

# Key (Anahtar)

SQL'deki key (anahtar) kavramı, veritabanı tablolarında veri bütünlüğünü sağlamak, kayıtları benzersiz şekilde tanımlamak ve tablolar arasında ilişkiler kurmak amacıyla kullanılır

## Foreign Key (Yabancı Anahtar)

- Bir tabloyu, başka bir tablodaki birincil anahtar ile ilişkilendirmek için kullanılır. Bu sayede ilişkisel veritabanında tablolar arasında bağlantı kurulur.
- Foreign Key, anahtarın bulunduğu tabloya veri bütünlüğünü sağlar, örneğin, yabancı anahtar sütununa yalnızca ana tabloda var olan değerler eklenebilir.

## Unique Key (Benzersiz Anahtar)

- Bir sütundaki tüm değerlerin benzersiz olmasını sağlar. Bir tabloya birden fazla benzersiz anahtar atanabilir.
- NULL değer kabul edebilir, fakat birden fazla NULL değeri bulunamaz.

## Primary Key (Birincil Anahtar)

- Bir tablodaki her kayıt benzersiz olarak tanımlayan sütun veya sütun kombinasyonudur.
- Her tabloda yalnızca bir tane olabilir ve NULL değer kabul etmez.



# ÖRNEK

Örnek:

sql

Kodu kopyala

```
CREATE TABLE Students (  
    student_id INT PRIMARY KEY, -- Primary Key  
    first_name VARCHAR(50),  
    last_name VARCHAR(50),  
    birth_date DATE  
);
```

- Bu örnekte, **student\_id** sütunu her öğrenci için benzersiz bir kimlik olacaktır. İki farklı öğrenci aynı kimliğe sahip olamaz ve boş (NULL) bırakılamaz.

Örnek:

sql

Kodu kopyala

```
CREATE TABLE Employees (  
    employee_id INT PRIMARY KEY,  
    email VARCHAR(100) UNIQUE, -- Unique Key  
    phone_number VARCHAR(20)  
);
```

- Bu örnekte, **email** sütunundaki her bir çalışan için benzersiz bir e-posta adresi olması zorunludur. Aynı e-posta adresi başka bir çalışanda olamaz.

Örnek:

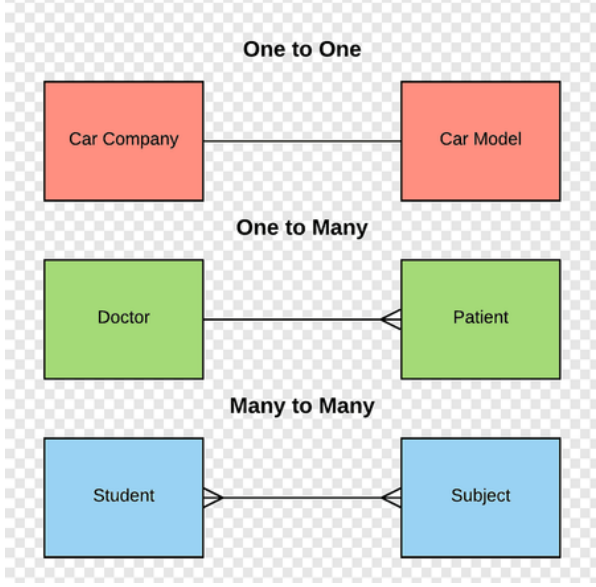
sql

Kodu kopyala

```
CREATE TABLE Orders (  
    order_id INT PRIMARY KEY,  
    order_date DATE,  
    customer_id INT,  
    FOREIGN KEY (customer_id) REFERENCES Customers(customer_id) -- Foreign Key  
);
```

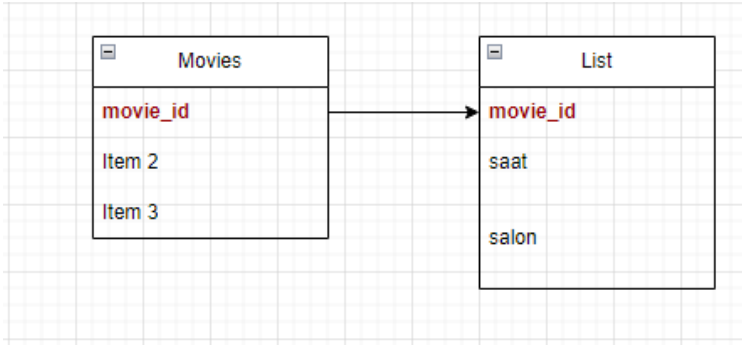
- Bu örnekte, **Orders** tablosundaki **customer\_id** sütunu, **Customers** tablosundaki **customer\_id** sütununa referans verir. Yani, sipariş verilen müşterinin veritabanındaki bir müşteri olması zorunludur.

# VERİTABANI İLİŞKİ TİPLERİ



**Veritabanı ilişki tipleri, tablolar arasındaki verilerin nasıl organize edildiğini ve birbirleriyle nasıl etkileşime geçtiğini gösterir.**

**Veritabanı tasarımında doğru ilişki tipini seçmek, verilerin bütünlüğünü ve performansını korumak için önemlidir. Temel olarak üç ana ilişki türü bulunur:**



İki tablo arasındaki ilişkiyi sağlamak için kullanacağımız alanların ikisinde kendi tablolarında birincil anahtar (primary key) ise birebir ilişki olur.

İki tablo arasındaki ilişkiyi sağlamak için kullanacağımız alanlardan birisi birincil anahtar (primary key) diğeri yabancı anahtar (foreign key) ise bire çok ilişki olur.



Çoktan çoğa (many-to-many) ilişki, ilişkisel veritabanı tasarımında sıkça karşılaşılan ve önemli bir kavramdır. Bu ilişki türü, bir tablodaki bir kaydın başka bir tabloda birden fazla kayıtlarla ilişkilendirilebilmesi ve aynı şekilde, bu ikinci tablodaki bir kaydın da ilk tabloda birden fazla kayıtlarla ilişkilendirilebilmesi durumunu ifade eder.

