**What is JOIN?**

Bu bölümde size JOIN deyimlerini tanıtacağız. JOIN yan tümcesi, iki veya daha fazla tabloyu tek bir tabloda birleştirmek için kullanılır. Bu işlem tablolar arasında ilgili bir sütuna göre gerçekleştirilir.

Şimdi aşağıdaki iki tabloyu inceleyelim. "Departmanlar" tablosunda gereksiz yere çalışanların isimleri yazılmaz, sadece çalışanların kimlik numaraları yazılır.

"Çalışanlar" tablosuna bakalım:

tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

O halde "departmanlar" tablosuna bakalım:

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

"Departmanlar" tablosuna bakıldığında hangi çalışanın hangi departmanda çalıştığını anlamak kolay değildir, çünkü bu tablo çalışanların isimlerini içermemektedir. Bunun için iki tablo birleştirilmeli ve "departmanlar" tablosundaki her bir çalışan kimliği için "çalışan" tablosuna bakılarak o çalışanın adı ve soyadı bulunmalıdır. "emp\_id", yukarıdaki iki tablo arasındaki ilişkiyi oluşturacak sütundur.

Temel olarak beş tür JOIN vardır: INNER JOIN, LEFT JOIN, RIGHT JOIN, FULL OUTER JOIN ve CROSS JOIN. Özel bir durum olarak, bir tablo kendisine katılabilir ve bu türe SELF JOIN adı verilir.

INNER JOIN: Her iki tablodaki ortak kayıtları döndürür.

LEFT JOIN: Soldaki tablodaki tüm kayıtları ve sağdaki tablodaki eşleşen kayıtları döndürür.

RIGHT JOIN: Sağdaki tablodaki tüm kayıtları ve soldaki tablodaki eşleşen kayıtları döndürür.

FULL OUTER JOIN: Hem sol hem de sağ tabloların tüm kayıtlarını döndürür.

CROSS JOIN: Birleştirilmiş tablolardaki kayıtların Kartezyen çarpımını döndürür.

SELF JOIN: Bir tablonun kendisine katılması.

**Inner JOIN**

INNER JOIN, en yaygın JOIN türüdür. İki veya daha fazla tablodan ortak sütunlardaki değerleri temel alarak yeni bir sonuç tablosu oluşturur. INNER JOIN, yalnızca belirtilen birleştirme koşullarını karşılayan birleştirilmiş satırları içeren bir tablo döndürür.



Bu sözdiziminde,

columns: tablo\_A veya tablo\_B'den sütun adları.

table\_A, table\_B: Birleştirilmiş tabloların adları.

join\_conditions: Birleştirilmiş her satır çifti için değerlendirilecek koşulları belirtir.

Bir birleştirme tamamlama şu.sütun alır: table\_A.column = table\_B.column. Bu ifadedeki operatörlerdendir (=), herhangi bir olası sürümde kullanılabilir.

**💡İpuçları:**

INNER JOIN koşulunu belirtmek için ON anahtar sözcüğünü not edin.

AND veya OR deyimleri kullanılarak çoklu birleştirme koşulları yazılabilir.

Ayrıca, INNER JOIN yan tümcesi kullanılarak üç veya daha fazla tablo birleştirilebilir. Üç veya daha fazla tabloyu birleştirmek için kullanılan sözdizimi aşağıdaki gibidir:

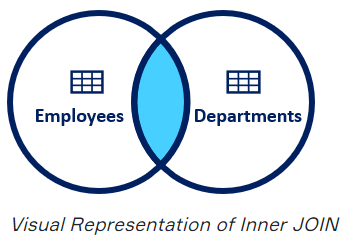
metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Örnek**

"Çalışanlar" tablosuna ve "departmanlar" tablosuna katılmak istediğinizi varsayalım. Bu örnekte, iki tablodaki emp\_id sütunlarına dayalı bir INNER JOIN oluşturulmuştur:

INNER JOIN'in nasıl çalıştığını kolayca anlamak için aşağıdaki görsel açıklamayı kullanabiliriz. İki tablonun kesişimi, eşleşen satırları temsil eder.



**Sözdizimi**

INNER JOIN anahtar sözcüğü, sütunlar arasında bir eşleşme olduğu sürece hem "çalışanlar" hem de "departmanlar" tablolarından tüm satırları seçer. "Çalışanlar" tablosunda "departmanlarda" eşleşmesi olmayan kayıtlar varsa bu kayıtlar çıktıda gösterilmez.

