

1906003022015

Veritabanı Yönetim Sistemleri

BAİBÜ Bilgisayar Müh.

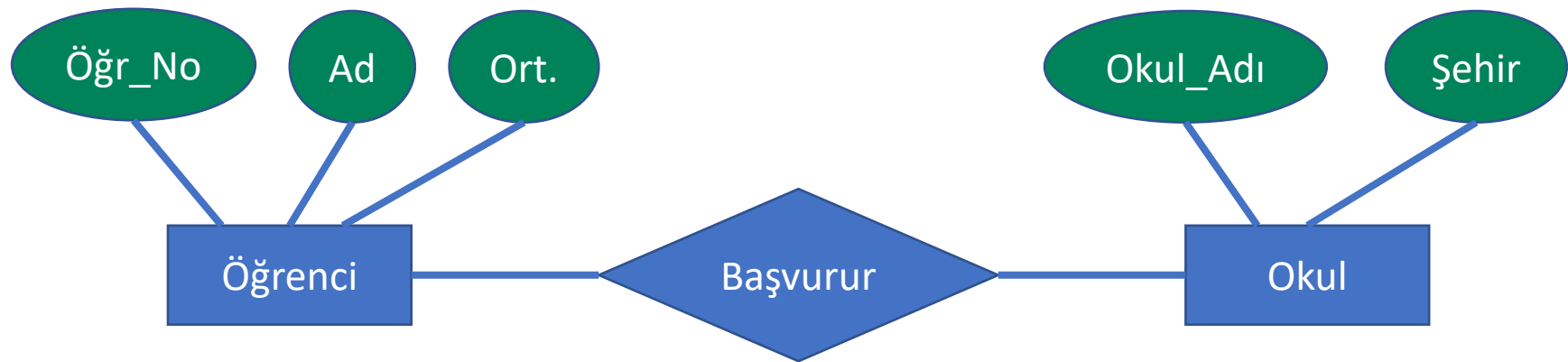
Dr. Öğr. Üyesi İsmail Hakkı Parlak

ismail.parlak@ibu.edu.tr

Oda: 329

Veri Modelleme

- Varlık İlişkisi (Entity Relationship - E/R) Modeli



- UML'in Veri Modelleme Alt Kümesi

UML

- Birleşik modelleme dili (UML), bir sistemin tasarımını görselleştirmek için standart bir yol sağlamayı amaçlayan genel amaçlı bir modelleme dilidir.
- UML diyagramları, bir sistem modelinin iki farklı görünümünü temsil eder:
 - Statik (veya yapısal) görünüm: nesneleri, nitelikleri, işlemleri ve ilişkileri kullanarak sistemin statik yapısını vurgular. *Sınıf diyagramlarını ve bileşik yapı diyagramlarını* içerir.
 - Dinamik (veya davranışsal) görünüm: nesneler arasındaki işbirliğini ve nesnelerin iç durumlarındaki değişiklikleri göstererek sistemin dinamik davranışını vurgular. Bu görünüm *sıralama diyagramlarını, aktivite diyagramlarını ve durum makinesi diyagramlarını* içerir.

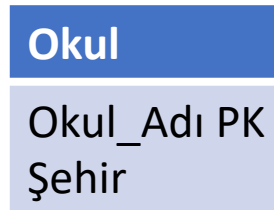
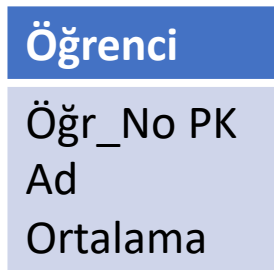
UML Veri Modelleme

Kavramlar:

1. Sınıflar (Classes)
2. İlişkiler (Associations)
3. İlişki Sınıfları (Association Classes)
4. Alt Sınıflar (Subclasses)
5. Bileşim ve Yığılımlar (Composition & Aggregation)

