

VERİ TABANI YÖNETİM SİSTEMLERİ LAB.

ÖN ÇALIŞMA: VARLIK-İLİŞKİ MODELİ

AMAÇ: Varlık-İlişki diyagramı verilen modelin tablolarını oluşturmak için SQL kodları yazınız.

ÖRNEK SQL KODLARI:

- **CREATE TABLE** ifadesi, bir veri tabanında yeni bir tablo oluşturmak için kullanılır.

```
CREATE TABLE table_name (  
    column1 datatype,  
    column2 datatype,  
    column3 datatype,  
    ....  
    PRIMARY KEY (columnID),  
    CONSTRAINT constraint_name FOREIGN KEY (columnID)  
    REFERENCES table_name(columnID)  
);
```

column parametreleri, tablonun sütunlarının isimlerini belirtir.

datatype parametresi, sütunun kaydedebileceği veri türünü belirtir (Örneğin; varchar, integer, date vb.).

Not: Kullanılabilir veri türleri hakkında genel bir bakış için **Veri Türleri** bağlantısına gidin.

- **PRIMARY KEY** kısıtı, bir tablodaki her kaydı benzersiz bir şekilde tanımlar. **PRIMARY KEY**'ler *UNIQUE* değerler içermelidir ve *NULL* değerler ıçeremezler. **PRIMARY KEY**, tek bir sütun veya birden fazla sütundan oluşabilir.

```
ALTER TABLE table_name ADD PRIMARY KEY (columnID);
```

```
ALTER TABLE table_name ADD CONSTRAINT constraint_name PRIMARY KEY (columnID);
```

```
ALTER TABLE table_name DROP CONSTRAINT constraint_name;
```

- **FOREIGN KEY** kısıtı, tablolar arasındaki bağlantıları bozabilecek işlemleri engellemek için kullanılır. **FOREIGN KEY**, bir tablodaki (veya birden fazla sütunu içeren) bir sütundur ve başka bir tablodaki **PRIMARY KEY**'e işaret eder.

ALTER TABLE *table_name* **ADD FOREIGN KEY** (*columnID*) **REFERENCES** *table_name*(*columnID*);

ALTER TABLE *table_name* **ADD CONSTRAINT** *constraint_name* **FOREIGN KEY** (*columnID*) **REFERENCES** *table_name*(*columnID*);

ALTER TABLE *table_name* **DROP CONSTRAINT** *constraint_name*;