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

Çıktıdan da görebileceğiniz gibi, "çalışanlar" tablosundaki yedi çalışanın "departman" tablosunda eşleşmeleri var. Kalan üç çalışan çıktıda listelenmez. Gördüğünüz gibi çalışan kimliği 17679 hem tablolarda hem de eşleşmelerde olduğundan çıktı tablomuza dahil edilmiştir. Öte yandan, çalışan kimliği 49714, çalışan tablosunda yer alıyor ancak departman tablosunda yer almıyor ve eşleşme yok. Bu nedenle, çalışan KIMLIĞI 49714 çıktı tablosuna dahil edilmez.

**Left JOIN**

Bu JOIN deyiminde, sorguda soldaki tablonun tüm kayıtları ve sağdaki tablonun ortak kayıtları döndürülür. JOIN işlemi sırasında sağdaki tabloda eşleşen satır bulunamazsa, bu değerler NULL olarak atanır.

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

Bu sözdiziminde,

columns: tablo\_A veya tablo\_B'den sütun adları.

table\_A, table\_B: Birleştirilmiş tabloların adları.

join\_conditions: Birleştirilmiş her satır çifti için değerlendirilecek koşulları belirtir.

LEFT JOIN ve LEFT OUTER JOIN anahtar sözcükleri tamamen aynıdır. OUTER anahtar sözcüğü isteğe bağlıdır.

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

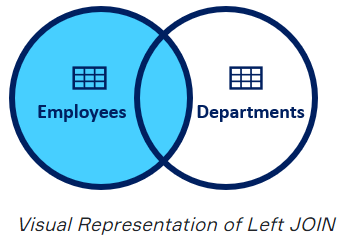
**💡İpuçları:**

Belirli bir satır için eşleşme bulunamazsa, NULL döndürülür.

**Örnek**

"Çalışanlar" tablosuna ve "departmanlar" tablosuna katılmak istediğinizi varsayalım. Bu örnekte, iki tablodaki emp\_id sütunlarına dayanan bir LEFT JOIN oluşturulmuştur:

LEFT JOIN'in nasıl çalıştığını kolayca anlamak için aşağıdaki görsel açıklamayı kullanabiliriz. Gölgeli alan, "çalışanlar" tablosundaki tüm satırları ve "departmanlar" tablosundaki tüm eşleşen satırları temsil eder.



metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

LEFT JOIN işlemi sonucunda "personel" tablosundan on satırın tamamı ve "departments" tablosundaki eşleşen tüm satırlar döndürülür. Doğru tabloyla eşleşmeyen satırlar null olarak döndürülür.

**Right JOIN**

Basit bir ifadeyle, RIGHT JOIN, LEFT JOIN'in tersidir. Bu birleştirme ifadesinde, sağdaki tablonun tüm kayıtları ve soldaki tablonun ortak kayıtları sorguda döndürülür. JOIN işlemi sırasında soldaki tabloda eşleşen satır bulunamazsa, bu değerler NULL olarak atanır.

İşte RIGHT JOIN yan tümcesinin sözdizimi:

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

Bu sözdiziminde,

columns: tablo\_A veya tablo\_B'den sütun adları.

table\_A, table\_B: Birleştirilmiş tabloların adları.

join\_conditions: Birleştirilmiş her satır çifti için değerlendirilecek koşulları belirtir.

RIGHT JOIN ve RIGHT OUTER JOIN anahtar sözcükleri tamamen aynıdır. OUTER anahtar sözcüğü isteğe bağlıdır.

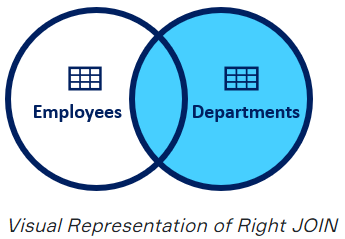
**💡İpuçları:**

Belirli bir satır için eşleşme bulunamazsa, NULL döndürülür.

**Örnek**

"Çalışanlar" tablosuna ve "departmanlar" tablosuna katılmak istediğinizi varsayalım. Bu örnekte, iki tablodaki emp\_id sütunlarına dayanan bir RIGHT JOIN oluşturulmuştur:

RIGHT JOIN'in nasıl çalıştığını kolayca anlamak için aşağıdaki görsel açıklamayı kullanabiliriz. Gölgeli alan, "çalışanlar" tablosundaki tüm satırları ve "departmanlar" tablosundaki tüm eşleşen satırları temsil eder.



metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin, tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

RIGHT JOIN uygulandıktan sonra, "çalışanlar" tablosundaki emp\_id sütununda bir eşleşme olup olmadığına bakılmaksızın çıktı tablosu "departmanlar" tablosundaki tüm satırları içerir. Görüldüğü gibi "departmanlar" tablosundaki iki çalışanın kimliği, "çalışan" tablosundakilerle eşleşmiyor.

