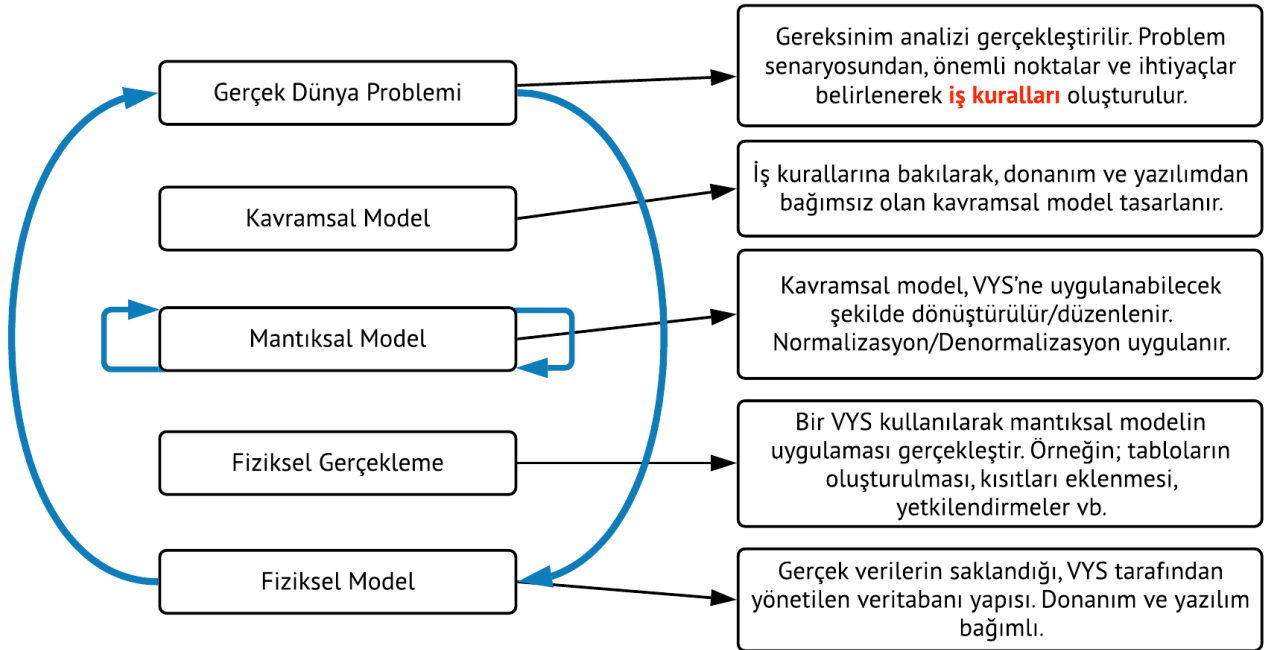


İlişkisel Model

Konular

- Veritabanı Tasarım Aşamaları
- Temel Kavramlar
- Tablo
- Anahtar
- Bütünlük Kuralları
- Birincil Anahtar Özellikleri
- Vekil Birincil Anahtar
- Varlık Bağlantı Modelinin İlişkisel Modele Dönüştürülmesi
- İndeks
- Veri Sözlüğü (Sistem Kataloğu)
- Örnek Uygulama - Araç Üreticisi
- Örnek Uygulama – Elektronik Ticaret Sistemi
- Kaynaklar

Veritabanı Geliştirme Yaşam Döngüsü



Temel Kavramlar

- **Varlık (Entity)**: Varolan ve benzerlerinden ayırt edilebilen herşey.
- **Varlık Kümesi (Entity set)**: Benzer varlıkların oluşturduğu kümeye denir. Varlık kümelerinin ayrık kümeler olması gerekmez. Bunlar iç içe, kesişen ya da ayrık kümeler olabilir. Örnek olarak, Öğrenciler, Kız öğrenciler, Yurt öğrencileri, Açılan Dersler, Doktorlar vb. sayabiliriz.

- **Nitelik (Attribute):** Bir varlık kümesindeki nesnelerin özelliklerini göstermek ve varlıkları birbirinden ayırt etmek için kullanılır. Tablo (table) içerisindeki "sütunlara (column)" karşılık gelir. "Alan" ismi de verilir.
- **Değer Alanı (Domain):** Bir niteliğin alabileceği değerler uzayına ilgili niteliğin değer alanı denir.
- **İlişki (Relation):** Satır ve sütunlardan oluşan bir tablo.
 - İlişki = Tablo
- **Kayıt (Tuple):** Bir tablodaki kayıt veya satırlar.
 - Bir ilişki (tablo), satırlar ya da kayıtlar kümesi olarak düşünülebilir (tüm satırlar farklıdır).
- **Sütun Sayısı (Degree):** İlişkideki (tablodaki) sütun sayısı.
- **Satır Sayısı (Cardinality):** İlişkideki (tablodaki) satır (kayıt) sayısı.
- **Birincil Anahtar (Primary Key):** Tek tanımlayıcı.

<u>acilanDersNo</u>	dersNo	donem	ogretimUyesi	sube
1	T001	17G	21	A
2	T002	18G	56	A
3	C001	17B	75	A
4	T001	18G	67	B
5	C001	18B	75	B

- **İlişkisel Veritabanı (Relational Database):** İlişkiler (tablolar) kümesi. Satır ve sütunlardan meydana gelen tablolar topluluğu.
- **İlişkisel Şema (Relational Schema):** İlişkinin (tablonun) adını ve ek olarak her bir sütunun adını ve tipini belirtir. Metin ile ifade edilebileceği gibi diyagram kullanılarak da gösterilebilir.
 - Örneğin Öğrenciler isminde bir ilişkinin (tablonun) ilişkisel şeması (metinsel olarak) aşağıda verilmiştir.
 - Öğrenciler(**ogrenciNo: string**, adi: string, soyadi: string, yas: integer, ortalama: real)

Tablo

Tablo Karakteristikleri

- Bir tablo (table), satır (row) ve sütunlardan (column) oluşan iki boyutlu bir yapı olarak algılanır.
- Her tablo satırı (kayıt/tuple), varlık kümesi (entity set) içerisindeki tek bir varlık (entity) oluşumunu temsil eder.
- Her tablo sütunu bir niteliği (attribute) temsil eder ve her bir sütun farklı bir isme sahiptir.
- Her satır-sütun kesişimi tek bir veri değerini temsil eder.
- Bir sütun içerisinde yer alan tüm değerler aynı veri tipine sahip olmalıdır.
- Her sütun, nitelik değer alanı (attribute domain) olarak bilinen kesin bir değerler aralığına sahiptir.
- Satır ve sütunların sırası Veritabanı Yönetim Sistemi (VTYS/DBMS) tarafından bakıldığında önemsizdir.
- Her tablo, her bir satırını eşsiz bir şekilde tanımlayan bir niteliğe veya nitelik birleşimine sahip olmak zorundadır. Buna birincil anahtar ismi verilir.

