

Bir DBMS kullanmanın avantajları az önce listelenenlerle sınırlı değildir. Aslında, veritabanlarının teknik detayları ve uygun tasarımları hakkında daha fazla bilgi edindikçe daha birçok avantaj keşfedeceksiniz.

1-3b Veritabanı Türleri

Bir DBMS birçok türde veritabanı oluşturmak için kullanılabilir. Her veritabanı belirli bir veri topluluğunu depolar ve belirli bir amaç için kullanılır. Yıllar geçtikçe, teknoloji ve veritabanlarının yenilikçi kullanımları geliştikçe, veritabanlarını sınıflandırmak için farklı yöntemler kullanılmıştır. Örneğin, veri tabanları desteklenen kullanıcı sayısına, verilerin bulunduğu yere, depolanan veri türüne, amaçlanan veri kullanımına ve verilerin yapılandırılma derecesine göre sınıflandırılabilir.

Tek kullanıcı veritabanı

Aynı anda yalnızca bir kullanıcıyı destekleyen bir veritabanı.

Çok kullanıcı veritabanı

Birden fazla eşzamanlı kullanıcıyı destekleyen bir veritabanı.

Masaüstü veritabanı

Kişisel bir bilgisayarda çalışan tek kullanıcı bir veritabanı.

Çalışma grubu veritabanı

Genellikle 50'den az kullanıcıyı destekleyen veya bir kuruluştaki belirli bir departman için kullanılan çok kullanıcı bir veritabanı.

Kurumsal veritabanı

Şirketin genel veri temsili mevcut ve gelecekte beklenen ihtiyaçlar için destek sağlar.

Merkezi veritabanı

Tek bir sitede bulunan bir veritabanı.

Dağıtılmış veritabanı

Fiziksel olarak bağımsız iki veya daha fazla sitede depolanan mantıksal olarak ilişkili bir veritabanı.

Bulut veritabanı

Microsoft Azure veya Amazon AWS gibi bulut hizmetleri kullanılarak oluşturulan ve bakımı yapılan bir veritabanı.

Genel amaçlı veritabanı

Birden fazla disiplinde kullanılan çok çeşitli verileri içeren bir veritabanı.

Disipline özgü veri tabanı

Belirli bir konu alanına odaklanmış verileri içeren bir veritabanı.

tiuser. **Tek kullanıcı** bir **veritabanı** aynı anda yalnızca bir kullanıcıyı destekler. Başka bir deyişle, A kullanıcısı veritabanını kullanıyorsa, B ve C kullanıcıları A kullanıcısının işi bitene kadar beklemelidir. Kişisel bir bilgisayarda çalışan tek kullanıcı bir veritabanına **masaüstü veritabanı** denir. Buna karşılık, **çok kullanıcı** bir **veritabanı** aynı anda birden fazla kullanıcıyı destekler. Çok kullanıcı veritabanı nispeten az sayıda kullanıcıyı (genellikle 50'den az) veya bir kuruluş içindeki belirli bir departmanı desteklediğinde, buna **çalışma grubu veritabanı** denir. Veritabanı tüm kuruluş tarafından kullanıldığında ve birçok departmanda birçok kullanıcıyı (50'den fazla, genellikle yüzlerce) desteklediğinde, veritabanı **kurumsal veritabanı** olarak bilinir.

Konum, veritabanını sınıflandırmak için de kullanılabilir. Örneğin, tek bir sitede bulunan verileri destekleyen bir veritabanına **merkezi veritabanı** denir. Birkaç farklı siteye **dağıtılmış** verileri destekleyen bir veritabanı ise **dağıtılmış** veritabanı olarak adlandırılır. Bir veritabanının ne ölçüde dağıtılabileceği ve bu dağıtımın nasıl yönetileceği Bölüm 12, Dağıtık Veritabanı Yönetim Sistemleri'nde ayrıntılı olarak ele alınmaktadır.