UML Veri Modelleme - Sınıflar

Ad ve nitelikler



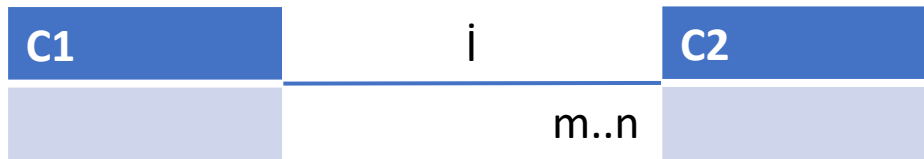
UML Veri Modelleme - İlişkiler

2 sınıfa ait nesneler arasındaki ilişkiler



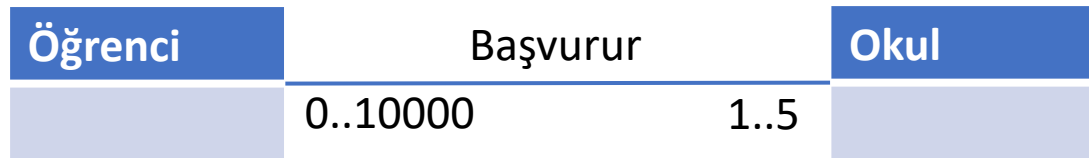
UML Veri Modelleme - İlişkiler (Çokluk)

C1 sınıfındaki nesneler, en az m en çok n tane C2 sınıfındaki nesne ile ilişkilidir.



UML Veri Modelleme - İlişkiler (Çokluk)

Öğrenciler başvuru yapmak zorundadır ancak 5'ten fazla okula başvuramazlar. Bir okul 10000'den fazla başvuru alamaz.



UML Veri Modelleme - İlişkiler (Çokluk)

İlişki çeşitleri:

- **Bire bir**
- Çoktan bire
- Çoktan çoğa
- Tam



UML Veri Modelleme - İlişkiler (Çokluk)

İlişki çeşitleri:

- Bire bir
- **Çoktan bire**
- Çoktan çoğa
- Tam



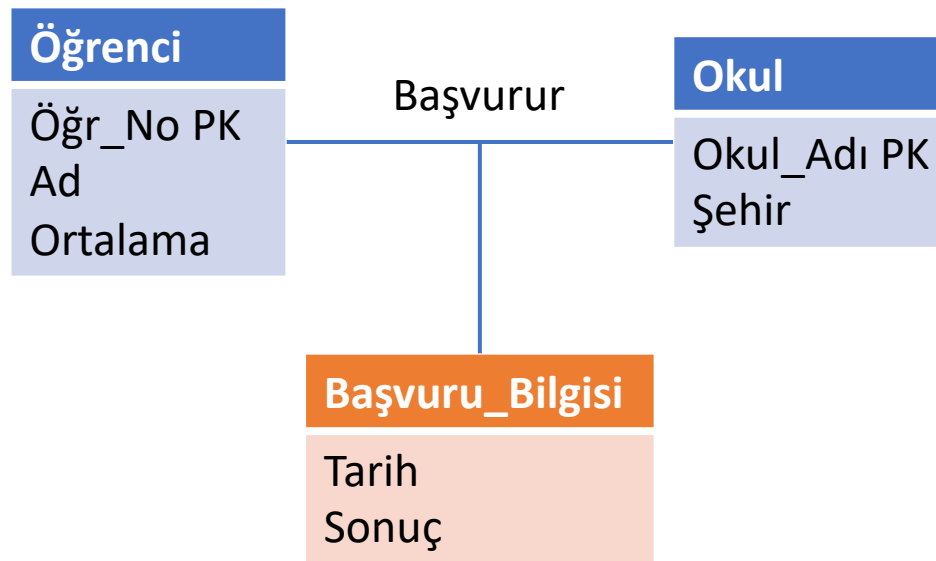
UML Veri Modelleme - İlişkiler (Çokluk)

İlişki çeşitleri:

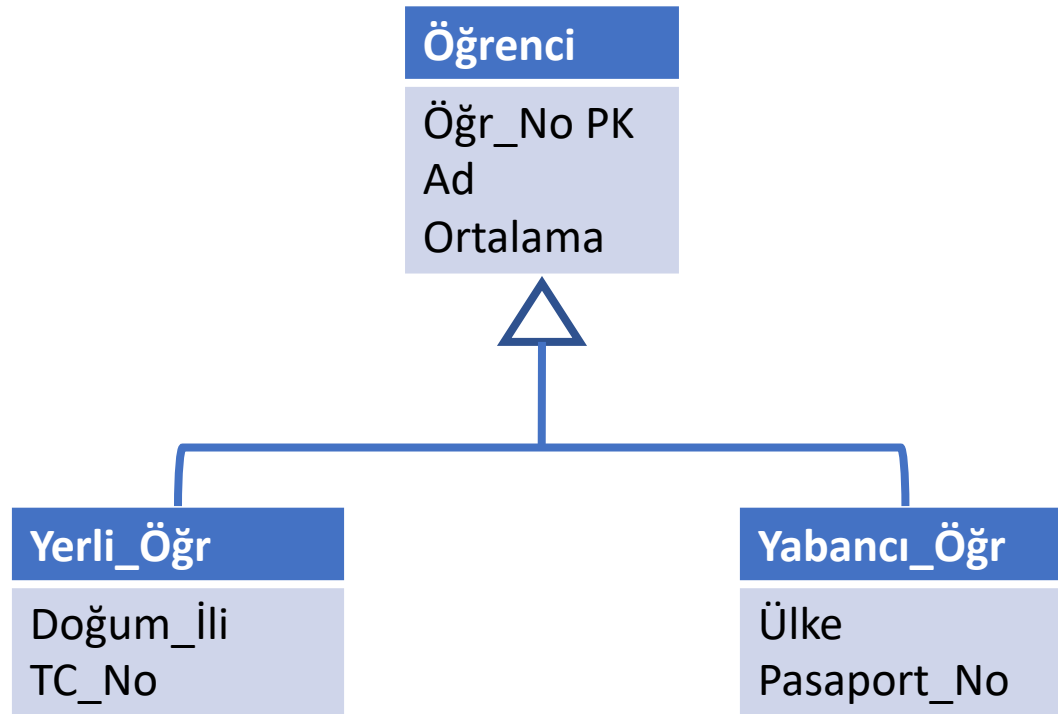
- Bire bir
- Çoktan bire
- Çoktan çoğa
- **Tam:** Her nesne, sınıflar arası ilişkiye dahil olmalıdır.
 - Tam bire bir
 - Tam çoktan bire
 - Tam çoktan çoğa

