

A spiral-bound notebook with a silver pen resting on it. The notebook is light blue and the pen is silver. The text is centered on the page.

4

SQL- Yapısal Sorgulama Dili

Veritabanı 1

SQL- Yapısal Sorgulama Dili

SQL ifadeleri yapısal olarak üç gruba ayrılır.

- **Veri Tanımlama Dili**
(DDL - Data Definition Language)
- **Veri İşleme Dili**
(DML - Data Manipulation Language)
- **Veri Kontrol Dili**
(DCL – Data Control Language)

Veri Tanımlama Dili (DDL-Data Definition Language)

- Veri tanımlama dili verinin ne olduğu ile değil verinin nerede ve nasıl tutulacağı ile ilgilenir.
- Veri tanımlama dilinin temel ifadeleri
 - **Create**; Nesne oluşturmak için kullanılır
 - **Alter**; Nesneler üzerinde değişiklik yapmak için kullanılır.
 - **Drop**; Nesneleri silmek için kullanılır.

Veri Tanımlama Dili - Create

- Create ; Veritabanı nesnelerini ve veritabanının kendisi oluşturmak için kullanılır.
 - CREATE DATABASE veritabanı ismi
 - CREATE TABLE tablo ismi ...
 - CREATE FUNCTION ...
 - CREATE PROCEDURE ...
 - ...

Veri Tanımlama Dili – Create Database

```
CREATE DATABASE db_dersOrnek
```

```
ON
```

```
    (NAME = db_dersOrnek           --(Dosya Adı)  
    , FILENAME= 'c:\db_dersOrnek.mdf' --(Fiziksel Dosya Adı)  
    , SIZE = 10                    --(Dosya Boyutu)  
    , MAXSIZE = 50 -               -(Maksimum Dosya Boyutu)  
    , FILEGROWTH =5)              --(Artım Miktarı)
```

```
LOG
```

```
ON
```

```
    (NAME = db_dersOrnek  
    , FILENAME= 'c:\db_dersOrnek.ldf'  
    , SIZE = 5  
    , MAXSIZE = 40  
    , FILEGROWTH =2)
```

Veri Tanımlama Dili – Create Table

- Veritabanı üzerinde yeni tablo oluşturur.
 - Sütun adları
 - Veri tipleri
 - Constraintler
 - Otomatik arttırma
 - NULL yada Not NULL

- Kullanım Şekli

CREATE TABLE <tablo_adi> (alan_adi
veri_türü)

Veri Tanımlama Dili – Create Table

```
CREATE TABLE tbl_ogrenci  
(  
    ogr_id INT IDENTITY(1,1) NOT NULL PRIMARY KEY,  
    ogrNo NVARCHAR(8) NOT NULL,  
    tcNo DECIMAL(11) NOT NULL UNIQUE,  
    ad NVARCHAR(100) NOT NULL,  
    soyad NVARCHAR(100) NOT NULL,  
    dogumTarihi DATETIME NULL,  
    dogumYeri NVARCHAR(50)  
)
```

Veri Tanımlama Dili – Create Table

```
CREATE TABLE tbl_ogrenciNot
```

```
(
```

```
  ogrNot_id INT IDENTITY(1,1) NOT NULL PRIMARY KEY,
```

```
  ogr_id INT NOT NULL FOREIGN KEY REFERENCES tbl_ogrenci(ogr_id),
```

```
  ders_id INT NOT NULL FOREIGN KEY REFERENCES tbl_ders(ders_id),
```

```
  notu TINYINT CHECK (notu>=0 and notu<=100),
```

```
  notGirisTarihi DATETIME NOT NULL DEFAULT GETDATE())
```


Veri Tanımlama Dili – Create Table

```
CREATE TABLE tbl_ogrenciNot(  
  ogrNot_id INT IDENTITY(1,1) NOT NULL,  
  ogr_id INT NOT NULL,  
  ders_id INT NOT NULL,  
  notu TINYINT,  
  notGirisTarihi DATETIME NOT NULL  
  CONSTRAINT pk_key PRIMARY KEY (ogrNot_id),  
  CONSTRAINT ck_not CHECK (notu >= 0 AND notu <= 100),  
  CONSTRAINT uk_ogr_id_ders_id UNIQUE (ogr_id, ders_id),  
  CONSTRAINT fk_key_ogr_id FOREIGN KEY (ogr_id) REFERENCES  
tbl_ogrenci(ogr_id),  
  CONSTRAINT fk_key_ders_id FOREIGN KEY (ders_id) REFERENCES  
tbl_ders(ders_id) ON UPDATE CASCADE ON DELETE CASCADE  
)
```

Veri Tanımlama Dili - ALTER

- Alter; Daha önce oluşturulmuş veritabanı nesnesinin özelliklerini değiştirmek için kullanılır.
 - **ALTER TABLE** tablo ismi ...
 - ALTER FUNCTION ...
 - ALTER PROCEDURE ...
 - ALTER TRIGGER
 - ...