Oğrenciler																	
ogrenciNo TC Kimlik No	adi	soyadi	sifre md5 formatında saklanıyor...	telefonNo	eposta	babaAdi	adres	dogumTarihi	il	ilce	kayitTarihi	ogrenimDurumu	aciklama	cinsiyet E,K, G(Girilmemiş)			
0000000001	Ayşe	Demirr	a7f4e18520f1a28fb9b1edb53f9fd6b6		ad@a.com	Hasan	Bilinmiyor	0000-00-00	34	409	0000-00-00	NULL	Yok	K			
0000000003	Hasan	Çelik	hasancelik		hc@a.com	Hasan	Bilinmiyor	NULL	01	001	0000-00-00	NULL	aciklama	E			
0000000004	Tamer	Yorulmaz	e1e6205a7c630320a8f854df101905fb		ty@a.com	Yılmaz	Bilinmiyor	1975-05-01	01	008	2011-05-26	NULL	Yok	E			
0000000008	Ayşe	Eren	e78c265a4f809993ccb24c6ea5c308dc		aer@a.com	Mustafa	Konya	1994-06-07	42	560	2011-06-06	NULL		K			
0000000009	Ayşe	Yılmaz	9693bb4495eae586d84e2001f1d665ac		ay@a.com	Ahmet	Kocaeli	1999-05-01	41	533	2011-05-26	NULL	Bilinmiyor	K			
0000000021	Ayten	Gül	035e15c85c630a56ebfd9d44f7796da1	1234567892	Girilmemiş	Girilmemiş	Gebze	1993-09-01	00	940	2011-09-12	5	Bilinmiyor	K			
0000000041	Ayşe	Meteee	94b592cdf868a1e6c8d70d836b7aaed0		Girilmemiş	Girilmemiş	Bilinmiyor	1994-06-01	03	026	2011-06-06	NULL	Bilinmiyor	K			
0000000061	Ahmet	Meteee	9c285d744cc297a717c14a7f918deeb8	3456565656	Girilmemiş	Girilmemiş	Bilinmiyor	0000-00-00	00	940	0000-00-00	0	Bilinmiyor	E			
0000000062	Buse	Yılmaz	ae28fe45152c26c9ff2608fb7dfca64d	3456789123	Girilmemiş	Girilmemiş	Merkez	1996-09-02	00	940	2011-09-12	5	Bilinmiyor	K			
1000000001	Ayla	Mert	42bf6175cfa9cd1486653179569cb10c	5053032323	Girilmemiş	Girilmemiş	Bilinmiyor	0000-00-00	00	940	0000-00-00	0	Bilinmiyor	K			
1000000002	Aylin	Mert	2ebd77fd5463daa1f55620defa89c3c	5053032323	Girilmemiş	Girilmemiş	Bilinmiyor	0000-00-00	00	940	0000-00-00	0	Bilinmiyor	K			
1000000003	Aydin	Mert	b695848a22e162c5abb493d94f583c8	5053032323	Girilmemiş	Girilmemiş	Bilinmiyor	0000-00-00	00	940	0000-00-00	0	Bilinmiyor	E			
1000000004	Ayhan	Mert	8edc5485602b71d1a939b5239f719f74	5053032323	Girilmemiş	Girilmemiş	Bilinmiyor	0000-00-00	00	940	0000-00-00	0	Bilinmiyor	E			
1000000005	Ayhan	Metin	7c064883cf91819986026da9c4d420f0	5053032323	Girilmemiş	Girilmemiş	Bilinmiyor	0000-00-00	00	940	0000-00-00	0	Bilinmiyor	E			
1000000006	Ayhan	Mete	704a61e1660f5a4b65457f7b9d887e68	5053032323	Girilmemiş	Girilmemiş	Bilinmiyor	0000-00-00	00	940	0000-00-00	0	Bilinmiyor	E			
1000000007	Aydin	Mete	e12b2500edf1c14735863d074146da13	5053032323	Girilmemiş	Girilmemiş	Bilinmiyor	0000-00-00	00	940	0000-00-00	0	Bilinmiyor	E			
1000000008	Aydin	Aymaz	4089c90d768512aaa9d75f03cd572ed7	5053032323	Girilmemiş	Girilmemiş	Bilinmiyor	0000-00-00	00	940	0000-00-00	0	Bilinmiyor	E			

- Ogrenciler tablosu, 17 satır (kayıt) ve 15 sütundan (nitelik) oluşan bir iki boyutlu yapı olarak görülür.
- Ogrenciler tablosundaki her satır, varlık kümesindeki bir varlığı tanımlar (Varlık kümesi, Ogrenciler tablosu ile temsil edilir).
 - Örneğin 1. satır, Ayşe Demir ismindeki bir öğrenciyi tanımlar.
- Her sütun, bir niteliği temsil eder ve her sütunun ayrı bir ismi mevcuttur.
- Bir sütundaki tüm değerler, niteliğin karakteristikleri ile uyumludur.
 - Örneğin, dogumTarihi sütunu, tüm satırlar için doğum tarihi bilgilerini içerir.
- Veriler, biçimine ve fonksiyonuna göre sınıflandırılmalıdır. VTYS'lerin desteklediği temel veri tipleri:
 - Nümerik (Numeric): Üzerinde anlamlı aritmetik işlemler yapılabilen veri. Örneğin not ortalamasının saklandığı bir alan nümerik tiptedir.
 - Karakter (Character): Metin verisi, karakter dizisi. Aritmetik işlemler için tanımlanmamış olan karakter ve semboller. Örneğin adı, soyadı, telefon numarası vb. bilgilerin saklandığı alanlar.
 - Tarih, Zaman (Date, Time): Tarih, zaman verisi için kullanılır. Örneğin; doğum tarihi, sisteme son giriş zamanı vb. alanlar.
 - Mantıksal (Logical): Doğru ya da yanlış (evet veya hayır - boolean) değerleri.
- Bir sütun için izin verilen değerler aralığına, o sütunun değer alanı (domain) adı verilir.
 - Örneğin tablodaki öğrenci not bilgileri 0-4 aralığında sınırlandırılmış olduğundan ötürü, bu sütunun değer aralığı [0,4] olarak ifade edilir.
- ogrenciNo alanı birincil anahtardır.

Anahtar

Fonksiyonel Bağımlılık

- Anahtar:** Kayıtların ayırt edilebilmesini (süper, aday, birincil, ikincil ve vekil anahtarlar) ve/veya tablolar arasında ilişki kurulabilmesini (birincil, yabancı anahtarlar) sağlayan yapı.
- Anahtarlar belirlenirken **fonksiyonel bağımlılık** kavramının bilinmesi gerekir.
- Örnekte, ogrenciNo alanı kullanılarak öğrenci adı belirlenebilir. Tersisi doğru değildir.
- Bu durumda:
 - ogrenciNo alanı, ogrenciAdi alanını belirler.
 - ogrenciAdi alanı, ogrenciNo alanına **fonksiyonel bağımlıdır**.
 - ogrenciNo → ogrenciAdi şeklinde gösterilir.