UML Veri Modelleme - İlişki Sınıfları

2 sınıfa ait nesneler arasındaki ilişkiler ve bu ilişkilere ait nitelikler

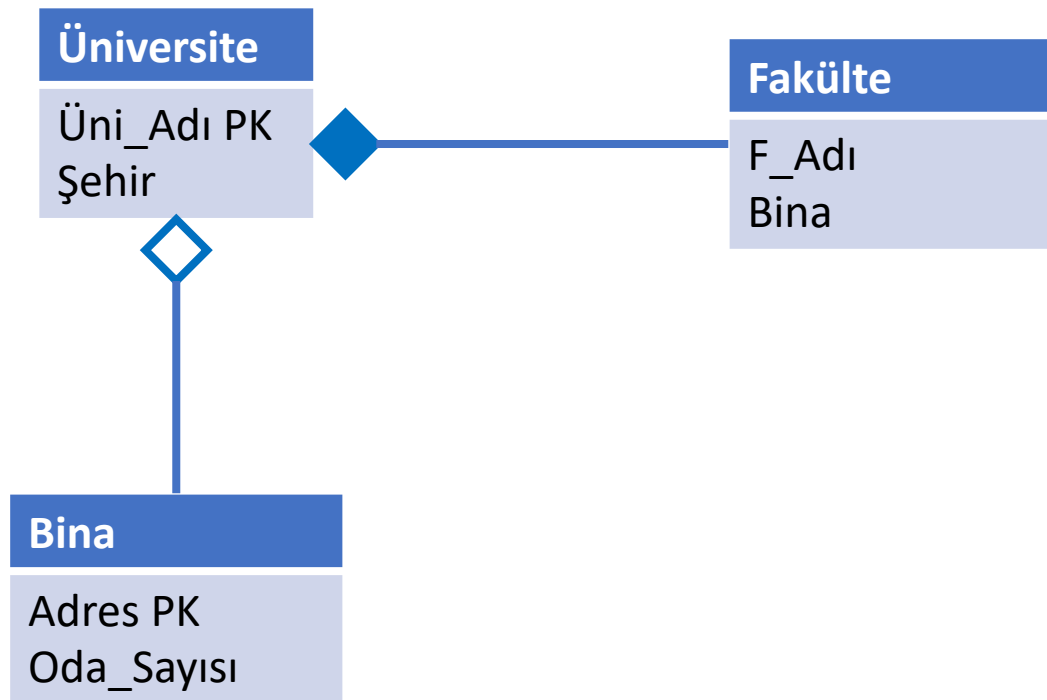


UML Veri Modelleme - Alt Sınıflar



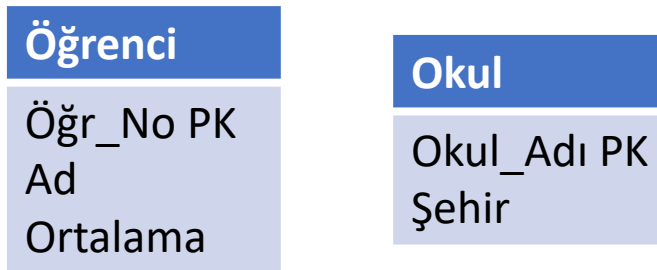
UML Veri Modelleme - Bileşim ve Yığışım

Bir sınıfın nesneleri diğer bir sınıfa aittir.



UML'den DB'ye

Sınıflar



Öğrenci(Öğr_No, Ad, Ortalama)

Okul(Okul_Adı, Şehir)

UML'den DB'ye

İlişkiler (Associations)



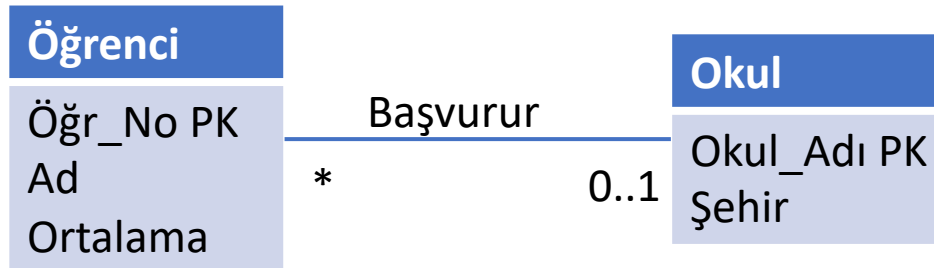
Öğrenci(Öğr_No, Ad, Ortalama)

Okul(Okul_Adı, Şehir)

Başvurur(Öğr_No, Okul_Adı)

UML'den DB'ye

İlişkiler (Associations) için anahtarlar çokluk bilgisine bağlıdır.



