

VERİTABANI TASARIMI

Öğretim Görevlisi A. Berika VAROL MALKOÇOĞLU

İçindekiler

- Join ifadeleri
- Veritabanı güvenliği
- Savunma yöntemleri

Join

- Veritabanı tasarımı ve tablo oluşturma sırasında tablolar çeşitli parçalara ayrılarak veri tekrarının önüne geçilmiş olur.
- Parçalara ayrılan tablolar tek başına bir anlam ifade etmeyebilir.
- Parçaların anlamlı olabilmesi için tabloların birleştirme işlemine tabi tutulması gerekir.
- SQL join ifadeleri kullanılarak tablolar **sanal** olarak birleştirilir.
- JOIN ifadeleri iki veya daha fazla tabloyu birleştirmek için kullanılır.

Join

- Tabloları birleştirmek için;





```
SELECT tablo_A.sutun_adi FROM tablo_A  
{INNER JOIN | LEFT JOIN | RIGHT JOIN}  
tablo_B ON  
tablo_A.sutun_adi = tablo_B.sutun_adi;
```

Join Türleri

- **INNER JOIN:** Birden çok tablodaki verileri birleştirmek için kullanılır.
- **LEFT (OUTER) JOIN:** Birden çok tablodaki eşleşen kayıtlar ve eşleşmeyen sol kayıtlar için kullanılır.
- **RIGHT (OUTER) JOIN:** Birden çok tablodaki eşleşen kayıtlar ve eşleşmeyen sağ kayıtlar için kullanılır.
- **FULL OUTER JOIN:** Birden çok tablodaki eşleşen kayıtlar ve eşleşmeyen sağ ve sol kayıtlar için kullanılır. LEFT ve RIGHT JOIN birleşimidir.



Inner Join

```
8 • SELECT muster_i.ad, muster_i.soyad, muster_i.urun_adet, urun.ad,urun.fiyat as "Bir ürün için"  
9 FROM eczane.muster_i  
10 INNER JOIN urun ON muster_i.urun = urun.urun_no;  
11
```

<					
Result Grid   Filter Rows: <input type="text"/> Export:  Wrap Cell Content: 					
	ad	soyad	urun_adet	ad	Bir ürün için
▶	Aslı	Can	2	Aspirin	3.98
	Ayşe	Al	1	Katarin	7.99
	Fatma	Saygın	1	Bepanthen	17.49



Inner Join

```
13 • SELECT kitap.kitap_ad, yazar.yazar_ad,yazar.yazar_soyad, kategori.kategori_ad
14 FROM kitaplik.kitap
15 INNER JOIN kitaplik.yazar ON kitap.yazar_id = yazar.yazar_id
16 INNER JOIN kitaplik.kategori ON kitap.kategori_id = kategori.kategori_id;
```

<				
Result Grid				
Filter Rows: <input type="text"/>				
Export: 				
Wrap Cell Content: 				
	kitap_ad	yazar_ad	yazar_soyad	kategori_ad
▶	kürk mantolu madonna	sabahattin	ali	türk klasikleri
	kuyucaklı yusuf	sabahattin	ali	türk klasikleri
	kağrı	sabahattin	ali	roman
	benim adım kırmızı	orhan	pamuk	roman
	kar	orhan	pamuk	roman
	kırmızı saçlı kadın	orhan	pamuk	roman
	masumiyet müzesi	orhan	pamuk	roman
	XXXX KİTABI	orhan	pamuk	fantastik
	sevgi duvarı	can	yücel	şiiir
	can yücel'in şiirleri	can	yücel	şiiir
	adı: aylin	ayşe	kulin	roman
	sevdalinka	ayşe	kulin	roman
	tutsak güneş	ayşe	kulin	roman
	veda	ayşe	kulin	roman
	sinekli bakkal	halide edip	adıvar	türk klasikleri
	ateşten gömlek	halide edip	adıvar	türk klasikleri

Inner Join

```
21 • select y.*,k.kitap_ad from kitaplik.yazar y inner join kitaplik.kitap k on k.yazar_id=y.yazar_id;
```