Anahtar Çeşitleri

- **Süper Anahtar:** Herbir satırı eşsiz olarak tanımlayabilen anahtara süper anahtar denir. Bir ya da birden çok alanı içerebilir.
 - ogrenciNo
 - ogrenciNo, adi
 - ogrenciNo, adi, soyadi , ...
 - eposta
 - kimlikNo ...
- **Aday Anahtar:** Herbir satırı eşsiz olarak tanımlayabilen ve en az sayıda alana sahip olan süper anahtara aday anahtar denir.
 - "ogrenciNo, adi" ikilisi süper anahtardır. Sadece ogrenciNo alanı kayıtları eşsiz olarak tanımlayabildiğinden dolayı bu ikili aday anahtar değildir.
- **Birincil Anahtar (Primary Key, PK):** Herbir satırı eşsiz olarak tanımlayabilen ve aday anahtarlar içerisinde seçilen anahtar.
 - Birincil anahtar eşsiz değerlere sahip olmalıdır, NULL değerler alamaz.
 - ogrenciNo, kimlikNo, pasaportNo alanları aday anahtarlardır. Her biri birincil anahtar olarak seçilebilir (sadece biri seçilmelidir).
- **Vekil Birincil Anahtar:**
 - Uygulamada çoğunlukla vekil birincil anahtarlar kullanılır. Ayrıntılar "Vekil Birincil Anahtar" başlıklı bölümde verilmektedir.
- **İkincil/Alternatif Anahtar:** Birincil anahtar olarak seçilmeyen diğer aday anahtarlara ikincil/alternatif anahtar denir.
- **Yabancı Anahtar (Foreign Key, FK):** Bağıntı kurulan tablo içerisinde yer alan ve bağıntı kuran tablodaki birincil anahtarın değerlerine benzeyen değerler içeren alan.
 - İki tablo arasında bağıntı kurulurken bağıntı kuran tablonun birincil anahtarı ile bağıntı kurulan tablonun yabancı anahtarı kullanılır. Dolayısıyla iki anahtar da aynı tipte olmalıdır.
- Aşağıda yabancı anahtar örneği görülmektedir.

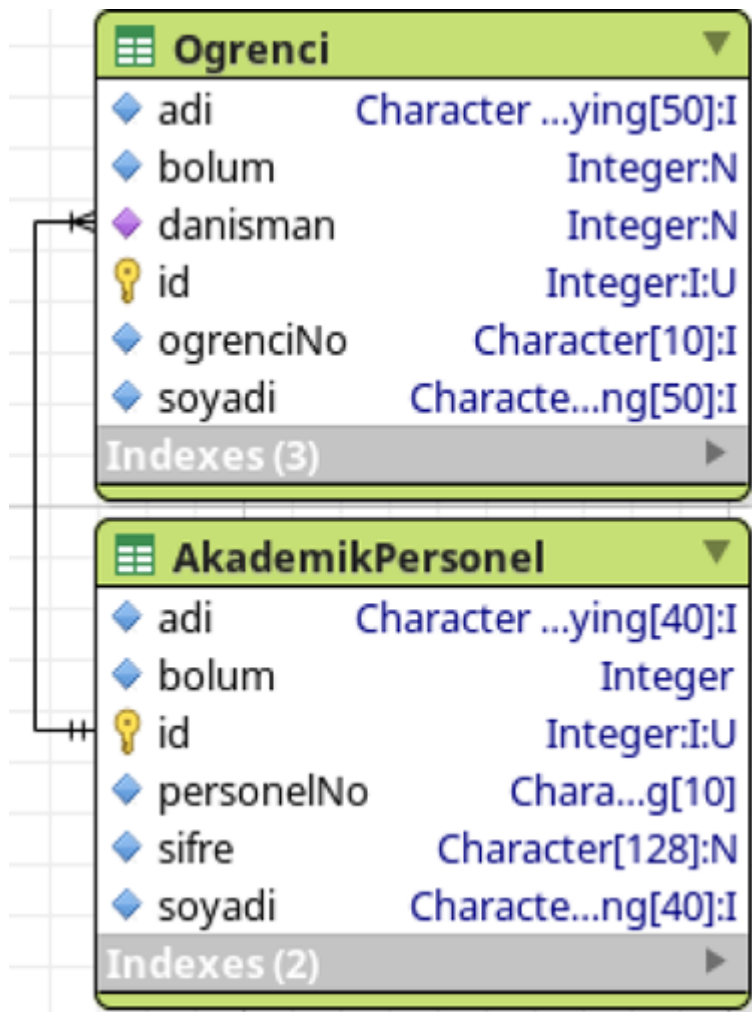
Database: **OgrenciBilgiSistemi** Schema: **public** Table: **Ogrenci**

	id	ogrenciNo	adi	soyadi	bolum	danisman
3	3	B151212322	Ayşe	Güler	1	4
4	4	B111212134	Ayşen	Şahin	1	3
5	5	B101212320	Ahmet	Mert	1	3
6	6	B101212322	Ayşe	Mert	1	3
7	7	B111212323	Mehmet	Aydın	1	3
8	8	B131212122	Buse	Yılmaz	2	3
9	9	B121212134	Ayşen	Aydın	1	3
10	10	B111212321	Berkay	Aydın	1	3
11	11	B131212123	Berat	Yılmaz	2	3
12	12	B121212135	Aylin	Aydın	1	4
13	13	B151212349	Mehmet	Gen.	<NULL>	<NULL>
14	14	B151212350	Ali	Katı	<NULL>	<NULL>
15	15	B151212351	Alican	Konyalı	<NULL>	<NULL>

15 from 43

Related Table: **AkademikPersonel : Insk_Ogrenci_AkademikPersonel (M:1)** Mode: show all Set: OR (unic

	id	personelNo	adi	soyadi	bolum	sifre
1	3	A1	Ahmet	Yılmaz	1	c4ca4238a0b923820dcc509a6f75849b
2	4	A2	Umut	Işık	1	c81e728d9d4c2f636f067f89cc14862c



Bütünlük Kuralları

- **Varlık Bütünlüğü (Entity Integrity):** Varlık bütünlüğünün sağlanabilmesi için her tabloda birincil anahtar (PK) olmalıdır.
 - Her varlığın "not null" kısıtını sağlayan eşsiz ("unique" kısıtı) bir tanımlayıcısı olması garanti edilir.
 - Varlık bütünlüğü kullanılarak, örneğin; aynı öğrenci numarasına sahip çok sayıda öğrencinin ya da öğrenci numarası olmayan öğrencinin bulunamaması garanti edilir.
- **Referans Bütünlüğü (Referential Integrity):** Referans bütünlüğünün sağlanabilmesi için yabancı anahtar(FK) tanımlanır.
 - Yabancı anahtar alanı ya NULL olabilir ya da bağıntı kuran tablodaki birincil anahtar alanı değerlerinden biri olabilir.
 - Her yabancı anahtar değerinin diğer tablonun birincil anahtarı değeriyle tutarlı olması garanti edilir.
 - Referans bütünlüğü kullanılarak, örneğin; öğrencinin açılmayan bir dersi alması, derse kayıtlı bir öğrencinin okuldan kaydının silinmesi, olmayan bir ürünün sipariş edilmesi vs. önlenir.