Öğrenci(Öğr_No, Ad, Ortalama)

Okul(Okul_Adı, Şehir)

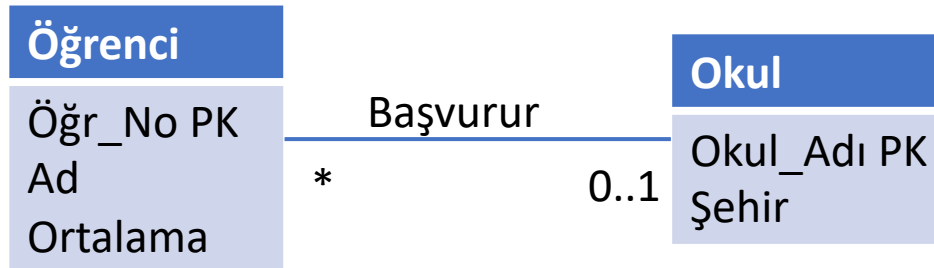
Başvurur(Öğr_No, Okul_Adı)

Bir tarafta 0..1 veya 1..1 varsa ilişki anahtarını diğer taraftan alır.
Başvurur tablosu kesinlikle gerekli midir?

UML'den DB'ye

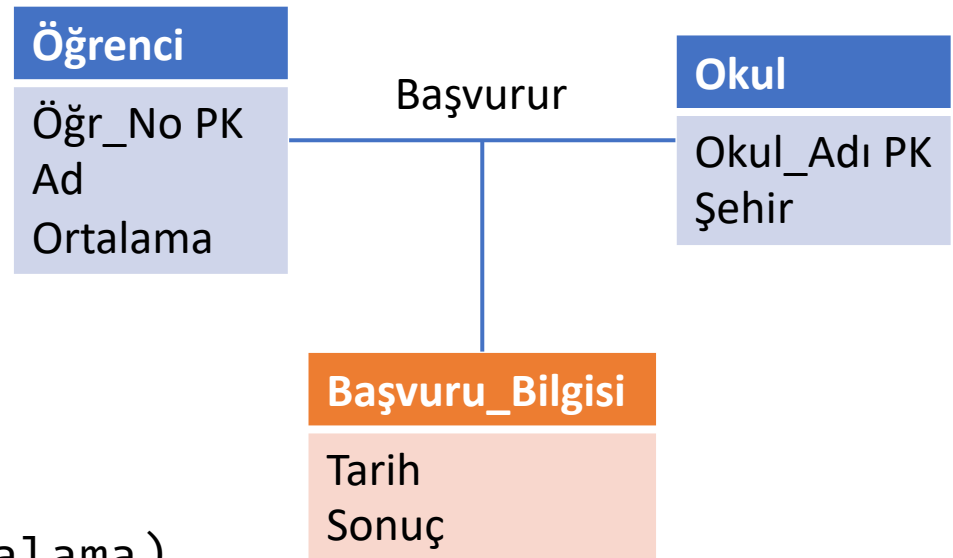
İlişkiler (Associations) için anahtarlar çokluk bilgisine bağlıdır.

Başvurur tablosu kesinlikle gerekli midir?



UML'den DB'ye

İlişki sınıfı tablosuna kendine ait nitelikler eklenir.



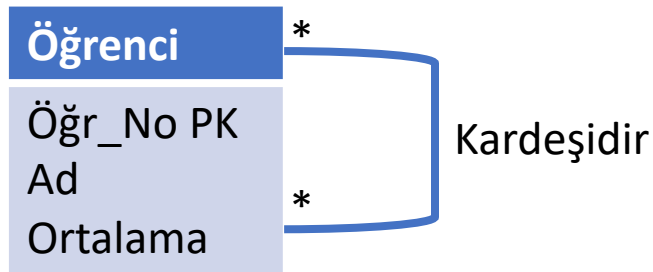
Öğrenci(Öğr_No, Ad, Ortalama)

Okul(Okul_Adı, Şehir)

Başvurur(Öğr_No, Okul_Adı, Tarih, Sonuç)