Result Grid |  Filter Rows: | Export:  | Wrap Cell Content: 

	yazar_id	yazar_ad	yazar_soyad	kitap_ad
	10	ahmet	ümit	masal içinde masal
	10	ahmet	ümit	istanbul hatırası
	11	yaşar	kemal	ince memed
	11	yaşar	kemal	kuşlar da gitti
	11	yaşar	kemal	ölmez otu
	11	yaşar	kemal	sarı sıcak

Inner Join

```
30
31 • SELECT ci.Name, ci.Population
32 FROM world.country co
33 INNER JOIN world.city ci ON co.Code=ci.CountryCode Where co.code='TUR';
34
```

<




Result Grid |  Filter Rows: | Export:  | Wrap Cell Content: 

	Name	Population
▶	Istanbul	8787958
	Ankara	3038159
	Izmir	2130359
	Adana	1131198
	Bursa	1095842
	Gaziantep	789056
	Konya	628364
	Mersin (İAşel)	587212
	Antalya	564914
	Diyarbakir	479884
	Kayseri	475657
	Eskisehir	470781
	Sanliurfa	405905
	Samsun	339871
	Malatya	330312
	Gebze	264170
	Denizli	253848
	Sivas	246642
	Erzurum	246535
	Tarsus	246206
	Kahramanma...	245772

Left Join

```
21 • select y.*,k.kitap_ad from kitaplik.yazar y left join kitaplik.kitap k on k.yazar_id=y.yazar_id;
```




<

Result Grid |  Filter Rows: | Export:  | Wrap Cell Content: 

	yazar_id	yazar_ad	yazar_soyad	kitap_ad
	10	ahmet	ümit	istanbul hatrası
	11	yaşar	kemal	ince memed
	11	yaşar	kemal	kuşlar da gitti
	11	yaşar	kemal	ölmez otu
	11	yaşar	kemal	sarı sıcak
	12	alan	moore	HULL

Left Join

```
29 • SELECT c.calisan_no, c.ad, c.soyad, m.* FROM eczane.calisan c LEFT JOIN musteri m ON c.calisan_no = m.calisan;
```

Result Grid  Filter Rows: <input type="text"/> Export:  Wrap Cell Content: 									
	calisan_no	ad	soyad	musteri_no	ad	soyad	urun	urun_adet	calisan
▶	10	Ali	Öz	100	Aslı	Can	1	2	10
	10	Ali	Öz	104	NULL	NULL	5	1	10
	11	Ayşe	İz	101	Ayşe	Al	4	1	11
	11	Ayşe	İz	102	Fatma	Saygin	5	1	11
	12	Burak	Ulutaş	103	Emine	Yılmaz	6	1	12
	13	Ceren	Çubuk	NULL	NULL	NULL	NULL	NULL	NULL
	14	Zeynep	Güven	NULL	NULL	NULL	NULL	NULL	NULL

Right Join

```
19 • select k.*,y.yazar_ad,y.yazar_soyad from kitaplik.kitap k right join kitaplik.yazar y on k.yazar_id=y.yazar_id;  
20
```

<						
Result Grid						
Filter Rows: <input type="text"/> Export: Wrap Cell Content:						
	kitap_id	kitap_ad	yazar_id	kategori_id	yazar_ad	yazar_soyad
	29	istanbul hatrası	10	6	ahmet	ümit
	10	ince memed	11	1	yaşar	kemal
	16	kuşlar da gitti	11	1	yaşar	kemal
	30	ölmez otu	11	1	yaşar	kemal
	35	sarı sıcak	11	1	yaşar	kemal
	NULL	NULL	NULL	NULL	alan	moore

Right Join

```
23 • SELECT m.*, u.ad, u.fiyat FROM eczane.musteri m RIGHT JOIN urun u ON M.urun = u.urun_no;
```