Database: OgrenciBilgiSistemi Schema: public Table: Ogrenci

		id	ogrenciNo	adi	soyadi	bolum	danisman
3		3	B151212322	Ayşe	Güler	1	4
4		4	B111212134	Ayşen	Şahin	1	3
5		5	B101212320	Ahmet	Mert	1	3
6		6	B101212322	Ayşe	Mert	1	3
7		7	B111212323	Mehmet	Aydın	1	3
8		8	B131212122	Buse	Yılmaz	2	3
9		9	B121212134	Ayşen	Aydın	1	3
10		10	B111212321	Berkay	Aydın	1	3
11		11	B131212123	Berat	Yılmaz	2	3
12		12	B121212135	Aylin	Aydın	1	4
13		13	B151212349	Mehmet	Gen.	<NULL>	<NULL>
14		14	B151212350	Ali	Katı	<NULL>	<NULL>
15		15	B151212351	Alican	Konyali	<NULL>	<NULL>

15 from 43

Related Table: AkademikPersonel : Ink_Ogrenci_AkademikPersonel (M:1) Mode: show all Set: OR (unic

			id	personelNo	adi	soyadi	bolum	sifre
1			3	A1	Ahmet	Yılmaz	1	c4ca4238a0b923820dcc509a6f75849b
2			4	A2	Umut	Işık	1	c81e728d9d4c2f636f067f89cc14862c

- **Referans Bütünlüğü Uygulama Notu**
 - Referans bütünlüğünün sağlanması için yabancı anahtar alanının NULL olabileceği de söylenmişti.
 - NULL değer sorgularda sorunlara neden olabileceğinden genellikle NULL değer yerine özel değerler tercih edilir.
 - Örneğin; il bilgisi olarak NULL yerine "00: Bilinmiyor" verilebilir. Bu durumda İller tablosunda 00, Bilinmiyor değerlerine sahip bir satır bulunmalıdır. Kisiler tablosundaki il alanının varsayılan değeri de 00 yapılabilir. Böylece, kişinin il bilgisi girilmediği zaman varsayılan olarak bilinmiyor bilgisi atanmış olur.
- **NOT NULL (Boş Olamaz):** Alan içerisine değer yazılmasını zorlamak istiyorsak NOT NULL yapabiliriz. Örneğin, kişinin adının zorunlu olarak girilmesini bu şekilde sağlayabiliriz.
- **UNIQUE (Eşsiz):** Alan içerisine yazılan değerlerin her satır için eşsiz olmasını istiyorsak kullanılır. Örneğin, kişilerin TC Kimlik numaralarının aynı yazılamamasını bu şekilde garanti etmiş oluruz.

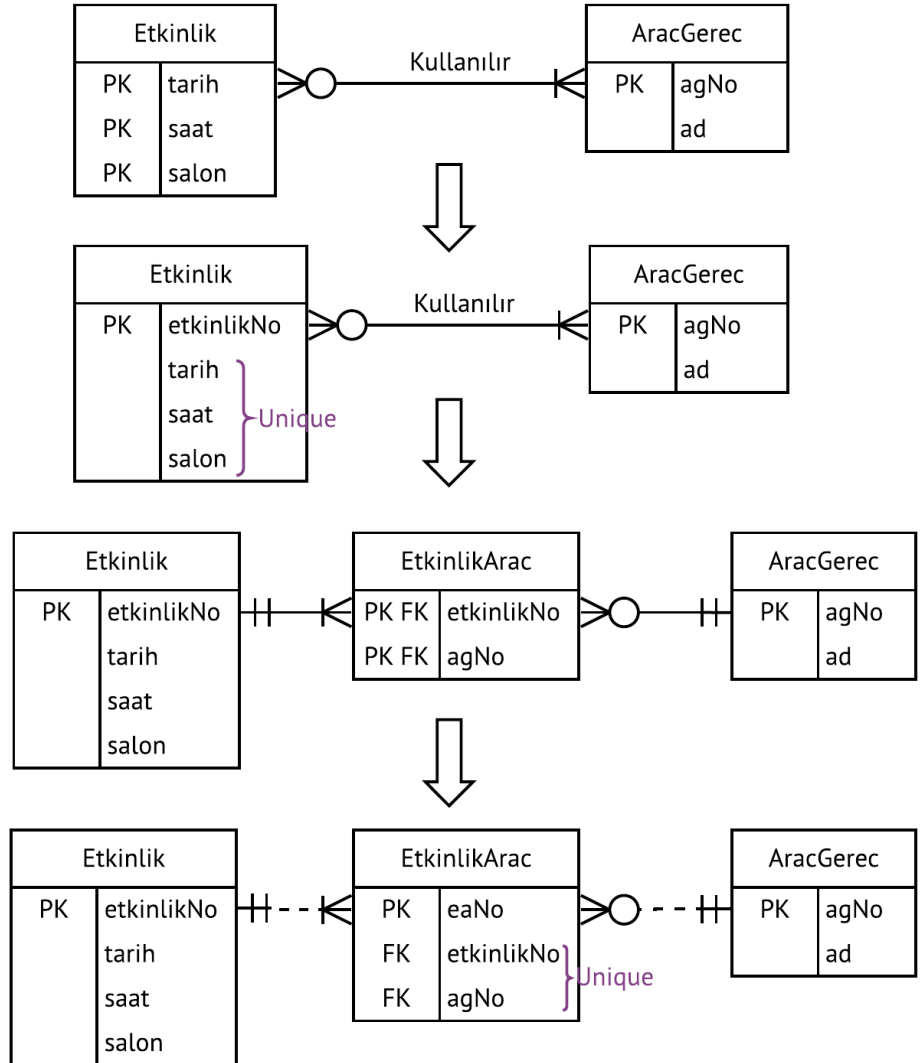
Birincil Anahtar Özellikleri

- UNIQUE (kayıtları eşsiz olarak tanımlayabilmesi) ve NOT NULL (boş olmaması) bütünlük kurallarını sağlamalı.
- Zamanla değişme ihtimali olan alanlar tercih edilmemeli.
- Mümkün olduğunca tek bir alandan oluşmalı.
 - Çok alandan oluşursa bağıntıları kurmak, sorguları yazmak karmaşıklaşır. Gecikme ve kaynak kullanımı artar.
- Sayısal alanlar tercih edilmeli.
 - İşlem hızı ve otomatik artma (auto increment) desteği sağlanabilir.
- Güvenlik riski bulunan alanlar tercih edilmemeli.
 - Yazılımlarda yoğun olarak birincil anahtar değeri kullanılır.