UML'den DB'ye

Kendisiyle ilişki



`Öğrenci(Öğr_No, Ad, Ortalama)`

`Kardeşidir(Öğr_No_1, Öğr_No_2)`

UML'den DB'ye

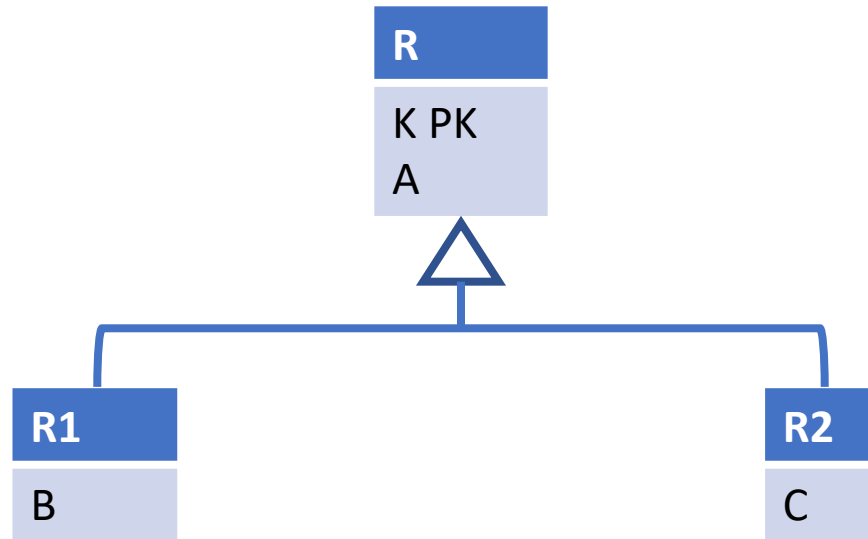
Alt sınıflar için stratejiler:

1. Alt sınıf tabloları üst sınıfın anahtarını ve kendi özel niteliklerini barındırabilir.
2. Alt sınıf tabloları bütün nitelikleri barındırabilirler.
3. Tüm alt sınıf ve üst sınıf niteliklerini barındıran dev bir tablo oluşturulabilir.

En iyi strateji duruma göre belirlenir.