Hem merkezi hem de merkezi olmayan (dağıtık) veritabanları, veritabanını uygulamak ve işletmek için iyi tanımlanmış bir altyapı (donanım, işletim sistemleri, ağ teknolojileri, vb.) gerektirir. Tipik olarak, altyapı, veritabanını oluşturan ve işleten kuruluş tarafından ve sürdürülür. Ancak son yıllarda, bulut veritabanlarının kullanımı popülerlik kazanmaktadır. **Bulut veritabanı**, Microsoft Azure veya Amazon AWS gibi **bulut** veri hizmetleri kullanılarak oluşturulan ve bakımı yapılan bir veritabanıdır. Üçüncü taraf satıcılar tarafından sağlanan bu hizmetler, veritabanı için tanımlanmış performans ölçütleri (veri depolama kapasitesi, gerekli iş hacmi ve kullanılabilirlik) sağlar, ancak bunu uygulamak için temel altyapıyı belirtmek zorunda değildir. Veri sahipleri, veritabanlarını desteklemek için hangi donanım ve yazılımın kullanıldığını bilmek ya da bu konuda endişe duymak zorunda değildir. Veritabanı üzerindeki iş talepleri değiştikçe performans kapasiteleri bulut sağlayıcısı ile yeniden müzakere edilebilir. Örneğin, hastanelerde sağlık hizmetleri analiz yazılımı alanında dünyanın en büyük sağlayıcısı olan 3M Health Information Systems, birden fazla BT merkezini konsolide etmek için Amazon'un AWS bulut veritabanı hizmetlerini kullandı. 3M'in herhangi bir donanım, işletim sistemi ya da ağ cihazı satın alması, kurması, yapılandırması ya da bakımını yapması gerekmedi. Sadece verileri ve uygulamaları için depolama ve işleme kapasitesi satın aldı. Veritabanlarına olan talep arttıkça, gerektiğinde ek işleme ve depolama kapasiteleri satın alınabiliyordu. Sonuç olarak, daha önce tamamlanması 10 hafta süren sunucu hazırlama işlemleri yalnızca birkaç dakika içinde yapılabiliyordu. Bu da şirketin müşterilerin ihtiyaçlarına daha duyarlı olmasını ve daha hızlı yenilik yapmasını sağlıyor.

Araştırma ortamları gibi bazı bağlamlarda, veritabanlarını sınıflandırmanın popüler bir yolu, içlerinde depolanan verilerin türüne göre yapılır. Bu kriter kullanılarak veri tabanları iki kategoriye ayrılır: genel amaçlı ve disipline özel veri tabanları. **Genel amaçlı** veri **tabanları**, birden fazla disiplinde kullanılan çok çeşitli verileri içerir; örneğin, genel demografik verileri içeren bir nüfus sayımı veri tabanı ve çeşitli konulara ilişkin gazete, dergi ve gazete makalelerini içeren LexisNexis ve ProQuest veri tabanları gibi. **Disipline özgü** veri **tabanları** belirli konu alanlarına odaklanmış veriler içerir. Bu tür veri tabanındaki veriler esas olarak küçük bir disiplin kümesi içinde akademik veya araştırma amaçlı kullanılır. Disipline özgü veritabanlarına örnek olarak CompuStat veya CRSP gibi veritabanlarında saklanan finansal veriler verilebilir.

(Center for Research in Security Prices), coğrafi ve diğer ilgili verileri depolayan coğrafi bilgi sistemi (GIS) veritabanları ve gizli tıbbi geçmiş verilerini depolayan tıbbi veritabanları.