Vekil Birincil Anahtar

- Birincil anahtar fonksiyonlarını yerine getirmek için eklenen alandır.

tarih	saat	salon
1.1.2000	10.00	1
1.1.2000	11.00	1
2.1.2000	10.00	1
2.1.2000	11.00	1
2.1.2000	11.00	2



Varlık Bağntı Modelinin İlişkisel Modele Dönüştürülmesi

- Her varlık için bir tablo (ilişki/relation) oluşturulur.

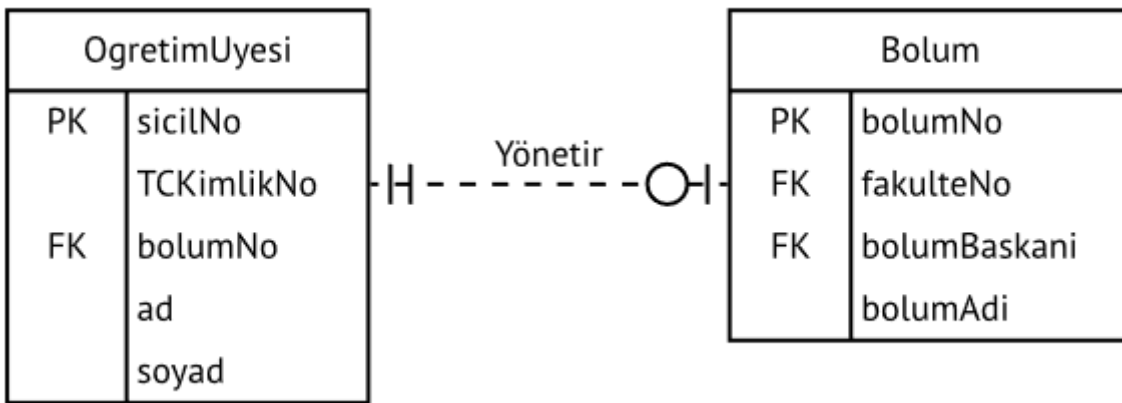
Ogrenci	
PK	ogrenciNo
FK	bolumNo
	ad
	soyad

- **İlişkisel Şema (Metinsel Gösterim)**

- Ogrenci(**ogrenciNo: String**, bolumNo: Integer, adi: String, soyadi: String)

Bir-Bir Bağıntısı

- 1 öğretim üyesi **en çok 1** bölüm yönetebilir.
- 1 bölüm yalnızca **en çok 1** öğretim üyesi tarafından yönetilebilir.



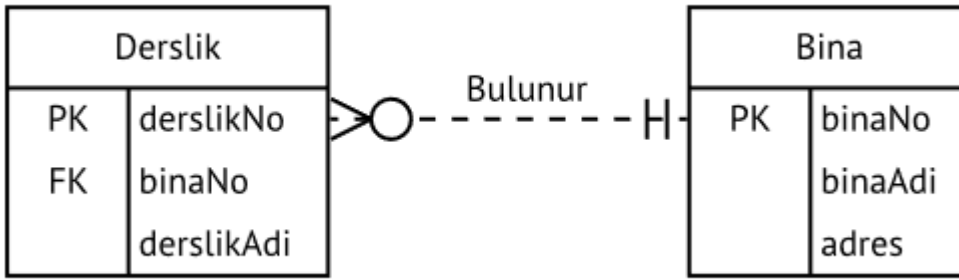
- Yalnızca 1 tarafının (en az 1 - en çok 1) varlığının (OgretimUyesi) birincil anahtarı (sicilNo) diğer varlığa yabancı anahtar alan (bolumBaskani) olarak eklenir.

- **İlişkisel Şema (Metinsel Gösterim)**

- OgretimUyesi(**sicilNo: String**, TCKimlikNo: Char(11), bolumNo: Integer, ad: String, soyad: String)
- Bolum(**bolumNo: Integer**, fakulteNo: Integer, bolumBaskani: String, bolumAdi: String)

Bir-Çok Bağıntısı

- 1 derslik **en çok 1** binada bulunur.
- 1 binada **en çok çok** sayıda derslik bulunur.



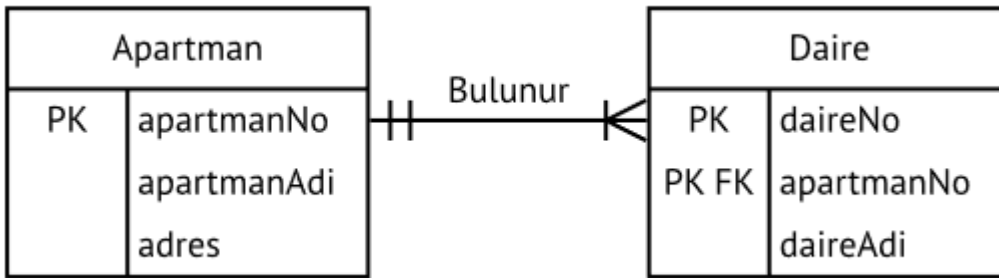
- 1 tarafının varlığının (Bina) birincil anahtarı (binaNo) diğer varlığa yabancı anahtar alan (binaNo) olarak eklenir

- **İlişkisel Şema (Metinsel Gösterim)**

- Derslik(**derslikNo: Integer**, binaNo: Integer, derslikAdi: String)
- Bina(**binaNo: Integer**, binaAdi: String, adres: String)

Bir-Çok Bağıntısı / Varolma Bağımlılığı, Tanımlama Bağıntısı

- 1 apartmanda **en çok çok** sayıda daire bulunur. En az bir daire bulunmak zorundadır.
- 1 daire **en çok 1** apartmanda (**en az 1 – en çok 1**) bulunmak zorundadır.

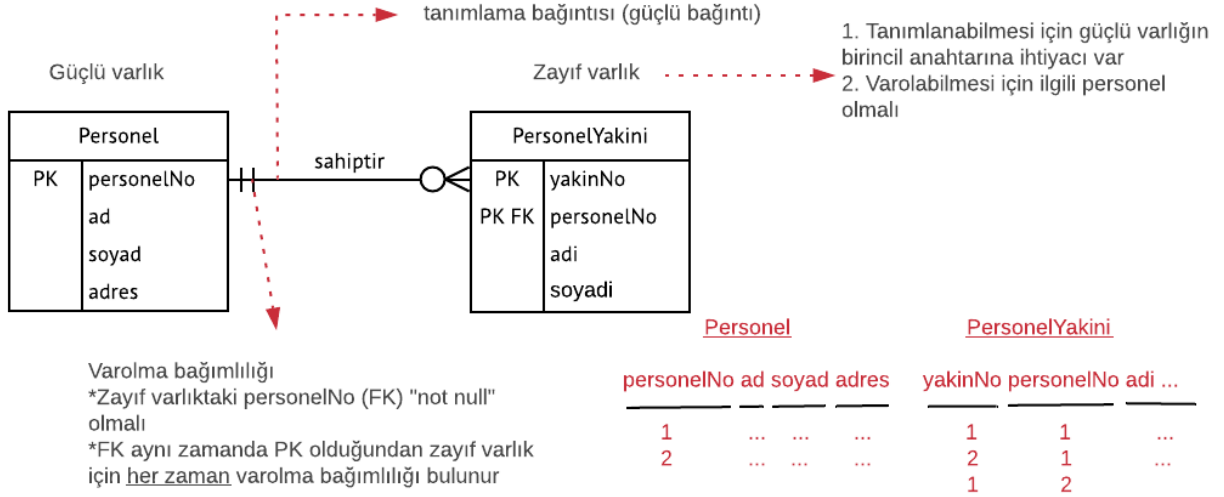


- 1 tarafının (Apartman) birincil anahtarı (apartmanNo) diğer varlığa hem yabancı anahtar hem de birincil anahtarın parçası (apartmanNo) olarak eklenir.

