

İLİŞKİSEL VERİTABANLARI

Veritabanı Nedir?

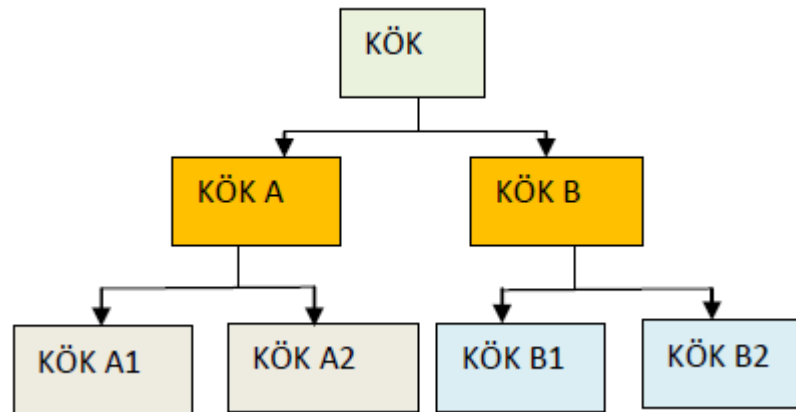
Veritabanı (database) en basit şekliyle verilerin belirli bir düzene göre tutulduğu, depolandığı bir sistemdir.

İlişkisel Veritabanı Nedir?

İlişkisel veritabanlarındaki yapılarda veriler tablolar halinde saklanır ve bu tablolar arasında bağlantılar(ilşkiler) mevcuttur.

İlişkisel veritabanlarından önce;

Hiyerarşik veritabanları kullanılırdı. Bu veritabanları için oluşturulan yapı da verinin diske yazılma sırası için iyi bir algoritma kullanılmıyorsa verileri okumak oldukça zaman alırdı. Bu yüzden zamanla hiyerarşik veritabanlarında yazılım karmaşıklığı artmış ve yeni yöntemler aranmaya başlanmıştı.



Ortaya Çıkışı

1970’li yıllarda BMI mühendislerinden Edgar Frank Codd tarafından yazılan “A Relational Model of Data for Large Shared Data Banks” adlı makale ile ilişkisel veritabanı ortaya çıkmış ve duyurulmuştur.

Makalesinde; ilişkisel veritabanı mantığını, tasarımının nasıl yapılacağını anlatmıştır. Gerçek hayattaki verilerin tablolar şeklinde tutulabileceğini, tablolarda satırlar ve sütunlar oluşacak şekilde veri sakladığından bahsetmiştir.

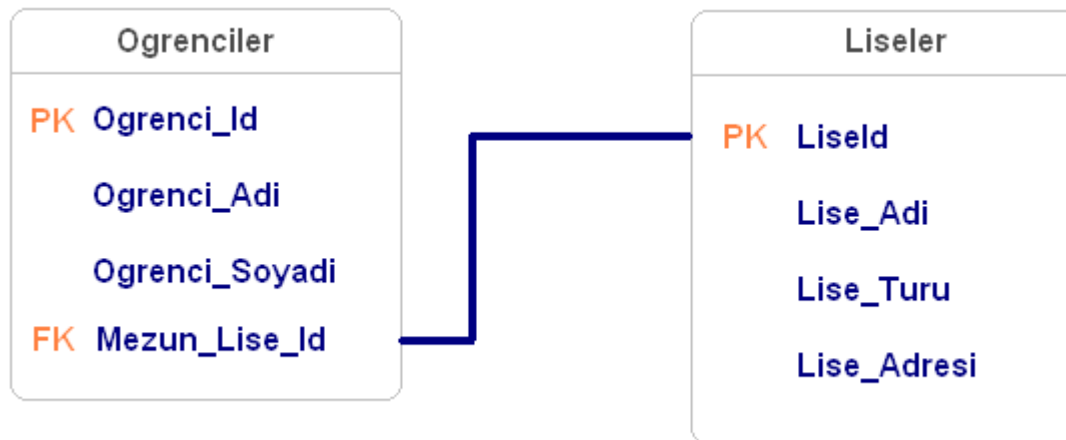
İlişkisel veritabanı için kesinleşmiş genel kural ve özelliklerin başında tablolarda tekrarlı ifadelerin bulunmayacağı, tablo konuları ile verilerin örtüşeceği ve tabloların çoğunda birincil anahtarın olduğu gelir. İlişkisel veritabanı için bazı kurallar ve avantajlar şöyledir:

- Bir bilgi bir veritabanında sadece bir kere bulunabilir. Bu ise karmaşıklığı azaltmamış olur. Çünkü bu sayede bir veri sadece bir yerden eklenebilir, silinebilir ve okunabilir.
- Veriler tablolar için belirlenen özelliklere göre yerleştirilir. Bu sayede veri bütünlüğü artar ve tablolar arası geçişler daha kolay hale gelir.

- Veriler tablolara satır ve sütunlara yerleştirilecek şekilde kaydedilir.
- Verilere genellikle birincil anahtar aracılığıyla ulaşım sağlanabilir.
- Farklı içerikteki tablolar arası bağlantılar için anahtarlar (keys) kullanılır.
- İlişkisel veritabanlarında sadece kayıt alma (yazma) ve okuma işlemleri yüksek hızla yapılmamaktadır. Bunların yanı sıra ekleme, güncelleme ve silme işlemleri de hızlı bir şekilde yapılabilir.

- İlişkisel veritabanlarının çoğunda bahsedilen ekleme, güncelleme, silme işlemleri farklı kullanıcılara verilebilir.
- İlişkisel veritabanı modelleri arasında kurallara bağlı kalınarak veri aktarımları mümkündür.
- Birden fazla kullanıcı, program kayıtlı olan veriye aynı anda ulaşım sağlamak isteyebilir. İlişkisel veritabanı bu konuda veri bütünlüğü bozulmadan işlem yapmaya yarar.