Ancak günümüzde veri tabanlarını sınıflandırmanın en popüler yolu, nasıl kullanılacaklarına ve onlardan toplanan bilgilerin zaman hassasiyetine dayanmaktadır. Örneğin, ürün veya hizmet satışları, ödemeler ve tedarik alımları gibi işlemler kritik günlük operasyonları yansıtır. Bu tür işlemler doğru ve anında kaydedilmelidir. Öncelikle bir şirketin günlük operasyonlarını desteklemek için tasarlanmış bir veritabanı, **çevrimiçi işlem işleme (OLTP) veritabanı, işlem veritabanı** veya **üretim veritabanı** olarak da bilinen **operasyonel bir veritabanı** olarak sınıflandırılır. Buna karşılık, **analitik bir veritabanı** öncelikle geçmiş verileri ve yalnızca taktiksel veya stratejik karar verme için kullanılan iş ölçümlerini depolamaya . Bu tür analizler tipik olarak fiyatlandırma kararları, satış tahminleri, pazar stratejileri ve benzerlerinin dayandırılacağı bilgileri üretmek için kapsamlı "veri masajı" (veri manipülasyonu) gerektirir. Analitik veritabanları, son kullanıcının sofistike araçlar kullanarak iş verilerinin gelişmiş analizini yapmasına olanak tanır.

Analitik veritabanları tipik olarak iki ana bileşenden oluşur: bir veri ambarı ve bir çevrimiçi analitik işleme ön ucu. **Veri ambarı**, verileri karar desteği için optimize edilmiş bir formatta depolayan özel bir veri tabanıdır. Veri ambarı, operasyonel veritabanlarından elde edilen geçmiş verilerin yanı sıra diğer harici kaynaklardan gelen verileri de içerir. **Çevrimiçi analitik işleme (OLAP)**, veri ambarından verilerin alınması, işlenmesi ve modellenmesi için gelişmiş bir veri analizi ortamı sağlamak üzere birlikte çalışan bir dizi araçtır. Son zamanlarda, bu veritabanı uygulama alanının önemi ve kullanımı artmış, öyle ki kendi disiplinine dönüşmüştür: iş zekası. İş **zekası** terimi, iş kararlarını desteklemek için bilgi üretmek amacıyla iş verilerini yakalamak ve işlemek kapsamlı bir yaklaşımı tanımlar. Bölüm 13, İş Zekası ve Veri Ambarları, bu konuyu ayrıntılı olarak ele almaktadır.

Veritabanları, verilerin yapılandırılma derecesini yansıtacak şekilde de sınıflandırılabilir. **Yapılandırılmamış veri**, orijinal (ham) haliyle, yani toplandığı formatta var olan veridir. Bu nedenle, yapılandırılmamış veriler, bilgi sağlayan işlemlere uygun olmayan formatta bulunur. **Yapılandırılmış veriler**, yapılandırılmamış verilerin depolanmasını, kullanılmasını ve bilgi üretilmesini kolaylaştırmak için biçimlendirilmesinin sonucudur. Veriler üzerinde gerçekleştirmeyi düşündüğünüz işleme türüne göre yapı (format) uygularsınız. Bazı veriler bazı işleme türleri için hazır (yapılandırılmamış) olmayabilir, ancak diğer işleme türleri için hazır (yapılandırılmış) olabilirler. Örneğin, 37890 veri değeri bir posta kodunu, bir satış değerini veya bir ürün kodunu ifade ediyor olabilir. Bu değer bir posta kodunu veya ürün kodunu temsil ediyorsa ve metin olarak saklanıyorsa, bununla matematiksel hesaplamalar yapamazsınız. Öte yandan, bu değer bir satış işlemini temsil ediyorsa, sayısal olarak biçimlendirilmelidir.