- **İlişkisel Şema (Metinsel Gösterim)**

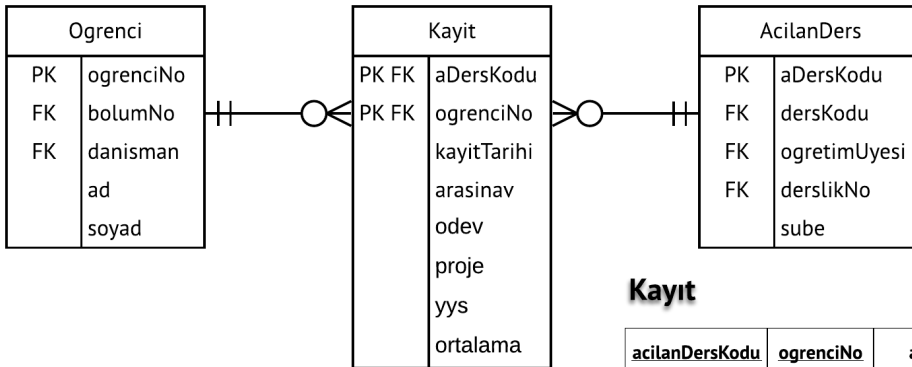
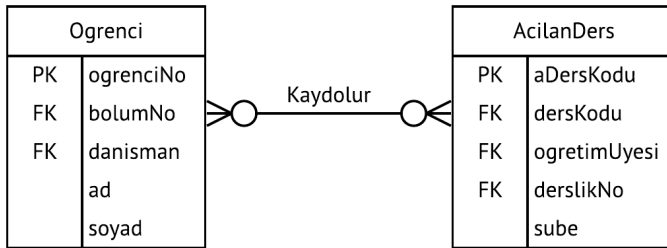
- Apartman(**apartmanNo: Integer**, apartmanAdi: String, adres: String)
- Daire(**daireNo: Integer**, **apartmanNo: Integer**, daireAdi: String)

- Var olma Bağımlılığı, Tanımlama Bağıntısı



Çok-Çok Bağıntısı

- 1 öğrenci **en çok çok** sayıda derse kayıt yaptırabilir.
- 1 ders **en çok çok** sayıda öğrenci tarafından alınabilir.



acilanDersKodu	ogrenciNo	arasinav	odev	proje ...
1	1	80	21	100
1	2	90	56	90
3	1	90	75	40
3	2	50	67	95
3	3	40	75	100

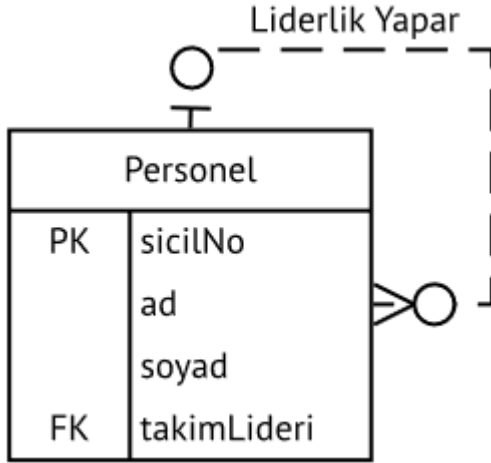
İlişkisel Şema (Metinsel Gösterim)

- Oğrenci(**ogrenciNo: String**, bolumNo: Integer, danisman: String, ad: String, soyad: String)
- Kayıt(**aDersKodu: String**, **ogrenciNo: String**, kayitTarihi: Date, not: Float)

- AcilanDers(**aDersKodu: String**, dersKodu: String, ogretimUyesi: String, derslikNo: Integer, sube: Char(1))

Tekli Bağıntı

- 1 personele **en çok 1** personel liderlik yapar.
- 1 personel **en çok çok** sayıda personele liderlik yapar.



- **İlişkisel Şema (Metinsel Gösterim)**
 - Personel(**sicilNo: String**, ad: String, soyad: String, takimLideri: String)

İndeks

- **İndeks:** Arama, listeleme işlemlerini hızlandırmak için kullanılan nesne.
- Aranılan bilginin konumunun, anahtar değerinden yararlanılarak bulunabilmesini sağlar.
- Bir indeks, diğer indeksleri etkilemeksizin yok edilebilir ya da oluşturulabilir.
- Genelde aramaların sık yapıldığı özellikler indeks yapılıdır.
- İndeks alanlarla yapılan aramalar çok hızlı olurken kayıt ekleme ve silme (INSERT, DELETE) işlemleri nispeten yavaş olur.
- İndeks alanlar sıralı olarak listelenir.
- Birincil anahtar (primary key) alanlar otomatik olarak eşsiz indeks (unique index) olurlar.
- Bir tabloda çok sayıda indeks tanımlanabilir. Her indeks yalnızca bir tabloya ait olabilir.
- Birden fazla alan beraber indeks olarak tanımlanabilir.

İndeks		Tablo				
dersNo (İndex Anahtarı)	İsaretçi	<u>aDersKodu</u>	dersNo	donem	ogretimUyesi	sube
T001	1,4	A130	T001	17G	21	A
T002	2	A257	T002	18G	56	A
C001	3,5	A102	C001	17B	75	A
		A131	T001	18G	67	B
		A103	C001	18B	75	B

Veri Sözlüğü (Sistem Kataloğu)

- **Veri Sözlüğü (Sistem Kataloğu):** Üst verilerin (metadata) saklandığı veritabanıdır.
- Veritabanı Yönetim Sistemi tarafından yönetilir.
- Oluşturulan tüm veritabanlarına ait üst veriler (tablolar, tablo alanları, alan tipleri, değer aralıkları, anahtarlar, indeksler, bağıntılar, kısıtlar vb.) burada saklanır.
- Kullanıcı tarafından sorgulanabilir.