**Cross JOIN**

SQL'de, CROSS JOIN, ilk tablonun her satırını ikinci tablonun her satırıyla birleştirmek için kullanılır. Birleştirilmiş tablolardan satır kümelerinin Kartezyen çarpımını döndürdüğü için Kartezyen birleştirme olarak da bilinir.

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

bu sözdiziminde,

sütunlar: tablo\_A veya tablo\_B'den sütun adları.

table\_A, table\_B: Birleştirilmiş tabloların adları.

CROSS JOIN'in başka bir uygulaması da var. Burada, CROSS JOIN yan tümcesini kullanmazsınız. İşte sözdizimi:



Örnek

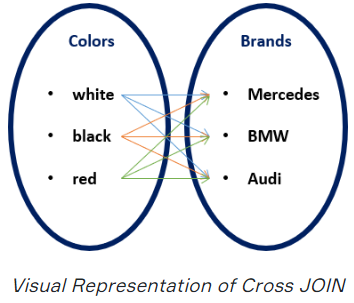
"Renkler" ve "markalar" adında iki tablomuz olduğunu varsayalım. Bu iki tablo satılacak otomobillerin renklerini ve markalarını belirtmek için oluşturulmuştur.

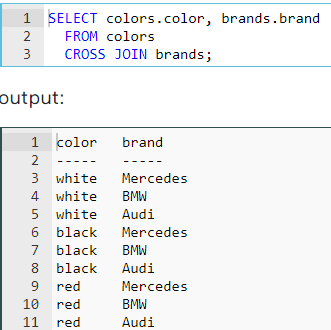
tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

İnsanlar tüm renk ve marka kombinasyonlarında araba satın almak istiyor. Dolayısıyla bu iki tabloyu birleştirerek yeni bir tablo oluşturacağız.





İki tabloya göre dokuz olası araba var. Üç marka ile çarpılan üç renk, dokuz olası sıra ile sonuçlanır.

**Self Join**

SELF JOIN, bir tablonun kendisine katılmasıdır. Bir tablonun kendisine katılması, tablonun her satırının kendisiyle ve tablonun diğer satırlarıyla birleştirilmesi anlamına gelir. SELF JOIN, aynı tablonun iki kopyasının birleşimi olarak tanımlanabilir. Bu, SQL komutuyla gerçekleştirilir, tablo aslında ikinci kez kopyalanmaz. Kendi kendine birleştirme oluşturmak için INNER JOIN veya LEFT JOIN kullanırız.

Satırları aynı tablodaki diğer satırlarla birleştiren bir sonuç kümesi oluşturmak için kendi kendine katılmayı kullanırsınız. Bir sorguda aynı tabloya birden fazla başvuramayacağınız için, self-join kullandığınızda tabloya farklı bir ad atamak için bir tablo takma adı kullanmanız gerekir.

Kendi kendine birleştirme, aynı tablodaki aynı veya farklı sütunların değerlerini karşılaştırır. Kendi kendine birleştirmede yalnızca bir tablo yer alır. Bir tabloda depolanan ebeveyn/alt ilişkisini sorgulamak veya değişen toplamları elde etmek için genellikle kendi kendine katılmayı kullanırsınız.

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

n bu sözdizimi,

columns: Tablodaki sütun adları.

tablo: Tablonun adı.

join\_conditions: Birleştirilmiş her satır çifti için değerlendirilecek koşulları belirtir.

**Örnek**

Bu örnekte, "çalışanlar" tablosunu değiştirdik ve "reports\_to" sütununu ekledik.

tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

Değiştirilen çalışanlar tablosu, yalnızca çalışan verilerini değil, aynı zamanda kuruluş için hiyerarşik verileri de depolar. "reports\_to" sütunu, çalışanların rapor vermesi gereken yöneticinin kimliğini içerir. SELF JOIN yöntemini kullanarak hiyerarşik ilişkilere sahip çalışanların isimlerinin olduğu tabloyu yazdırabiliriz.

Bir çalışan bir yöneticiye rapor veriyorsa, o çalışanın "reports\_to" sütunundaki değer, o yöneticinin "emp\_id" sütununun değerine eşittir. Bir çalışan kimseye rapor vermiyorsa yani yöneticisi yoksa "reports\_to" sütununun boş olması beklenir.

tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

SELF JOIN için ayrı bir ifade yoktur. INNER JOIN deyimi, "çalışanlar" tablosunu kendisiyle birleştirmek için kullanılır. INNER JOIN mekanizmasının çalışması nedeniyle çıktı tablosunda "reports\_to" sütununda null değerine sahip çalışan bulunmamaktadır.