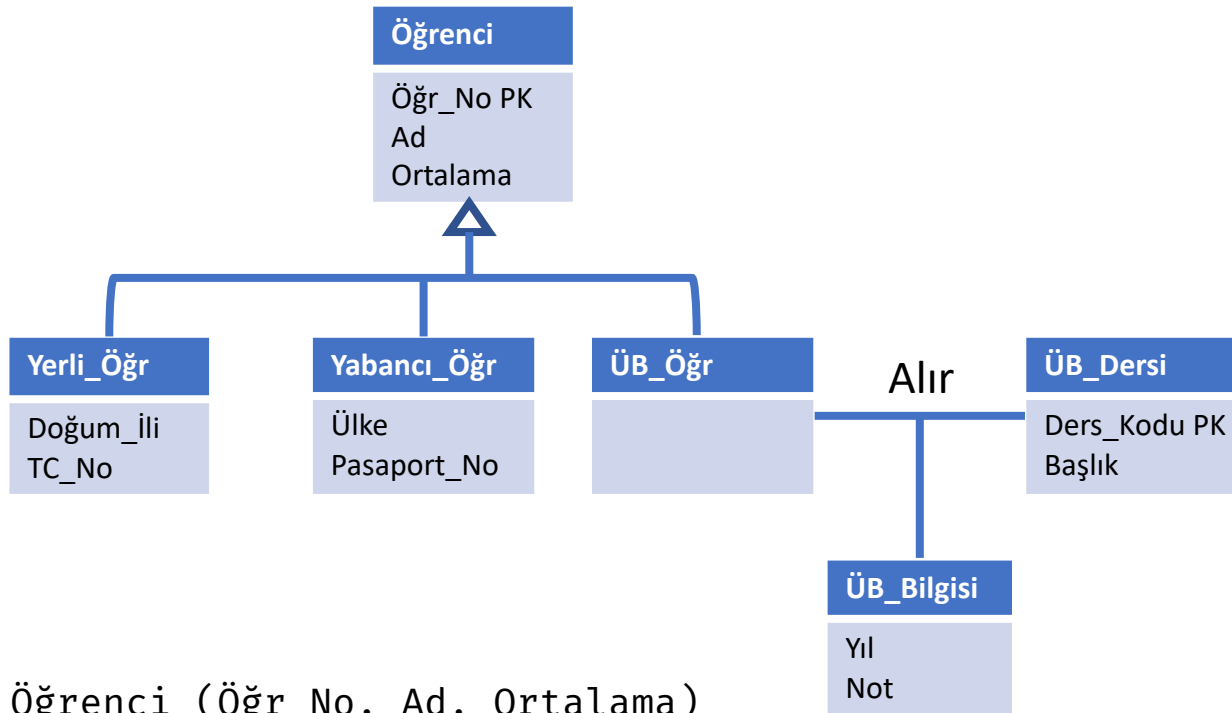
UML'den DB'ye

Alt Sınıflar



1. $R(\underline{K}, A)$ $R1(\underline{K}, B)$ $R2(\underline{K}, C)$ - Alt sınıflar örtüşmüyorsa ve üst sınıfta bulunan ancak alt sınıfta bulunmayan kayıtlar varsa.
2. $R(\underline{K}, A)$ $R1(\underline{K}, A, B)$ $R2(\underline{K}, A, C)$ - Alt sınıflar örtüşmüyorsa ve üst sınıfta bulunup da alt sınıfta bulunmayan bir kayıt yoksa (R silinebilir).
3. $R(\underline{K}, A, B, C)$ - Alt sınıflar çokça örtüşüyorsa.

UML'den DB'ye (Örnek)



Öğrenci (Öğr_No, Ad, Ortalama)

Yerli_Öğr (Öğr_No, Doğum_İli, TC_No)

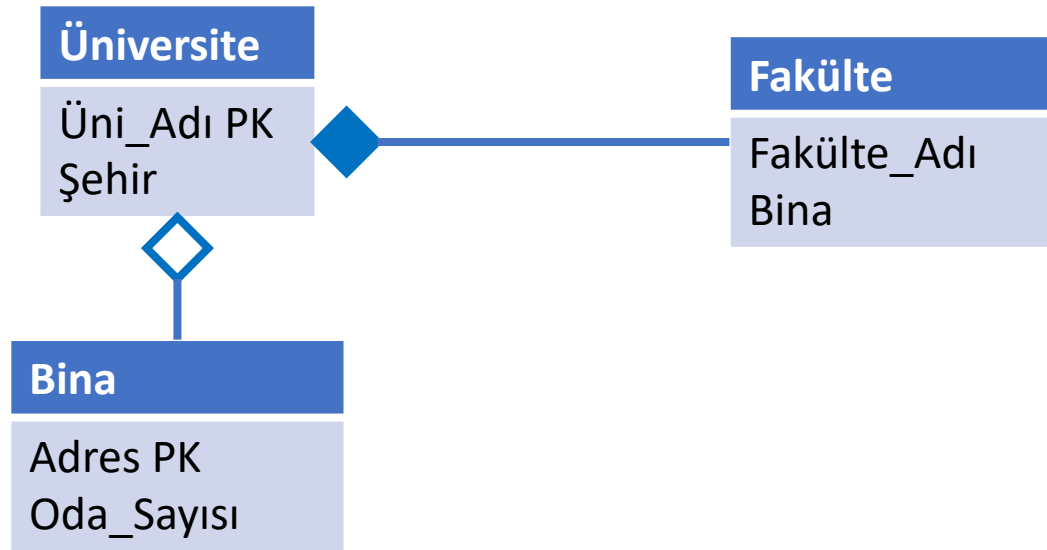
Yabancı_Öğr (Öğr_No, Ülke, Pasaport_No)

ÜB_Öğr (Öğr_No)

ÜB_Dersi (Ders_Kodu, Başlık)

Alır (Öğr_No, Ders_Kodu, Yıl, Not)

UML'den DB'ye (Bileşim ve Yığılım)



Üniversite (Üni_Adı, Şehir)

Fakülte (Fakülte_Adı, Bina, Üni_Adı)

Bina (Adres, Oda_Sayısı, Üni_Adı)