Veri Tanımlama Dili - ALTER

- ALTER TABLE tablo ADD sütun_adi özellikler

ALTER TABLE ogrenci

ADD dogumTarihi DateTime NOT NULL

- ALTER TABLE tablo DROP COLUMN sütun_adi

ALTER TABLE ogrenci

DROP COLUMN dogumTarihi

- ALTER TABLE tablo ALTER COLUMN sütun_adi özellikler

ALTER TABLE ogrenci

ALTER COLUMN dogumTarihi DateTime NULL

Veri Tanımlama Dili - ALTER

- ALTER TABLE ogrenci ADD UNIQUE (ogrNo)
- ALTER TABLE ogrenciNot ADD UNIQUE (ders_id,ogr_id)
- ALTER TABLE ogrenci ADD PRIMARY KEY (ogr_id)
- ALTER TABLE ogrenci ADD CONSTRAINT pk PRIMARY KEY (ogr_id)
- ALTER TABLE ogrenciNot ADD CONSTRAINT fk FOREIGN KEY (ogr_id) REFERENCES ogrenci(ogr_id)
- ALTER TABLE ogrenciNot ADD CHECK (notu>=0 and notu <=100)

Veri Tanımlama Dili - DROP

- Drop; Veritabanı nesnelelerini ve veritabanının kendisi silmek için kullanılır.
 - **DROP DATABASE** veritabanı ismi
 - **DROP TABLE** tablo ismi ...
 - DROP FUNCTION ...
 - DROP PROCEDURE ...
 - ...

Veri Tanımlama Dili - DROP

- DROP TABLE öğrenci
- DROP DATABASE okul

Veri İşleme Dili (DML-Data Manipulation Language)

- Veri işleme dili tutulan veriler üzerinde işlem yapar.
- Veri işleme dilinin temel ifadeleri
 - **Select**; Veritabanındaki kayıtları sorgular
 - **Insert**; Yeni kayıt eklemek için kullanılır.
 - **Update**; Daha önce olan kaydı günceller
 - **Delete**; Veritabanındaki kaydı silmek için kullanılır.

Veri İşleme Dili - SELECT

- Select; Tablo yada tablolarda istenilen verinin seçimi için kullanılır. Seçilen verilerde değişim olmaz.
 - **SELECT** sütunlar **FROM** tablo ismi
 - **select** * **from** ogrenciler
 - **select** ogrNo, ad, soyad **from** ogrenciler

Veri İşleme Dili - SELECT

Özellikleri

- Tablonun her sütunu yazılabilir.
- Birden fazla tabloda sorgulama yapılabilir.
- Mevcut kayıtlar belirli koşullara göre seçilebilir.
- Sorgulama sırasında birden fazla sütun birleştirilebilir veya sütunlara ek bilgiler eklenebilir.
- Sorgulamada sütunlar sayısal tipte ise matematiksel işlemler yapılabilir.
- Sorgulamada dönen değerler birden fazla sütuna göre sıralı seçilebilir.
- Ortak sütunlara sahip tablolar ilişkilendirilerek farklı tablolardan ortak özelliklere sahip sütunlar seçilebilir.
- Daha karmaşık sorgular için iç içe select yazılabilir.

Veri İşleme Dili - SELECT

Personel Tablosu

per_id	adi	soyadi	gorevi	sehir
1	Ali	Coşkun	Mühendis	Denizli
2	Mehmet	Kara	Tekniker	Ankara
3	Ayşe	Con	Mühendis	Antalya

- `Select * from tbl_Personel`
- `select per_id, adi, gorevi from personel`

Veri İşleme Dili - SELECT

- Sorgulamada belirli koşullara bağlı kayıtlar isteniyorsa **Where** ifadesi kullanılır.
 - <, >, <=, >=, =, <>, !=
 - AND, OR, NOT
 - IN
 - BETWEEN
 - LIKE

Select * From tbl_personel where adi='Ali'

Personel Tablosu

per_id	adi	soyadi	gorevi	sehir
1	Ali	Coşkun	Mühendis	Denizli
2	Mehmet	Kara	Tekniker	Ankara
3	Ayşe	Con	Mühendis	Antalya

per_id	adi	soyadi	gorevi	sehir
1	Ali	Coşkun	Mühendis	Denizli
2	Mehmet	Kara	Tekniker	Ankara
3	Ayşe	Con	Mühendis	Antalya