**Full Outer JOIN**

FULL OUTER JOIN, her iki tablodaki tüm satırları döndürür. Çıktı tablosu, NULL veri içeren satırları içerecektir, bu nedenle hiçbir şey dışarıda bırakılmaz. NULL değerler, tablolardaki eksik verileri tespit etmek için önemli olabilir. FULL OUTER JOIN, katılmak istediğiniz tablolardaki satır sayısına bağlı olarak büyük veri kümeleri oluşturacaktır.

FULL OUTER JOIN yan tümcesinin sözdizimi şöyledir:

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

columns: Tablodaki sütun adları.

table\_A, table\_B: Birleştirilmiş tabloların adları.

join\_conditions: Birleştirilmiş her satır çifti için değerlendirilecek koşulları belirtir.

**💡İpuçları:**

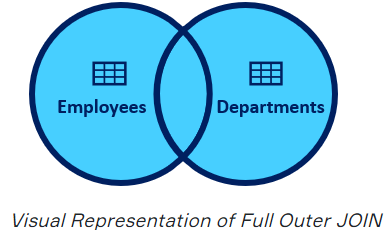
Tüm satırlar, eşleşen verilerden bağımsız olarak her iki tablodan da döndürülür.

Eşleşmeyen satırlar her iki tarafta NULL'larla doldurulur.

**Örnek**

"Çalışanlar" tablosuna ve "departmanlar" tablosuna katılmak istediğinizi varsayalım. Bu örnekte, iki tablodaki sütunları temel alan tam bir dış birleştirme oluşturulmuştur:

Aşağıdaki şekil FULL OUTER JOIN ile iki tablonun birleştirilmesini göstermektedir.



metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

FULL OUTER JOIN anahtar sözcüğü, diğer tablo eşleşsin veya eşleşmesin, her iki tablodaki tüm eşleşen kayıtları döndürür. Dolayısıyla, "çalışanlarda" "departmanlarda" eşleşmeyen satırlar varsa veya "departmanlarda" "çalışanlarda" eşleşmeyen satırlar varsa, bu satırlar da listelenecektir.

**Views**

Görünümler, birden çok veri türüne erişmek için kullanışlı araçlardır. Karmaşık sorgular görünümler içinde saklanabilir. Bu şekilde, sorguları her ihtiyacımız olduğunda yeniden oluşturmak yerine görünümü çağırabiliriz.

Bazen bir tablodaki bazı bilgilerin belirli kullanıcılara gizlenmesini isteriz. Görünüm bunun için uygun bir yoldur. Bu güvenlik açısından da önemlidir. Görünüm kullanılarak karmaşık yapılar sentezlenebilir ve son kullanıcı için kolay bir formatta sunulabilir.Görünümler, belirli bir proje veya raporlama kapsamında yalnızca ihtiyaç duyulan verilerin kullanılmasını sağlar.

Öte yandan, daha sonra kullanmayacağınız görünümleri geçici olarak oluşturarak yalnızca mevcut veritabanı bağlantısında görünür hale getirebiliriz. Bu şekilde, veritabanı bağlantısı kapatıldığında geçici görünüm otomatik olarak kaldırıldı. Böylece gereksiz depolama maliyetlerinden kurtulmuş oluyoruz.

**💡İpucu:**

Bir görünümün zaten var olup olmadığından emin değilseniz ve bunu saklı bir prosedürde veya işlevde çalıştırmak istiyorsanız, CREATE VIEW IF NOT EXISTS view\_name sözdizimi hata almanızı engeller.

**Örnekler**

Görünümler SQL Server'da salt okunurdur. Bu nedenle, görünüm aracılığıyla temel tablolardaki verileri güncellemek için INSERT, DELETE ve UPDATE deyimlerini kullanamazsınız. Görünümler için yalnızca CREATE , DROP ve ALTER ifadelerini kullanabilirsiniz. Bir görünümü güncellemek için önce görünümü bırakmalı ve ardından yeniden oluşturmalıyız.

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

Dikkat; drop view komutunu çalıştırdığımızda görünümleri kaldırıyor. Bu görünümün türetildiği temel tablolarda depolanan temel veriler değişmeden kalır. Bir kez bırakılan bir görünüm aynı adla yeniden oluşturulabilir.

Çalışan tablosu ile basit bir örnek görünüm oluşturalım ve ardından bırakalım.

tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

Yöneticilerden oluşan bir görünüm oluşturalım ve ardından bırakalım:

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

Yöneticilerden oluşan bir görünüm oluşturalım ve ardından bırakalım:

