Veritabanı genellikle bir Veritabanı Yönetim Sistemi DBMS (DataBaseManagementSystem) ile kontrol edilir

Çoğu veritabanında veri yazma ve sorgulama için yapılandırılmış sorgu dili SQL (Structured Query Language) kullanılır.

DATABASE'IN FAYDALARI NELERDIR

- 1) Yuksek miktarda bilgi depolanabili
- lusturma, Okuma, Degistirme ve Silme kolayligi Create, Read, Update, Delete (CRUD)
- 3) Girisin kolay ve kontrollu olmasi
- 4) Dataya ulasim kolayligi
- 5) Guvenlik
- API Application Programming Interface, bir uygulamaya ait yeteneklerin, başka bir uygulamada da kullanılabilmesi için, yeteneklerini paylaşan uygulamanın sağladığı arayüzdür.

Relational Databases, SQL Databases (Structured Query Language) olarak da adlandirilir

--> PRIMARY KEY -->

Bir veri tablosunda yer alan her satir icin identify (tanimlayici) gorevi gorur. Bir contraints'dir(kisitlama). ve essizdir ayni zamanda null degeri kabul etmez. Her database'de primary key olmak zorunda degildir. Ancak toplam tablolar dusunuldugunde en az bir tane primary key olmak zorundadir. Tek bir bir field ile olusturulamiyorsa birden fazla fieldin birlestirilmesi ile olusturulabilir.

• PRIMARY KEY OZELLIKLERI

- Primary Key bos gecilemez ve NULL degeri alamaz.
 DBMS'de mutlaka primary key olmalidir.
- Relational veri tabanlarında (relational database management system) mutlaka birincil anahtar olmalıdır.
- -Bir tabloda en fazla bir tane primary key olabilir.
- Not : Primary Key benzersiz (Unique) olmalidir ama her unique data Primary Key değildir
- Not : Primary key her turlu datayi icerebilir. Sayi, String..
- Not : Her tabloda Primary Key olmasi zorunlu değildir

Primary Key cesitleri --> Natural Key (TC, vergi no ...),

 kayit eklenmeden once uretilen sira numarasi gibi olanlara Surrogate key

CONSTRAINT Tablo1 pk PRIMARY KEY (id,isim));

birkac field in birlestirilmesi ile oluşturulan

```
composit PK

CREATE TABLE Tablo1
(
id char(10) UNIQUE,
isim varchar2(20),
brans varchar2(20),
cinsiyet varchar2(20)
```

FOREIGN KEY

- iki tablo arasinda Relation olusturmak icin kullanilir.
- Foreign Key baska bir tablonun Primary Key'ine baglidir.
- Referenced table (baglanilan tablo, Primary Key'in oldugu Tablo) parent table olarak
- adlandirilir. Foreign Key'in oldugu tablo ise child table olarak adlandirilir.
- Bir Tabloda birden fazla Foreign Key olabilir
- Toreign Key NULL degeri alabilir
- Note 1: "Parent Table" olmayan bir id'ye sahip datayi "Child Table"'a ekleyemezsiniz
- Note 2: Child Table'i silmeden Parent Table'i silemezsiniz.
 Once "Child Table" silinir, sonra
- "Parent Table" silinir.

```
CREATE TABLE urunler (
tedarikci_id char(10),
product_id char(10),

CONSTRAINT urunler_fk FOREIGN KEY (Tablo1_id)
REFERENCES Tablo1 (Tablo1_id) );
```

CHECK ile bir alana girilebilecek değerleri sınırlayabiliri

```
CREATE TABLE sehirler2 (
alan_kodu number(3) PRIMARY KEY,
isim varchar2(20) NOT NULL,
nufus number(8) CHECK (nufus>0) )
```

SELECT *

FROM students; // Tablodaki tum datalari getirir

SELECT *

FROM students;

WHERE yazili_not >80; // Tablodaki yazili_notu 80'den buyuk olan kayitlari getirir

- Ihtiyac olursa icice sorgu yazilabilir. Sorgunun WHERE veya SET kisminda baska bir sorgu ile elde edilen data kullaniliyorsa buna SUBQUERY denir.
- " DELETE FROM ogrenciler
- WHERE isim = 'Ali Can' OR veli_isim='Ayse' //kodu isim olarak Ali Can veya veli_isimolarak Ayse girilen kaydi (record) siler.

"TRUNCATE" kodu kullanılarak bir tablo silinirse dataların geri getirilme ihtimali olmaz

"Truncate" kodu geri getirilmesini (rolling back) istemeyeceginiz tablolari silmek icin kullanilir.

TRUNCATE TABLE customers DELETE FROM customers;

 DELETE FROM ile sildigimiz kayitlari geri getirebiliriz ama TRUNCATE ile silinen kayitlar geri getirilemez

SQL K

4. Veri Kontrol Dili (Data Control Language - DCL) veritabanı ve tablolar için yetki verilir veya geri alınır

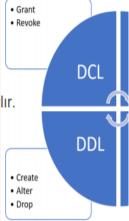
GRANT: Bir kullanıcıya yetki

vermek için kullanılır.

REVOKE: Bir kullanıcıya verilen

yetkiyi geri almak için kullanılır.

Veri Tanimlama Dili
 (Data Definition Language - DDL)
 veritabanı ve tabloları oluşturma,
 değiştirme ve silme işlemleri yapılır



CREATE: Bir veritabanı veya tablo oluşturur.
ALTER: Bir veritabanı veya tabloyu günceller.
DROP: Bir veritabanını veya tabloyu siler.