- `SELECT * FROM tbl_personel WHERE per_id<3`
- `SELECT * FROM tbl_personel WHERE gorevi='Mühendis' AND sehir='Denizli'`
- `SELECT * FROM tbl_personel WHERE sehir='Denizli' OR soyadi='Con'`
- `SELECT * FROM tbl_personel WHERE sehir IN ('Denizli','Ankara')`
- `SELECT * FROM tbl_personel WHERE ad LIKE 'A%'`

Veri İşleme Dili - INSERT

- Insert; Tabloya yeni veri girişi için kullanılır.
- Kullanım şekilleri
 - **INSERT INTO** tablo_ismi **VALUES** (değerler)
 - insert into tbl_personel values ('Kemal','Kara','Güvenlik','Denizli')
 - **INSERT INTO** tablo_ismi (sutunAdları) **VALUES** (değerler)
 - insert into tbl_personel (adi,soyadi,gorevi,sehir) values ('Kemal','Kara','Güvenlik','Denizli')

Veri İşleme Dili - UPDATE

- Update; Tabloya daha önce girilmiş verilerin güncellenmesi için kullanılır.

- **UPDATE** tablo_ismi
SET sütun1=deger...
WHERE koşul

```
UPDATE tbl_ogrenci          --TcNo sisteme yanlışlıkla 1 olarak kayıt
SET tcNo=2                  -- edilmiş 2 olması gerekiyor. tcNo su 1
WHERE tcNo=1                -- olanın tcNo su 2 olarak güncellendi.
```

```
UPDATE tbl_ogrenciNot      -- ders_id 1 olan hoca bu dersi alan herkese
SET notu=notu+10           -- +10 puan veriyor herkesin notunu
WHERE ders_id=1            -- 10 puan arttırıp güncelledik
```

Veri İşleme Dili - DELETE

- Delete; Tabloya daha önce girilmiş verilerin silinmesi için kullanılır.

– **DELETE FROM** tablo_ismi
WHERE koşul

```
DELETE FROM tbl_ders  
WHERE dersKod ='geng 103'
```

--bu ders yanlışlıkla eklenmiş
--siliyoruz.

```
DELETE FROM tbl_ders  
WHERE ders_id=1
```

--bu dersi silmeye çalışalım

Veri Kontrol Dili

(DCL-Data Control Language)

- Veri tabanı üzerindeki verilere erişim yetkilerini düzenler.
- Veri kontrol dilinin temel ifadeleri
 - **Grant**; Kullanıcıların kayıtlar üzerinde işlem yapmasına izin verir.
 - **Deny**; Kullanıcının kayıtları kullanmasını kısıtlar
 - **Revoke**; Daha önce yapılan tüm kısıtlama ve izinleri iptal eder.

Veri Kontrol Dili - GRANT

- Grant; Kullanıcıya veritabanı veya nesnelere üzerinde çeşitli izinler vermek için kullanılır.

- **GRANT** izinler
- **ON** izin alanı
- **TO** kullanıcılar

VeriTabanı için	Tablo İçin	Görünüm İçin
CREATE DATABASE	DELETE	DELETE
BACKUP DATABASE	INSERT	INSERT
CREATE TABLE	SELECT	SELECT
CREATE VIEW	UPDATE	UPDATE

GRANT DELETE,INSERT, SELECT,UPDATE

ON tbl_personel

TO dersKull

Veri Kontrol Dili - DENY

- Deny; Kullanıcıya veritabanı veya nesnelere üzerinde çeşitli izinleri kısıtlamak için kullanılır.

- **DENY** izinler
- **TO** kullanıcılar

VeriTabanı için	Tablo için	Görünüm için
CREATE DATABASE	DELETE	DELETE
BACKUP DATABASE	INSERT	INSERT
CREATE TABLE	SELECT	SELECT
CREATE VIEW	UPDATE	UPDATE

DENY DELETE, INSERT, SELECT, UPDATE

TO dersKull

Veri Kontrol Dili - REVOKE

- Revoke; Kullanıcıya veritabanı veya nesnelere üzerinde verilen izinleri ve kısıtlamalara kaldırmak için kullanılır.

– REVOKE izinler

– TO kullanıcılar

VeriTabanı için	Tablo için	Görünüm için
CREATE DATABASE	DELETE	DELETE
BACKUP DATABASE	INSERT	INSERT
CREATE TABLE	SELECT	SELECT
CREATE VIEW	UPDATE	UPDATE

REVOKE DELETE, INSERT, SELECT, UPDATE

TO dersKull