Örnek Uygulama - Araç Üreticisi

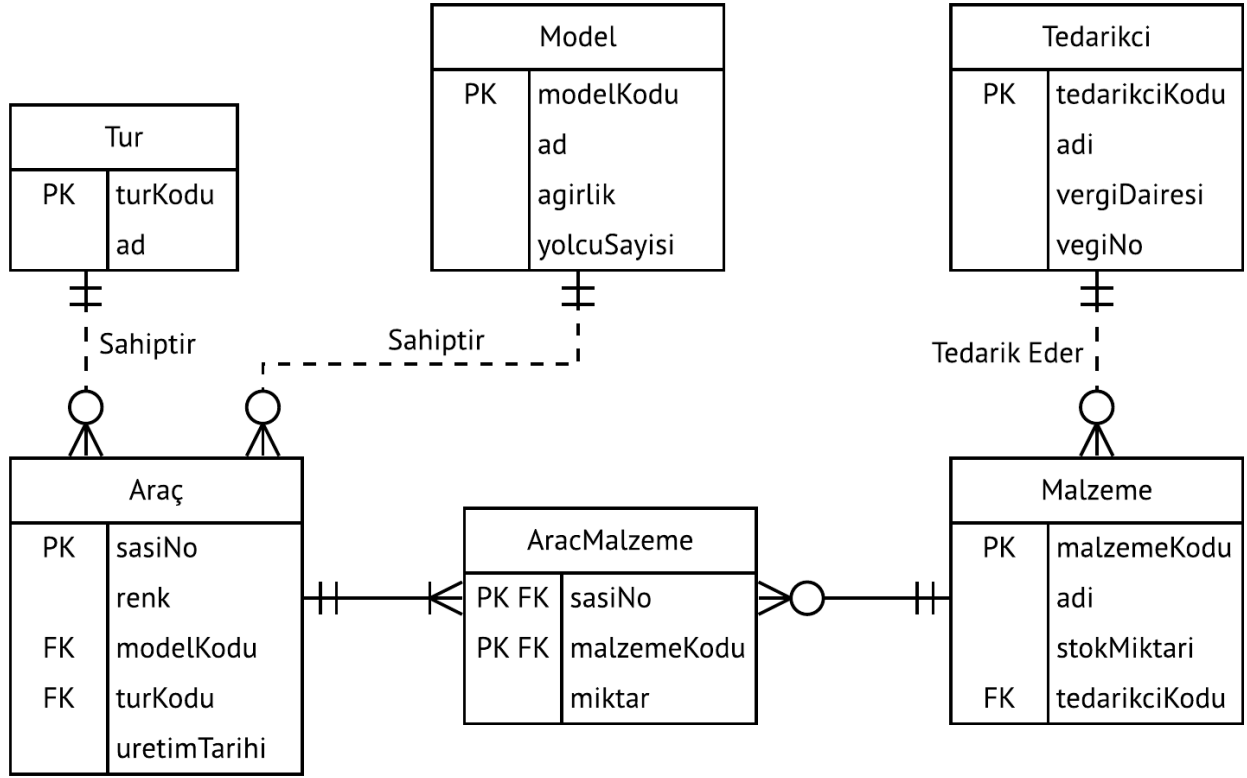
Senaryo

Bir araç üreticisi, farklı modellerde araçlar üretmektedir. Ürettiği araçlar, araçlarda kullanılan malzemeler, tedarikçiler vb. bilgilerini saklamak ve yönetmek istediği bir yazılım sistemi talep etmektedir.

İş Kuralları

- Üretilen her bir aracın şasi numarası, renk, tür (otomobil, kamyon, minibüs vb.), model ve üretim tarihi bilgileri mevcuttur. Araçlar birbirinden şasi numarası ile ayırt edilir.
- Araç türlerinin kodu ve adı bilgileri mevcuttur.
- Araç modellerinin kodu, adı, ağırlık ve yolcu sayısı bilgileri mevcuttur.
- Araçlar farklı miktarda ve farklı türlerde malzemelerden üretilir. Bu bilgiler de kaydedilmelidir.
- Malzemeler tedarikçi şirketlerden satın alınmaktadır.
- Malzemelerin kodu, adı, stok miktarı bilgileri mevcuttur.
- Tedarikçi şirketlerin kodu, adı, vergi dairesi, vergi numarası bilgileri mevcuttur.
- Bir tedarikçi hiçbir malzeme tedarik etmeyebileceği gibi bir veya daha fazla malzeme de tedarik edebilir. Bir malzeme yalnızca bir tedarikçi tarafından temin edilir.
- Bir araç çok sayıda malzemeden oluşur. En az bir adet malzemeden oluşmak zorundadır. Bir malzeme birden fazla araçta kullanılabilir. Ancak henüz hiçbir araçta kullanılmamış da olabilir.
- Bir araç yalnızca bir türe sahip olur. Bir türe ait hiçbir araç olmayabileceği gibi çok sayıda araç da olabilir.
- Bir araç yalnızca bir modele sahip olur. Bir modele ait hiçbir araç olmayabileceği gibi çok sayıda araç da olabilir.

Varlık Bağını Diyagramı



İlişkisel Şema (Metinsel Gösterim)

- Arac(**sasiNo:char**, renk: varchar, modelKodu: char, turKodu: char, uretimTarihi: date)
- AracMalzeme(**sasiNo: char**, **malzemeKodu: char**, miktar: int)
- Malzeme(**malzemeKodu: char**, adi: varchar, stokMiktari: int, tedarikciKodu: char)
- Tur(**turKodu: char**, ad: varchar)
- Model(**modelKodu: char**, ad: varchar, agirlik: real, yolcuSayisi: int)
- Tedarikci(**tedarikciKodu: char**, ad: varchar, vergiDairesi: varchar, vergiNo: char)