- İlişkisel veritabanlarında bir tablonun bir özelliğine (sütununa) kayıtlı olan verilere ulaşım sağlanmak istenirken diğer sütunlardan bağımsız erişim sağlanabilir. Böylece gereksiz iş yükü yapılmamış olur.
 - İlişkisel veritabanında veri bütünlüğünün korunması için belli kısıtlamalar alınabilir. Birincil anahtarlar bunların bir örneğidir. Mesela yabancı anahtarlar(foreign keys) ile bağlı olduğu tablo da kayıtlı olmayan bir verinin devam bilgileri alt tablo da tutulmaz.
- Yabancı anahtar(foreign key) ;** tablolar arası bağlantılarda alt tablo da, ana tablonun birincil anahtarının tutulduğu sütundur.



Bu gibi özelliklerin geliştirilmesi ve sağlanması sonucu ilişkisel veritabanı yapısı da oturmuş bulunmaktadır.

İlişkisel veri tabanları için dil, ilk olarak Codd ile ortaya çıktı ve SQL olarak adlandırıldı.

Günümüzde üzerine eklemeler yapılarak geliştirilmiş durumda olanları mevcuttur.



İlişkisel veritabanları için bazı terimler;

Tablo : Veritabanlarını oluşturan bilgi depolarıdır. Veritabanına girilen veriler tablolarda tutulur. İlişki tablolar arasında oluşturulur. Tablolar satırlar ve sütunlardan oluşur.

Satır : Tablodaki verileri bulundurur.

Sütun : Tablonun özelliklerini bulundurur.

Tuple : İlişkisel veritabanı sistemlerinde tablo içerisindeki bir satırı ifade eder .

Alanlar

First Name	Last Name	Address	City	Age
Mickey	Mouse	123 Fantasy Way	Anaheim	73
Bat	Man	321 Cavern Ave	Gotham	54
Wonder	Woman	987 Truth Way	Paradise	39
Donald	Duck	555 Quack Street	Mallard	65
Bugs	Bunny	567 Carrot Street	Rascal	58
Wiley	Coyote	999 Acme Way	Canyon	61
Cat	Woman	234 Purrfect Street	Hairball	32
Tweety	Bird	543	Itotltaw	28

Kayıt →

Relation Instance : Bir ilişkisel veritabanında sonlu sayıdaki tuple temsil eder. Tekrar eden kayıtlar yer alamaz.

Relation Schema : Veritabanının tanımlarının yer aldığı bütündür. İlişkinin adı ile ilişki niteliklerinin adlarından oluşur.

Primary Key : Birincil Anahtar. İlişkisel veritabanlarında tablolar oluşturulurken, tablonun her kaydının kimlik numarası niteliğinde bir kolon oluşturulur ve bu kolon içinde tablonun her kaydı için benzersiz(unique) bir değer verilir.

Constraints (Kısıtlamalar) : Her ilişki kurulurken belirlenmesi gereken şartlar olması gereklidir. Bu koşullar ile tutarlı halde kalırlar.

Kısıtlar veritabanındaki bütünlüğü sağlamak amacıyla oluşturulmuş kurallar olarak tanımlanabilir. Yani kısıtlar veritabanına hatalı giriş yapılmasını engeller.

Primary Key (Birincil Anahtar) Kısıtlaması : Birincil anahtar kısıtlaması bir tabloda bir birincil anahtar tanımlaması için kullanılır. Bir tabloda sadece bir tane birincil anahtar olabilir. Birincil anahtar olarak tanımlanan o alanda girilen her değer birbirinden farklı olmak zorundadır. Null değeri tanımlanmasına izin verilmez ve otomatik olarak belirlenen o alanda bir içerik yaratılır.

Null değer; herhangi bir veri türü olmayan bir alandır. Veri türü olmamasından dolayı herhangi bir işleme dahil edilmez. Dolayısıyla Null varlığı veritabanı uygulamalarında bir boşluğu ifade etmesinden dolayı işlemlerde dikkat edilmeli, göz önüne alınmalıdır.

•**Domain (İlgi Alanı) Kısıtlaması** : Tablolarda her alanın veri türü belirlidir. Farklı veri türünün girilmesini engellemek için kullanılan koşullar bütünüdür.

•**Foreign Key (Yabancı Anahtar) Kısıtlaması** : İlişkisel veritabanı temel mantığı gereği anlamlı bir veri birden fazla tabloda parçalanarak tutulması gerekebilir. Bunun amaçla tablolar arası ilişkiler kurulur. Aksi halde veri anlamsızlaşır. Dolayısıyla veritabanının da tutarsızlığa yol açar. Bu nedenle tablolar arasında uyum kontrolleri yada tablolar arasında bir tabloda yer alan kaydın anlamlı olabilmesi için diğer tabloda kaydın olması kontrolleri bu kısıtlama grubuna girer.

Veritabanları günümüzde birçok yerde kullanılır. Hastane, banka, kütüphane vb. aklımıza gelebilecek özellikle kayıt esaslı sistemlerde veritabanı olmazsa olmaz durumdadır. Bu açıdan veriler arası hızlı işlem yapmak, yapılan işlemin karmaşıklığını azaltmak ve güvenilirliği artırmak amaçlı ilişkisel veritabanları kullanılır.

KAYNAKÇA

- <https://doganakyurek.blogspot.com.tr/2016/08/iliskisel-veritabani-teorisi-tarihi.html>
- http://www.academia.edu/17793841/%C4%B0li%C5%9Fkisel_Veritaban%C4%B1_Modeli