Yapı kavramını daha iyi açıklamak için bir yığın basılı kağıt fatura düşünün. Bu faturaları yalnızca ileride almak ve görüntülemek için resim olarak saklamak istiyorsanız, tarayabilir ve grafik formatında kaydedebilirsiniz. Öte yandan, aylık toplam ve ortalama satışlar gibi bilgileri üretmek istiyorsanız, bu tür bir grafik depolama yararlı olmayacaktır. Bunun yerine, gerekli hesaplamaları yapabilmek için fatura verilerini (yapılandırılmış) bir elektronik tablo biçiminde saklayabilirsiniz. Aslında, karşılaştığınız çoğu veri en iyi yarı yapılandırılmış olarak sınıflandırılır. **Yarı yapılandırılmış veriler** zaten bir dereceye kadar işlenmiştir (genellikle biçimlendirme etiketleri veya bazı işaretleme dilleri kullanılarak) ancak ilişkisel modelin tipik katı tablo biçimine uymaz. Örneğin, tipik bir web sayfasına bakarsanız, verilerin bazı bilgileri iletmek için önceden düzenlenmiş bir biçimde sunulduğunu görürsünüz. Şimdiye kadar bahsedilen veritabanı türleri, yüksek düzeyde yapılandırılmış verilerin depolanması ve yönetimine odaklanmaktadır. Ancak, şirketler yapılandırılmış veri kullanımıyla sınırlı değildir. Yarı yapılandırılmış ve yapılandırılmamış verileri de kullanırlar. Şirket e-postalarında, notlarında ve prosedürler, kurallar ve web sayfaları gibi belgelerde bulunabilecek değerli bilgileri bir düşünün. Yapılandırılmamış ve yarı yapılandırılmış veri depolama ve yönetim ihtiyaçları, aşağıdaki gibi bilinen yeni nesil veritabanları aracılığıyla ele alınmaktadır

operasyonel

veritabanı Öncelikle bir şirketin günlük operasyonlarını

Öncelikle bir şirketin günlük desteklemek için tasarlanmış bir veritabanı. İşlemsel veritabanı olarak da bilinir,

OLTP veritabanı veya üretim veritabanı.

çevrimiçi işlem işleme (OLTP) veritabanı

Bkz. operasyonel veritabanı.

işlemsel veritabanı

Bkz. operasyonel veritabanı.

üretim veritabanı

Bkz. operasyonel veritabanı.

analitik veri tabanı

Öncelikle taktiksel veya stratejik karar verme için kullanılan geçmiş verileri ve iş metriklerini depolamaya odaklanan bir veritabanı.

veri ambarı

Geçmiş ve gelecek verileri depolayan özel bir veritabanı karar desteği için optimize edilmiş bir formatta toplanmış veriler.

çevrimiçi analitik işleme (OLAP)

Veri ambarından veri almak, işlemek ve modellemek için gelişmiş veri analizi sağlayan bir dizi araç.

iş zekası

İş kararlarını desteklemek için verileri yakalamak, toplamak, entegre etmek, depolamak ve analiz etmek kullanılan bir dizi araç ve süreç.

yapılandırılmamış veri

Orijinal, ham haliyle, yani toplandığı formatta var olan veriler.

yapılandırılmış veri

Depolama, kullanım ve bilgi üretimini kolaylaştırmak için biçimlendirilmiş veriler.

yarı yapılandırılmış veri

Bir dereceye kadar işlenmiş

Tablo 1.1 Veritabanı Türleri

Ürün	Kullanıcı Sayısı			Veri Konumu		Veri Kullanımı		XML
	Tek Kullancı	Çok kullanıcı		Merkezi	Dağıtılmış	Operasyonel	Analitik	
		Çalışma Grubu	Kurumsal					
MS Erişim	X	X		X		X		
MS SQL Server	X*	X	X	X	X	X	X	X
IBM DB2	X*	X	X	X	X	X	X	X
MySQL	X	X	X	X	X	X	X	X
Oracle RDBMS	X*	X	X	X	X	X	X	X

*Satışçı, tek kullanıcı/kişisel veya Express DBMS türlerini sunar

Genişletilebilir İşaretleme Dili (XML)

Veri öğelerini temsil etmek ve işlemek için kullanılan bir metal dil. Diğer biçimlendirme dillerinden farklı olarak XML, bir belgenin veri öğelerinin manipüle edilmesine izin verir.