<div><div><</div><div>Result Grid</div><div><div></div><div></div></div><div>Filter Rows: <input type="text"/></div><div>Export: <div></div></div><div>Wrap Cell Content: <div></div></div></div>								
	musteri_no	ad	soyad	urun	urun_adet	calisan	ad	fiyat
▶	100	Aslı	Can	1	2	10	Aspirin	3.98
	NULL	NULL	NULL	NULL	NULL	NULL	Parol	8.18
	101	Ayşe	Al	4	1	11	Katarin	7.99
	102	Fatma	Saygin	5	1	11	Bepanthen	17.49
	104	NULL	NULL	5	1	10	Bepanthen	17.49
	103	Emine	Yilmaz	6	1	12	aferin	6.89
	NULL	NULL	NULL	NULL	NULL	NULL	majezik	18.5
	NULL	NULL	NULL	NULL	NULL	NULL	coraspin	13.69

Veri Tabanının Ele Geçirilmesine Neden Olan Tehditler

Kullanıcı İzinleri

- Veritabanındaki kullanıcı izinlerinin iyi bir kontrol mekanizması ile yönetilmemesi
- Kullanıcılara ihtiyaçlarının üzerinde haklar verilmesi,
- Bu kullanıcıların
 - Hesap bilgilerinin çalınması,
 - Bilgizilikten/dikkatsizlikten kaynaklanan hataları veya
 - Kötü niyetli davranışları sonucunda sistemin kolayca ele geçirilmesine neden olacaktır.

SQL Enjeksiyon

- Veri tabanlarına yönelik gerçekleştirilen en yaygın ataklardan biridir.
- SQL enjeksiyon saldırıları veri tabanı veya veri tabanına ara yüz olarak çalışan web uygulamalarını hedef alarak sisteme ulaşmaya çalışmaktadır.
- SQL enjeksiyon atakları saldırganın veri tabanı üzerindeki hassas bilgilere erişmesini, kullanıcı haklarını yükseltmesini ve database işletim sistemi komutlarına ve veri tabanının kendisine ulaşması gibi tehlikeli işlemleri yapmasına neden olabilmektedir.

Zararlı Yazılım Bulaştırma

- Çalışanın;
 - Mail hesabına gönderilen oltalama maili,
 - Zararlı usb belleğin bilgisayarına takılması veya
 - Bilinmeyen sitelerden program indirmesi ile zararlı yazılımın veritabanı kullanıcısının bilgisayarı üzerinden veri tabanı sistemine bulaşması olabilir.

Raporlama ve Yedekleme Sorunu

- Otomatize edilmiş olay kayıtları veri tabanı sistemlerinin güvenliği için büyük önem taşımaktadır.
- Kayıtlar düzenli raporlanmadığı takdirde çeşitli sorunlara yol açacaktır.
- Yada olan sorunları çözümlemeyi engelleyebilir.
- Genelde canlı sistemler daha iyi korunmaya alınmaktadır.
- Ancak yedekleme dosyalarının ve yedekleme ortamının güvenliği de canlı sistemler kadar önemlidir.
- Saldırganlar şifrelenmemiş bir yedekleme dosyası üzerinden sisteme sızabilir.

DOS Saldırısı

- Hizmet engelleme saldırısı olarak adlandırılan DOS atakları ağ cihazlarına veya verilere erişimi engelleyen bir saldırı türüdür.
- Veri tabanı sistemlerine karşı gerçekleştirilen DOS saldırılarında, sistemde büyük oranda bir kaynak kullanımına neden olan arama yapma ve istek gönderme işlemleri gerçekleştirilerek sunucuya aşırı yük bindirilmesi ve hizmet veremez hale getirilmesi hedeflenir.

Veri Tabanı Güvenliđi Yöntemleri

Veri Tabanı Güvenliği Yöntemleri

- Verilerin sınıflandırılması
 - Bu yöntemle asıl güvene alınması gereken hassas verilerin daha kolay tanımlanabilmesinin sağlar.
 - Yalnızca veritabanını değil, verileri korumayı hedefler.
- Üçüncü parti bir denetim yazılımı ve ağ cihazı kullanılması
 - Hem koruma hem de monitoring servisi veren bu cihazlar veri tabanı sunucusundan ayrı olarak kuruldukları için sunucuya ek yük bindirmezler.
 - Kullanıcının gereksinimlerine göre özelleştirilebilen bu çözümler birçok özelliğe sahip olmasının yanı sıra aynı zamanda yüksek oranda analiz sonucu alabilmek için detaylı denetim yaparlar.

Veritabanı Güvenliği için;

- Durağan haldeki datanın şifrlenmesi kadar hareket halindeki datanın da güçlü kimlik doğrulama kontrolleri ile iletimi de büyük önem taşımaktadır.
- Durağan haldeki veriler için AES şifreleme algoritması önerilmektedir.
- Varsayılan kullanıcı şifrelerini değiştirmek.
- Kullanılmayan kullanıcı hesaplarını kilitlemek.
- Daha güçlü şifreler uygulamak.

Veritabanı Güvenliği için;

- Şifrenin ömrü kontrol edilmeli, kullanım süresi dolduğunda uyarı mesajı verilmeli.
- Eski şifrelerin yeniden kullanılmasını engelleyecek bir kısıtlama getirilmeli.
- Analiz edilip güvenliğinin sağlanması gereken veri tabanını tespit etmek.
- Veri tabanını zafiyetler, yanlış yapılandırmalar ve kritik değişiklikler için taramak.

Veritabanı Güvenliği için;

- Dataların sınıflandırılması esas alınarak yüksek önemdeki açıklıkların öncelik verilerek kapatılması.
- Zafiyetler için uygun bir düzeltme tespit etmek, zafiyetleri kapatmak ve sistemi güncellemek için yamaları uygulamak
- Zafiyet düzeltme işlemlerinin etkisini ölçmek için düzeltmeden sonra denetim işlemi tekrarlanmalıdır ve bu döngü periyodik olarak tekrarlanmalıdır.
- Kuruluşun varlıklarının yetkisiz veya farkında olmadan değiştirilme ya da kötüye kullanılma fırsatlarını azaltmak için, görevler ve sorumluluk alanları ayrılmalıdır.

Veritabanı Güvenliği için;

- İhtiyacınız olmayan modülleri ve hizmetleri ve/veya paketleri çıkartılmalı.
- Veritabanları için doküman onaylı yapılandırma referansı kullanılmalıdır.
- Girdi/Çıktı kontrolleri, kullanıcı girdisinde problem yaratabilecek SQL'e özel karakterler var mı? Girilen veri, veri tipine uyumluluk gösteriyor mu?
- Girilen veri limitler dahilinde mi? (Taşma hataları)

Veritabanı Güvenliği için;

- Gerekli hata denetimi fonksiyonlarıyla program çıktısı seçenekleri kontrol edilmelidir.
- Uygulamalar ve veritabanı, özellikle Web uygulaması arasındaki oturumları şifrelenmeli.
- Veritabanı güncellemelerini kontrol edilmeli.
- Uygulama tasarımı ve programlanması sırasında güvenlik ilkeleri ve oluşabilecek problemler dikkate alınmalıdır.
- Sistem üzerinde bilgi toplanmasına neden olabilecek başlık bilgileri ve hata mesajlarının açık bırakılmaması gereklidir.

Veritabanı Güvenliği için;

- Fiziksel bütünlüğü sağlamak için hesap verebilirlik ve denetim kontrolleri gereklidir.
- Veri tabanı trafiğini analiz eden, çalıştırmayı algılayan, engelleyen, yetkisiz sorgular ve SQL enjeksiyonları tespit eden gerçek zamanlı çözümler araştırılmalı.

Veritabanı Güvenliği için;

- Bu algılanan tehditler hakkında uyarılar ve raporları yönetici belirlenen kişilere e-posta veya anlık mesajlaşma ile uyarı gönderilmeli.
- Güncel olmayan kimlik doğrulama sistemlerine ait istemcilerin bağlantıları engellenmeli.
- Veri kaybolduğunda veya çalındığında ne yapacağınıza dair bir plan yapılmalı.
- Kullanıcıların sadece belirttikleri sistem üzerinden bağlanmalı.
- Kullanıcıyı yönetmek (yaratmak, silmek, denetlemek) için politikalar oluşturulmalı.
- Kullanıcı izleme politikasını belirlenmeli.