Örnek Uygulama – Elektronik Ticaret Sistemi

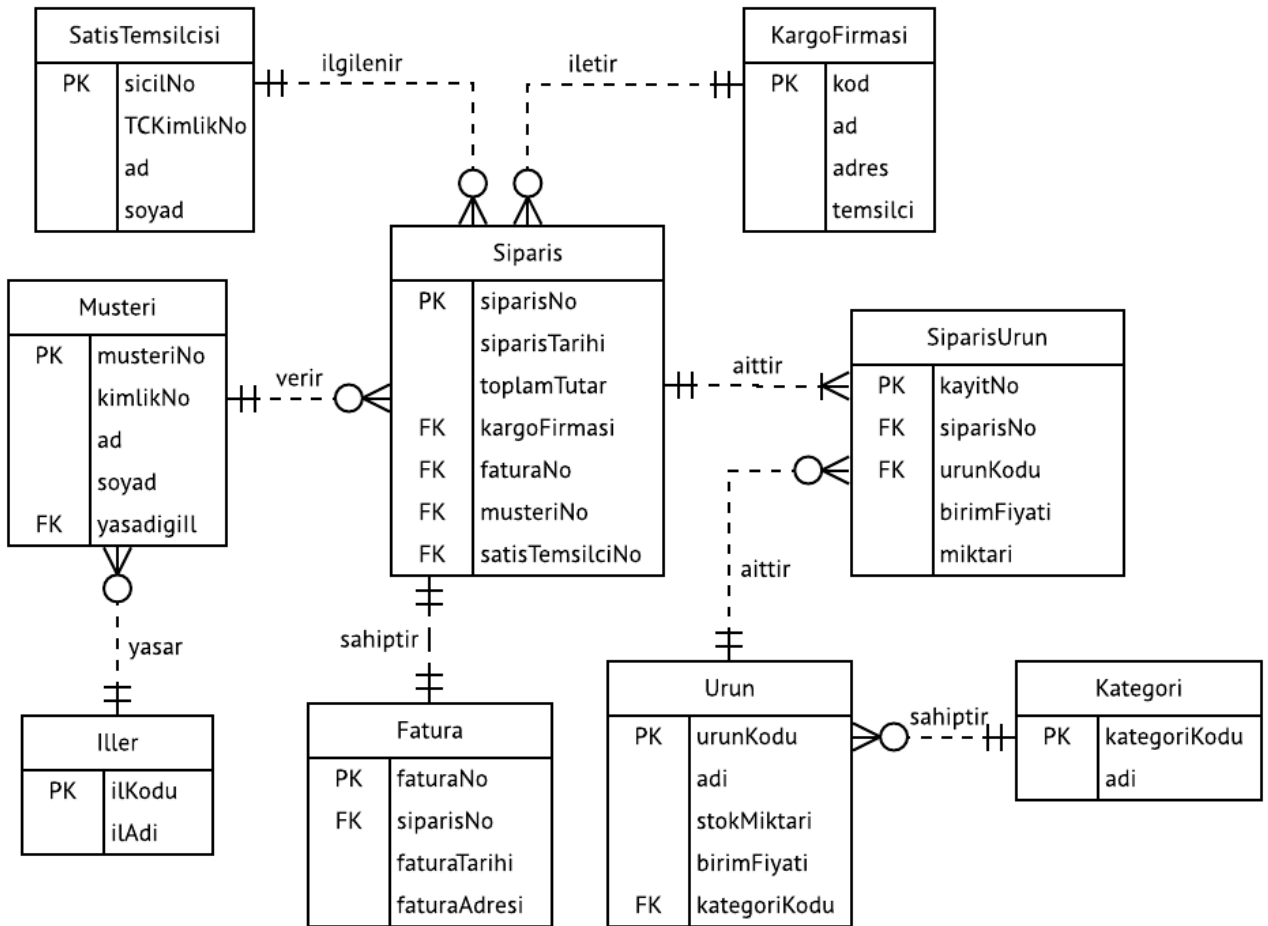
Senaryo

Elektornik ticaret yazılımının ihtiyacı olan verilerin yönetimi için bir veritabanı geliştirilmesi isteniyor. İş kuralları verilen bu veritabanının VB diyagramını ve ilişkisel şemasını oluşturunuz.

İş Kuralları

- Bu veritabanında her müşteriye bir numara verilerek müşterinin TC Kimlik numarası, adı, soyadı, yaşadığı ili saklanması düşünülüyor.
- İllerin plaka numarası ve adı saklanır.
- Her siparişe bakan bir satış temsilcisi mevcuttur. Satış temsilcilerinin TCKimlikNo, ad ve soyad bilgileri mevcuttur.
- Ürünlerin (kişisel bilgisayar, telefon vb.) kodu, adı, fiyatı ve stok miktarlarının saklanması gerekmektedir.
- Ürünlerin kategorileri (bilgisayar, ev elektroniği, kozmetik vb.) mevcuttur.
- Müşterilerin ürün siparişleri saklanarak her bir siparişe bir fatura kesilmesi sağlanmalıdır.

- Her siparişin eşsiz bir sipariş numarası ve sipariş tarihi mevcuttur.
- Sipariş edilen bir ürünün sipariş adedi ve birim fiyatı (kişiye özel indirim v.s. nedeniyle ürün tablosundaki fiyattan farklılık gösterebilir) da kaydedilmelidir.
- Faturaların fatura numarası, tarih ve fatura adresi bilgileri saklanmalıdır. (Toplam fiyat hesaplanabilir ya da saklanabilir).
- Siparişler bir kargo firması tarafından iletilir. Kargo firmasının kodu, adı, adresi bilgileri yer alır. Her kargo firmasında siparişlerden sorumlu bir yetkili yer alır.
- Bir ürünün yalnızca bir kategorisi mevcuttur. Bir kategori çok sayıda ürünün kategorisi olabilir.
- Bir siparişte en az bir ürün bulunur. Ancak çok sayıda ürün de bulunabilir. Bir ürün çok sayıda siparişte yer alabilir.
- Bir müşteri çok sayıda sipariş verebilir. Bir sipariş yalnızca bir müşteri tarafından verilebilir.
- Bir siparişin yalnızca bir faturası olabilir. Bir fatura yalnızca bir siparişin faturası olabilir.
- Bir sipariş ile yalnızca bir satış temsilcisi ilgilenir. Bir satış temsilcisi çok sayıda sipariş ile ilgilenilebilir.
- Bir müşteri yalnızca bir ilde yaşayabilir. Bir ilde çok sayıda müşteri yaşayabilir.
- Bir sipariş yalnızca bir kargo firması tarafından iletilir. Bir kargo firması çok sayıda sipariş iletebilir.



urunKodu	urunAdi	birimFiyati	seriNo	musteriNo
ELK001	Televizyon	1000	0001	11
ELK001	Televizyon	1000	0002	120
ELK001	Televizyon	1000	0003	130
BİL002	Bilgisayar	3000	0130	18
BİL002	Bilgisayar	3000	1200	10

id	urunKodu	seriNo	musteriNo
1	ELK001	0001	11
2	ELK001	0002	120
3	ELK001	0003	130
4	BİL002	0130	18
5	BİL002	1200	10

urunKodu	urunAdi	birimFiyati
ELK001	Televizyon	1000
BİL002	Bilgisayar	3000

İlişkisel Şema (Metinsel Gösterim)

- SatisTemsilcisi(**sicilNo: char**, TCKimlikNo: char, ad: varchar, soyad: varchar)
- KargoFirması(**kod: char**, ad: varchar, adres: varchar, temsilci: varchar)
- Musteri(**musteriNo: char**, TCKimlikNo: char, ad: varchar, soyad: varchar, yasadigill: varchar)
- Siparis(**siparisNo: char**, siparisTarihi: date, toplamTutar: real, kargoFirması: char, faturaNo: char, musterino: char, satisTemsilciNo: char)
- SiparisUrun(**kayitNo: int**, siparisNo: char, urunKodu: char, birimFiyati: real, miktar: int)
- Kategori(**kategoriKodu: char**, adi: varchar)
- İller(**plakaNo: varchar**, ilAdi: varchar)
- Fatura(**faturaNo: char**, tarih: date)
- Urun(**urunKodu: char**, adi: varchar, stokMiktari: int, birimFiyati: real, kategoriKodu: char)

Kaynaklar

- Carlos Coronel, Steven Morris, and Peter Rob, Database Systems: Design, Implementation, and Management, Cengage Learning.