

# SQL Tarih Formatlama

Veri tabanı programlama çalışmalarımnda karşıma çıkan ve zamanında epey zorlayan bir konu ile yeniden beraberiz. Başlıkta da kendini belli ediyor Microsoft Sql Server'da görüntüleme işlemi sırasında tarih formatlama;

Aşağıdaki kod parçasında bazı formatlama kodları ve çıktılarını görebilirsiniz:

---

## Ms Sql Server Tablo Boyutlarını Görme

Veri tabanı yöneticilerinin kontrol ettikleri konulardan biri de veri tabanı boyutlarındaki artışlardır. Aşırı artan bir veri tabanı boyutunda hangi tablonun buna sebep olduğunu bilmek sorunun çözümüne giden yolu da belirleyecektir.

Boyut artışları tablolardaki kayıt sayısının artışı ile hesaplanabilir ancak tablodaki kolon sayısındaki değişkenlikten dolayı doğru sonuç vermeyecektir. Aşağıdaki `sql` kodundan faydalanılarak veri tabanında yer alan tabloların KB ve MB cinsinden boyutları görülebilir.

---

# Sistem Nesneleri Dışında Veri Tabanından Her Şeyi Silme

Bazı veri tabanı çalışmalarında veri tabanında yer alan bütün nesnelere silinmesi gerekebiliyor. Burada akla hemen şu gelebilir: madem her şeyi siliyoruz, neden veri tabanını tamamen silip yeniden oluşturmuyoruz, tabi ki yetkiden dolayı yapamıyor olabiliriz, bu durumda tek tek silme ile uğraşmak yerine bunu bir kod parçası ile yapabiliriz.

---

# Sql Server Auto Increment (Otomatik Artan) Kolon Ayarlama

Veri tabanı çalışmalarında kullanılan en önemli başlıklardan biri de otomatik artan değerdir.

Örnek olarak sıra girilen kaydın sıra numarası veya Id kolonunun değeri otomatik artan değerde ayarlanması işlem kolaylığı sağlayacaktır.

Şimdi Sql Server üzerinde otomatik artan kolon (IDENTITY) nasıl

ayarlanır, beraberce inceleyelim.

Identity kolonu ayarlanırken göz önünde bulunması gereken en önemli unsur her tabloda en fazla bir identity kolonu olabileceğidir.



Otomatik Artan Kolon

Otomatik artan kolon tablo dizayn ekranında iki farklı şekilde yapılmaktadır.

- İlki kolonunun özelliklerinden “Identity Specification” özelliğini “Yes” olarak ayarlanır. Bu özelliğin altında yer alan iki özellik: “Identity Increment” artışın kaçar kaçar olacağını, “Identity Seed” ise ilk sayının değerini belirtir.
- İkinci şekilde de resmin sağ tarafında yer alan tablo özelliklerinden “Identity Column” seçeneğini seçmektir. Bu şekilde ayarlanması durumunda yukarıdaki iki değer varsayılan olarak 1 olarak ayarlanacaktır.

Kod ile belirtilmek istendiğinde de:

---

## Ms SQL Server'ın Gizli Gücü xp\_cmdshell

Sql Server dersini alırken sadece kullandığımız belli başlı 3 – 4 komut vardı, SELECT, UPDATE, DELETE... Ama hep dediğimiz bir şey daha vardı “Adamlar bu kadar büyük bir programı sırf SELECT için yazmış olamaz”, öyle ya bir yüklüyoruz bilgisayarda neredeyse 10GB yer kaplıyor.

Şimdi Sql Server'ın SELECT dışında yapabildiği binlerce işlemten birini beraber görelim.

**xp\_cmdshell**, SQL Server üzerinden cmd komut satırına komut göndermeyi ve çalıştırmaya imkan sağlayan stored prosedür. Bu özelliği sayesinde cmd ekranında yapabildiğimiz herşeyi:

- Windows Server üzerinde yeni kullanıcı açıpı buna full yetki verilip kuruluşumuzun her türlü gizli verisi ve işlemlerine ulaşılabilir
- "format X" komutunu göndererek Sql Server üzerinden X diskini formatlanabilir
- Windows Server içerisine her hangi bir .exe dosyası atılarak çalıştırılabilir, bu bir keylogger da olabilir

ve daha fazlası kısaca herşeyi.

xp\_cmdshell stored prosedürü varsayılan olarak güvenlik politikası olarak devre dışıdır. Prosedürü devreye almak için aşağıdaki kod çalıştırabilir veya arayüz üzerinde de işlem yapılabilir. Sql koduna ve hemen ardından arayüz adımlarına geçelim.

Alternatif olarak arayüz adımları:



xp\_cmdshell aktifleştirme

Bu işlemler sonucunda aktifleştirme gerçekleştirilir.

xp\_cmdshell komutu çalıştırıldığında dönen sonuç tek kolon ve text şeklinde olur. Çıktı vermeyen komutlar için başarılı olması durumunda 1, başarızlı olması durumunda 0 sonucu döner.

Şimdi de bazı örnekler yaparak çalışma şeklini görelim.

- Çıktı Veren Örnek  
Komut satırına 'DIR C:\' komutunu göndererek C:\ diski altında yer alan dosyaları göstereyim.



xp\_cmdshell ile C:\ diski altında yer alan dosyaların gösterilmesi

- Çıktı Vermeyen Örnek  
Komut satırın copy komutu göndererek C:\XP\Deneme.txt dosyasını aynı klasör altına Deneme2.txt adıyla kopyalayalım.

---

## Sql Server “Saving changes is not permitted...” Hatası



Saving changes is not permitted...

Sql Server üzerinde yer alan tablolarda yapısal deęişiklik yapıldığında Sql yukarıda ki gibi bir hata vermektedir, aslında bu hatadan çok bir güvenlik önlemidir. Bu durumu çözmek için şu yolu izleyebiliriz;

Sql Server Management Studio menüsünde **Tools > Options > Designers** penceresinde **“Prevent saving changes that require the table to be re-created”** seçeneğinin tıkını kaldırıp kaydedin. İşlem tamam.



---

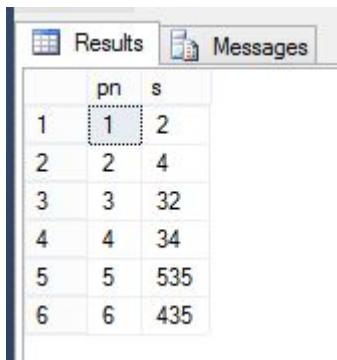
# SQL İle String Deęeri Paralama

Veri tabanı kolonlarında birden fazla bilgiyi karakter ile ayırarak tutmak tavsiye edilmese de bazı durumlarda bir hücrede birden fazla deęer tutulma efektif olabilmektedir.

Örneęin; **2|4|32|34|535|435** deęerlerini SQL ile işlemek için “|” karakterinden paralayıp işlemek gerekebilir. Bunun için gerekli olan SQL kodlarını içeren fonksiyon ařaęıdaki gibidir.

SQL Server üzerinde fonksiyon bu kod ile oluşturulduktan sonra kullanımı ve örnek ekran ıktısı ise řöyledir.

Örnek veri olarak yukarıdaki veriyi kullanacaęız:



The screenshot shows a SQL Server Results window with two tabs: 'Results' and 'Messages'. The 'Results' tab is active and displays a table with two columns, 'pn' and 's'. The table contains six rows of data. The first row has '1' in the 'pn' column and '2' in the 's' column. The second row has '2' in 'pn' and '4' in 's'. The third row has '3' in 'pn' and '32' in 's'. The fourth row has '4' in 'pn' and '34' in 's'. The fifth row has '5' in 'pn' and '535' in 's'. The sixth row has '6' in 'pn' and '435' in 's'. The cell containing '1' in the first row is selected.

	pn	s
1	1	2
2	2	4
3	3	32
4	4	34
5	5	535
6	6	435

---

# Bütün Veri Tabanı İçerisinde String Değer Arama

Merhaba, uzun bir aradan sonra tekrar yeni bir konu ve yeni bir heyecanla blogumuza dönüş yapıyoruz. Bu hafta ki konumuz Ms SQL Server üzerinde olacak.

Özellikle ters mühendislik işlemleri ile yapılan çalışmalarda bütün veri tabanı üzerinde bir değer aramak gerekebiliyor. Bu durumda bütün tabloların bütün satır ve kolonlarında arama yapmak gerekecektir. Bunun için aşağıda verilen SQL kodu ile ilgili stored procedure oluşturulabilir. Prosedür oluşturulduktan sonra aranmak istenen değer prosedüre parametre olarak verilip prosedür işlendiğinde: değer hangi tablonun hangi kolonunda geçtiği sonuç olarak dönecektir.

```
CREATE PROC SearchAllTables
(
@SearchStr nvarchar(100)
)
AS
BEGIN

CREATE TABLE #Results (ColumnName nvarchar(370), ColumnValue
nvarchar(3630))

SET NOCOUNT ON

DECLARE @TableName nvarchar(256), @ColumnName nvarchar(128),
```

```

@SearchStr2 nvarchar(110)
SET @TableName = ''
SET @SearchStr2 = QUOTENAME('%' + @SearchStr + '%','''')

WHILE @TableName IS NOT NULL

BEGIN
SET @ColumnName = ''
SET @TableName =
(
SELECT MIN(QUOTENAME(TABLE_SCHEMA) + '.' +
QUOTENAME(TABLE_NAME))
FROM      INFORMATION_SCHEMA.TABLES
WHERE     TABLE_TYPE = 'BASE TABLE'
AND      QUOTENAME(TABLE_SCHEMA) + '.' + QUOTENAME(TABLE_NAME) =
@TableName
AND      OBJECTPROPERTY(
OBJECT_ID(
QUOTENAME(TABLE_SCHEMA) + '.' + QUOTENAME(TABLE_NAME)
), 'IsMSShipped'
) = 0
)

WHILE (@TableName IS NOT NULL) AND (@ColumnName IS NOT NULL)

BEGIN
SET @ColumnName =
(
SELECT MIN(QUOTENAME(COLUMN_NAME))
FROM      INFORMATION_SCHEMA.COLUMNS
WHERE     TABLE_SCHEMA    = PARSENAME(@TableName, 2)
AND      TABLE_NAME      = PARSENAME(@TableName, 1)
AND      DATA_TYPE IN ('char', 'varchar', 'nchar', 'nvarchar',
'int', 'decimal')
AND      QUOTENAME(COLUMN_NAME) = @ColumnName
)

IF @ColumnName IS NOT NULL

BEGIN
INSERT INTO #Results
EXEC
(
'SELECT ''' + @TableName + '.' + @ColumnName + ''', LEFT(' +

```

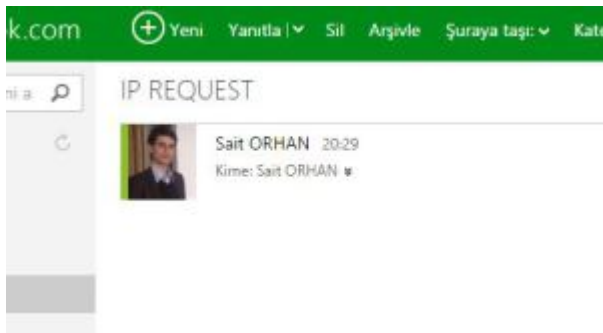


```
@ColumnName + ', 3630)
FROM ' + @TableName + ' (NOLOCK) ' +
' WHERE ' + @ColumnName + ' LIKE ' + @SearchStr2
)
END
END
END

SELECT ColumnName, ColumnValue FROM #Results
END
```

---

## Dinamik IP ile Ortadan Kaybolan SQL Server ve Diğer Serverlara IPlerini Soruyoruz



Daha önceki [yazımızda](#) dinamik ip'den dolayı daha sonra ulaşamadığımız sql serverdan söz etmiştik. Bu gün ortadan kaybolan bu server ve bilgisayarlara IPlerini soracağız, onlarda bize IP bilgilerini mail atacaklar, tabi biz de Ip bilgilerini mail



yoluyla soracaz :)))

Mantık şu şekilde çalışmaktadır: Ip bilgisi isteyen maillerin konu kısmı belli bir kalıpta olacak, örneğin "IP REQUEST". IP bilgisi istenen bilgisayarda da bir program sürekli veya belli aralıklarla belli bir mail hesabını sorgulayıp daha önce belirlenen başlıkla yeni mail olup olmadığını sorgulayacak. Eğer bu başlıkla yeni mail varsa otomatik olarak maili gönderen kişiye IP bilgilerini mail atacak □

Nasıl fikir □

Öncelikle [MailSystem.Net](https://www.mail-system.net/) kütüphanesini projenize referans olarak eklemelisiniz.

Yeni mail olup olmadığını kontrol edecek olan sınıf:

Şimdi de mail kontrolü yapacak olan metodumuz:

---

## Sorun: Dinamik IP Adresine

# Sahip SQL Server Sorunu :)

Bu yazımızda bir sorunumuz var, çözümlü de var elbette mesela statik IP adresi almak ☐ Ama o zaman da sorunlarımız var, bağlı olduğunuz ağ yöneticisi değilsiniz ve kafanıza göre kendinize IP alamıyorsunuz veya statik IP almak için para harcamak istemiyorsunuz ☐

Eee o zaman nasıl olacak bu iş, hem uzaktan SQL Servera bağlanmak istiyoruz hem de statik IPimiz yok. Diyelim ki bugün bağlandık ama bunun yarını da var ve yarın büyük ihtimal sunucumuzu o adreste bulamayacağız ☐ Şimdilik sorunu dile getirmeye çalıştık, müsait bir zamanda da kendimizce bir çözüm üretmeye çalışacağız.

Şimdilik gün ola hayır ola diyelim ☐