sayıda karakteri temsil eder (bunlara joker karakter de denir). %karakter * karakteriyle birlikte bir ifadede kullanılmadığı gibi, ? joker karakteriyle birlikte de bir ifadede kullanılamaz. % joker karakterini, _ joker karakterini de içeren bir ifadede kullanabilirsiniz.
Is Null	Maaşı alanına değer girilmeyen kayıtları döndürür.
Is Not Null	Maaşı alanında değer eksik olmadığı kayıtları döndürür.

SQL Dilinin Yapısı

- **Örnek:**
- Fiyatı 50 ile 100 TL arasında olan tüm ürünleri listelemek istersek SQL kodu aşağıdaki şekilde olacaktır.
- **SELECT * FROM URUN WHERE FİYAT BETWEEN 50 AND 100**
- İsmi “A” ile başlayan personeller listelenmek istenirse SQL kodu aşağıdaki şekilde olacaktır.
- **SELECT * FROM PERSONEL WHERE ADI LIKE “A%”**

SQL Dilinin Yapısı

- **Sıralama İşlemleri**
- Tabloların satırlarının herhangi bir sütuna göre sıralanmasının istendiği durumlarda SELECT deyimi **ORDER BY ile birlikte kullanılır.**

- **Örnek:**

- SELECT Soyadı, adı

FROM PERSONEL

ORDER BY Soyadı;

- **Örnek:**

- SELECT Soyadı, adı

FROM PERSONEL

ORDER BY Soyadı ASC;

	Soyadı	adı
	ak	ali
	ay	ayşe
	kara	veli
	tek	mehmet
*		

SQL Dilinin Yapısı

- Azalan şekilde sıralama yapmak için (Z'den A'ya, 9'dan 0'a), azalan şekilde sıralamak istenilen her alanın sonuna DESC sözcüğünün eklenmesi gerekir.

- **Örnek:**

- SELECT Soyadı, İl_Kodu

FROM PERSONEL

ORDER BY İl_Kodu DESC, Soyadı

	Soyadı	İl_Kodu
	ay	35
	ak	34
	kara	34
	tek	07
*		

Önümüzdeki Ders

SQL Fonksiyonları