XML veritabanı

Yarı yapılandırılmış XML şerhlerini depolayan ve yöneten bir veritabanı sistemi.

sosyal medya

"Her yerde, her zaman, her zaman açık" insan etkileşimlerini mümkün kılan web ve mobil teknolojiler.

NoSQL

Geleneksel ilişkisel veritabanı modeline dayanan yeni nesil DBMS.

XML veritabanları. **Genişletilebilir İşaretleme Dili (XML)**, veri öğelerini metinsel bir biçimde temsil etmek ve işlemek için kullanılan özel bir dildir. Bir **XML veritabanı**, yarı yapılandırılmış XML verilerinin depolanmasını ve yönetimini destekler.

Tablo 1.1, iyi bilinen birkaç veritabanı yönetim sisteminin özelliklerini karşılaştırmaktadır.

Web ve İnternet tabanlı teknolojilerin yeni "sosyal medya" neslinin temeli olarak ortaya çıkmasıyla birlikte, büyük miktarda veri depolanmakta ve analiz edilmektedir. **Sosyal medya**, "her yerde, her zaman, her zaman açık" insan etkileşimlerini mümkün kılan web ve mobil teknolojileri ifade eder. Google, Facebook, Instagram, Twitter ve LinkedIn gibi web siteleri son kullanıcılar ve tüketiciler hakkında büyük miktarda veri toplamaktadır. Bu veriler katlanarak büyümekte ve özel veritabanı sistemlerinin kullanılmasını gerektirmektedir. Örneğin, 2020 itibarıyla Twitter'da her gün 500 milyondan fazla tweet atılıyor ve bu sayı artmaya devam ediyor. Sonuç olarak, Twitter'ın kullanıcı içeriğini depolamak için kullandığı MySQL veri tabanı talep nedeniyle sık sık aşırı yükleniyordu.² Facebook da benzer zorluklarla karşı karşıya. Her gün gelen 500 terabayttan fazla veri ile 100 petabayttan fazla veriyi tek bir veri depolama dosya sisteminde saklamaktadır. Veritabanı, bu verilerden durum güncellemeleri, resim istekleri ve milyarlarca "Beğen" eylemi dahil olmak üzere kullanıcı eylemlerini işlemek için her saat 200 terabayttan fazla veriyi taramaktadır.³ Geçtiğimiz birkaç yıl içinde, bu yeni tür özel veritabanı karmaşıklık ve yaygın kullanım açısından büyümüştür. Şu anda bu yeni veritabanı türü NoSQL veritabanı olarak bilinmektedir. **NoSQL** (Not only SQL) terimi genellikle geleneksel ilişkisel veritabanı modeline dayanmayan yeni nesil DBMS'leri tanımlamak için kullanılır. NoSQL veritabanları, bu yeni iş gereksinimlerinin karakteristiği olan benzeri görülmemiş veri hacmini, çeşitli veri türlerini ve yapılarını ve veri işlemlerinin hızını işlemek için tasarlanmıştır. Bu tür sistemler hakkında daha fazla bilgiyi Bölüm 2, Veri Modelleri'nde bulabilirsiniz.

Bu bölümde birçok veri tabanı türünden kısaca bahsedilmiştir. Daha önce öğrendiğiniz gibi, veri tabanı son kullanıcı verilerini barındıran ve yöneten bir bilgisayar yapısıdır. Bir veritabanı uzmanının ilk görevlerinden biri, geçerli ve zamanında bilgi elde etmek için son kullanıcı verilerinin uygun şekilde yapılandırılmasını sağlamaktır. Bunun için iyi bir veri tabanı tasarımı şarttır.

1-4 Veritabanı Tasarımı Neden Önemlidir?

Elektronik tablolar ve masaüstü veritabanı programları gibi kişisel üretkenlik araçlarının kullanımıyla birlikte ortaya çıkan bir sorun, kullanıcıların genellikle uygun veri modelleme ve veri tabanı tasarımı becerilerinden yoksun olmalarıdır. İnsanlar doğal olarak çevrelerindeki verilere "dar" bir açıdan bakarlar.

⁽²⁾www.internetlivestats.com/twitter-statistics/

⁽³⁾Josh Constine, "Facebook'un verileri ne kadar büyük? 2,5 milyar içerik parçası ve her gün 500+ terabayt veri alınıyor," Tech Crunch, 22 Ağustos 2012, <https://techcrunch.com/2012/08/22/how-big-is-facebook-data-2-5-billion-pieces-of-content-and-500-terabytes-ingested-every-day/>

Örneğin, bir öğrencinin ders programını düşünün. Program muhtemelen öğrencinin kimlik numarasını ve adını, sınıf kodunu, sınıf tanımını, sınıf kredi saatlerini, sınıf eğitmeninin adını, sınıf toplantı gün ve saatlerini ve sınıf oda numarasını içerir. Öğrencinin zihninde bu çeşitli veri öğeleri tek bir birim oluşturur. Eğer bir öğrenci organizasyonu üyelerinin programlarının kaydını tutmak isterse, son kullanıcı program bilgilerini saklamak için bir elektronik tablo hazırlayabilir. Öğrenci masaüstü veritabanları dünyasına girse bile, program verilerine bakışını taklit eden tek bir tablodan oluşan bir yapı oluşturması muhtemeldir. İlerleyen bölümlerde öğreneceğiniz gibi, bu tür dar bir veri görünümünü iki boyutlu tek bir tablo yapısına dönüştürmek kötü bir veritabanı tasarım tercihidir.

Veritabanı tasarımı, son kullanıcı verilerini depolamak ve yönetmek için kullanılacak veritabanı yapısının tasarımına odaklanan faaliyetleri ifade eder. Tüm kullanıcı gereksinimlerini karşılayan bir veritabanı kendiliğinden oluşmaz; yapısı dikkatlice tasarlanmalıdır. Aslında veritabanı tasarımı, veritabanlarıyla çalışmanın o kadar önemli bir yönüdür ki bu kitabın büyük bir kısmı iyi veritabanı tasarım tekniklerinin geliştirilmesine ayrılmıştır. İyi bir VTYS bile kötü tasarlanmış bir veritabanı ile kötü performans gösterecektir.

Veri, bir kuruluşun en değerli varlıklarından biridir. Müşteriler, çalışanlar, siparişler ve makbuzlar hakkındaki verilerin tümü bir şirketin varlığı için hayati önem taşır. Temel büyüme ve performans göstergelerinin izlenmesi de gelecekteki başarıyı sağlamak için stratejik ve taktiksel planlar açısından hayati önem taşır; bu nedenle bir kuruluşun verileri hafife alınmamalı veya dikkatsizce kullanılmamalıdır. Verilerin uygun şekilde kullanılmasını ve şirkete en fazla faydayı sağlayacak şekilde yararlanılmasını sağlamak için kapsamlı planlama, şirketin finansal kaynaklarından en iyi şekilde yararlanmasını sağlamak için uygun finansal planlama kadar önemlidir.

Mevcut nesil VTYS'lerin kullanımı kolay olduğundan, talihsiz bir yan etki de bilgisayar konusunda bilgili birçok iş kullanıcısının işlevsel bir veritabanı oluşturma becerileri konusunda yanlış bir güven duygusu kazanmasıdır. Bu kullanıcılar veritabanı nesnelerinin oluşturulmasında etkili bir şekilde yol alabilirler, ancak veritabanı tasarımını doğru bir şekilde anlamadan, sistemin iş gerçeklerine karşılık gelen verileri doğru bir şekilde depolamasını engelleyen kusurlu, aşırı basitleştirilmiş yapılar üretme eğilimindedirler, bu da veriler geri alındığında eksik veya hatalı sonuçlar üretir. Şekil 1.5'te gösterilen ve bir kuruluşun çalışanları ve onların becerileri hakkında kayıt tutma çabalarını gösteren verileri ele alalım. Bazı çalışanlar herhangi bir beceride sertifika testini geçmemişken, diğerleri birkaç beceride sertifika almıştır. Bazı sertifikalı beceriler birkaç çalışan tarafından paylaşılırken, diğer becerilerde bu sertifikalara sahip hiçbir çalışan yoktur.

veritabanı tasarımı

Veritabanı yapısının tanımlanmasını sağlayan ve veritabanı bileşenlerini belirleyen süreç Veritabanı yaşam döngüsünün ikinci aşaması.

Şekil 1.5 Kötü Bir Tasarımda Çalışan Becerileri Sertifikasyonu

Neden 9. ve 10. satırlarda boşluklar var?

Çalışanların alfabetik bir listesi nasıl oluşturulur?

Temel Veritabanı Manipülasyonu konusunda kaç çalışanın sertifikalı olduğunu nasıl hesaplayabilirim?

Temel Veritabanı Manipülasyonu, Temel DB Manipülasyonu ile aynı mıdır?

Bir çalışan dördüncü bir sertifika alırsa ne olur?
Ekleyelim mi
Başka bir sütun mu?

ID	ENum	Name	Title	HireDate	Skill1	Skill1Date	Skill2	Skill2Date	Skill3	Skill3Date
1	02345	Brian Oates	DBA	2/14/2001	Basic Database Management	2/14/2008	Advanced Database Management	2/14/2011	Basic Web Design	8/9/2009
2	08273	Marco Bienz	Analyst	7/28/2012	Basic Web Design	3/8/2015	Advance Process Modeling	8/19/2018		
3	06234	Jasmine Patel	Programmer	8/10/2011	Basic Web Design	8/10/2013	Advanced C# programming	8/10/2013	Basic DB manipulation	1/29/2018
4	03373	Franklin Johnson, Jr.	Purchasing Agent	3/15/2008	Advanced Spreadsheets	6/20/2017				
5	13567	Almond, Robert	Analyst	9/30/2018	Basic Process Modeling	9/30/2020	Basic Database Design	5/23/2021		
6	10282	Richardson, Amanda	Clerk	4/11/2017						
7	09382	Susan Mathis	Database Programmer	8/2/2016	Basic DB Design	8/2/2016	Basic Database Manipulation	8/2/2018	Advanced DB Manipulation	5/1/2019
8	14311	Duong, Lee	Programmer	9/1/2020	Basic Web Design	9/1/2022				
9					Master Database Programming					
10					Basic Spreadsheets					
11	09002	Wade Gaither	Clerk	5/20/2016	Advanced Spreadsheets	5/16/2019	Basic Web Design	5/16/2019		
12	13383	Raymond F. Matthews	Programmer	3/12/2018	Basic C# Programming	3/12/2020				
13	09283	Chavez, Juan	Clerk	7/4/2016						
14	04893	Patricia Richards	DBA	6/11/2010	Advanced Database Management	6/11/2012	Advanced Database Manipulation	9/20/2018		
15	13932	Lee, Megan	Programmer	9/29/2019						

Verilerin bu şekilde depolanmasına bağlı olarak, aşağıdaki sorunlara dikkat edin:

- Çalışanların soyadlarına göre alfabetik bir listesini çıkarmak imkansız olmasa da zor olacaktır.
- Temel Veritabanı Manipülasyonu'nda kaç çalışanın sertifikalı olduğunu belirlemek için, Beceri1'de kaydedilen bu sertifikaların sayısını sayan ve bir değişkene yerleştiren bir programa ihtiyacınız olacaktır. Daha sonra Beceri2'deki bu sertifikaların sayısı hesaplanabilir ve değişkene eklenebilir. Son olarak, Beceri3'teki bu sertifikaların sayısı hesaplanabilir ve toplamı üretmek için değişkene eklenebilir.
- Bir becerinin adını o beceride sertifikası olan her çalışanla birlikte gereksiz yere saklarsanız, adın farklı çalışanlar için farklı şekilde yazılması riskiyle karşı karşıya kalırsınız. Örneğin, *Temel Veritabanı Manipülasyonu* becerisi Şekil 1.5'te en az bir çalışan için *Temel DB Manipülasyonu* olarak da girilmiştir, bu da sertifikaya sahip çalışanların doğru bir şekilde sayılmasını zorlaştırır.
- Bir çalışan dördüncü bir beceride sertifika aldığında tabloya daha fazla sütun eklenerek veritabanının yapısının değiştirilmesi gerekecektir. Bir çalışan beşinci bir beceride sertifikalandırılırsa tekrar değiştirilmesi gerekecektir.

Bu zayıf tasarımı, verilerin birbiriyle ilişkili üç tabloya ayrıştırılmasıyla tasarımın iyileştirildiği Şekil 1.6'da gösterilen tasarımla karşılaştırın. Bu tablolar, aşağıdaki tablolara aynı verileri içerir

Şekil 1.6 İyi Bir Tasarımda Çalışan Becerileri Sertifikasyonu

Tablo adı: EMPLOYEE

Employee_ID	Employee_FName	Employee_LName	Employee_HireDate	Employee_Title
02345	Brian	Oates	2/14/2003	DBA
03373	Franklin	Johnson	3/15/2010	Purchasing Agent
04893	Patricia	Richards	6/11/2012	DBA
06234	Jasmine	Patel	8/10/2013	Programmer
08273	Marco	Bienz	7/28/2014	Analyst
09002	Wade	Gaither	5/20/2018	Clerk
09283	Juan	Chavez	7/4/2018	Clerk
09382	Susan	Mathis	8/2/2018	Database Programmer
10282	Amanda	Richardson	4/11/2019	Clerk
13383	Raymond	Matthews	3/12/2020	Programmer
13567	Robert	Almond	9/30/2020	Analyst
13932	Megan	Lee	9/29/2021	Programmer
14311	Lee	Duong	9/1/2022	Programmer

Tablo adı: SKILL

Skill_ID	Skill_Name	Skill_Description
100	Basic Database Management	Create and manage database user accounts.
110	Basic Web Design	Create and maintain HTML and CSS documents.
120	Advanced Spreadsheets	Use of advanced functions, user-defined functions, and macroing.
130	Basic Process Modeling	Create core business process models using standard libraries.
140	Basic Database Design	Create simple data models.
150	Master Database Programming	Create integrated trigger and procedure packages for a distributed environment.
160	Basic Spreadsheets	Create single tab worksheets with basic formulas
170	Basic C# Programming	Create single-tier data aware modules.
180	Advanced Database Management	Manage Database Server Clusters.
190	Advance Process Modeling	Evaluate and Redesign cross-functional internal and external business processes.
200	Advanced C# Programming	Create multi-tier applications using multi-threading
210	Basic Database Manipulation	Create simple data retrieval and manipulation statements in SQL.
220	Advanced Database Manipulation	Use of advanced data manipulation methods for multi-table inserts, set operations, and correlated subqueries.

Veritabanı adı: Ch01_Text

Tablo adı: CERTIFIED

Employee_ID	Skill_ID	Certified_Date
02345	100	2/14/2008
02345	110	8/9/2009
02345	180	2/14/2011
03373	120	6/20/2017
04893	180	6/11/2012
04893	220	9/20/2018
06234	110	8/10/2013
06234	200	8/10/2013
06234	210	1/29/2018
08273	110	3/8/2015
08273	190	8/19/2018
09002	110	5/16/2019
09002	120	5/16/2019
09382	140	8/2/2018
09382	210	8/2/2018
09382	220	5/1/2019
13383	170	3/12/2020
13567	130	9/30/2020
13567	140	5/23/2021
14311	110	9/1/2022