**SQL Server 2008 Veri Tipleri**

Kod yazarken ve veritabanı oluştururken veri tiplerinin seçimi çok önemlidir. İhtiyacımızı karşılayan en uygun tipi seçmemiz hem performans hem de kullanılan alan açısından önemlidir.

Sql Server 2008 de yer alan veri tiplerini ayrıntılı olarak inceleyeceğiz.

**Kesin Sayısal Veri Tipleri (Exact numerics):** bigint, bit, decimal, int, money, numeric, smallint, smallmoney, tinyint

**Yaklaşık Sayısal Veri Tipleri (Approximate Numerics):** float, real

**Tarih ve Zaman Veri Tipleri (Date and Time):** date, datetime, datetime2, datetimeoffset, smalldatetime, time

**Karakter Veri Tipleri (Character Strings):** char, varchar, text

**Unicode Karakter Veri Tipleri (Unicode Character Strings**): nchar, nvarchar, ntext

**Binary Veri Tipleri (Binary Strings):** binary, varbinary, image

**Diğer Veri Tipleri (Other Data Types):** cursor, hierarchyid, sql\_variant, table, timestamp, uniqueidentifier, xml

**1. Kesin Sayısal Veri Tipleri (Exact numerics)**

**bit:** 1 bayt yer kaplar. 1 ve 0 değerlerini alır. C#’ta bool’a karşılık gelir. true/false değerlerini bu tipte saklayabiliriz.

**tinyint:** 1 bayt yer kaplar. 0 ile 255 arası tam sayıları tutar. C#’ta byte tipine karşılık gelir.

**smallint:** 2 bayt yer kaplar. -32.768 ile 32.767 arası tam sayıları tutar. C#’ta short tipine karşılık gelir.

**int:** 4 bayt yer kaplar. -2.147.483.648 ile 2.147.483.647 arası tam sayıları tutar. C#’ta da int tipine karşılık gelir.

**bigint:** 8 bayt yer kaplar. -9.223.372.036.854.775.808 ile 9.223.372.036.854.775.807 arası tam sayıları tutar. C#’ta long tipine karşılık gelir.

**smallmoney:** 4 bayt yer kaplar. -214.748,3648 ile 214.748,3647 arası ondalık sayıları tutar. Virgülden sonra 4 basamak alır. Parasal verileri smallmoney veri tipinde saklayabiliriz. C#’ta direkt karşılığı yoktur, decimal kullanılır.

**money:** 8 bayt yer kaplar. kaplar. -922.337.203.685.477,5808 ile 922.337.203.685.477,5807 arası ondalık sayıları tutar. Virgülden sonra 4 basamak alır. Parasal verileri money veri tipinde saklayabiliriz. C#’ta direkt karşılığı yoktur, decimal kullanılır.

**2. Yaklaşık Sayısal Veri Tipleri (Approximate Numerics)**

**float:** kullanılmak istenen boyuta göre ortalama değer alır. **float(n)** şeklinde kullanılır. mesela virgülden sonra 20 bitlik bir alan kullanılmasını istiyoruz. o zaman float(20) olarak yazılır ve verdiğimiz bu boyuta göre kaydetmek istediğimiz sayı yuvarlanır. kesin değer değil de yaklaşık değer kaydedilmiş olur. n kısmı 1 ile 53 arasında olmalıdır. C#’ta double tipine karşılık gelir.

**real:** float(24) ile aynı özelliktedir. 4 bayt yer kaplar. – 3.40E+38 ile -1.18E-38, 0 ve 1.18E-38 ile 3.40E+38 arası ondalık sayıları tutar. C#’ta Single tipine karşılık gelir.

**decimal ve numeric:** kapladığı alan kullanılan basamak sayısına göre değişir. virgülden önce ve sonra toplam basamak sayısı 1-9 arası 5 bayt; 10-19 arası 9 bayt; 20-28 arası 13 bayt; 29-38 arası 17 bayt yer kaplar. – 10^38 +1 den 10^38 – 1 e kadar olan ondalık sayıları tutar. numeric ile decimal birebir aynı veri tipidir. C#’ta decimal tipine karşılık gelirler.

**3. Tarih ve Zaman Veri Tipleri (Date and Time)**

**date:** YYYY-MM-DD şeklinde tarihi tutar. 3 bayt yer kaplar. 0001-01-01 ile 9999-12-31 arası tarih değerlerini saklar.

**datetime:** YYYY-MM-DD hh:mm:ss[.mmm] şeklinde tarihi tutar (2011-01-21 12:35:29.123 gibi). 8 bayt yer kaplar. 1753-01-01 00:00:00.000 ile 9999-12-31 23:59:59.999 arası tarih değerlerini saklar.

**datetime2:** datetime göre daha hassas şekilde YYYY-MM-DD hh:mm:ss[.nnnnnnn] tarih tutar (2011-01-21 12:35:29.1234567 gibi). saniye bölümünün ondalık kısmında 7 basamağa kadar değer tutabilir. Bu değer sadece datetime2 olarak kullanıldığında varsayılan olarak 7 basamaktır.**datetime(n)** şeklinde kullanarak basamak değerini belirleyebiliriz. eğer n değeri 0 ile 2 arasında ise 6 bayt; 3 veya 4 ise 7 bayt; 5 ile 7 arasında ise 8 bayt yer kaplar.

**smalldatetime:** YYYY-MM-DD hh:mm:ss şeklinde tarih tutar. 4 bayt yer kaplar. 1900-01-01 00:00:00 ile 2079-06-06 23:59:59 arası tarih değerlerini saklar.

**time:** sadece saati hh:mm:ss[.nnnnnnn] şeklinde tutar. 00:00:00.0000000 ile 23:59:59.9999999 arası değer alır. datetime2′de olduğu gibi**time(n)** şeklinde n değerini belirleyebiliyoruz. n değeri 0 ile 2 arasında ise 3 bayt; 3 veya 4 ise 4 bayt; 5 ile 7 arasında ise 5 bayt yer kaplar. sadece time olarak kullanılırsak varsayılan n değeri 7′dir.

**datetimeoffset:** kullanımı ve tarih aralığı datetime2 ile aynıdır. Ülkelere göre değişen zaman farkını da tutmamıza olanak sağlar. YYYY-MM-DD hh:mm:ss[. nnnnnnn] [{+|-}hh:mm] şeklinde tarihi tutar (22.01.2012 02:07:23.1234567 +03:00). Saat farkı -14:00 ile +14:00 arasında değer alır. Yaptığımız uygulamada farklı ülkelerin tarih ve saat bilgilerini tutuyorsak bu veri tipini kullanabiliriz. datetimeoffset(n) şeklinde kullanılır.  n değeri 0 ile 2 arasında ise 8 bayt; 3 veya 4 ise 9 bayt; 5 ile 7 arasında ise 10 bayt yer kaplar. sadece datetimeoffset olarak kullanılırsak varsayılan n değeri 7′dir.

**4. Karakter Veri Tipleri (Character Strings)**

**char:** sabit uzunlukta karakter dizilerini tutar. **char(n)** şeklinde kullanılır. n karakter sayısıdır ve 1 ile 8000 arasında değer alır. belirlediğimiz n değerinden daha kısa uzunlukta olan veriler boşluk ile belirlediğimiz bu n değerine tamamlanır. ve bu n değerine göre (n x 1 bayt) yer kaplarlar.

**varchar:** sabit uzunlukta karakter dizilerini tutar. **varchar(n)** şeklinde kullanılır. n karakter sayısıdır ve 1 ile 8000 arasında değer alır. belirlediğimiz n değeri alabileceği maksimum karakter sayısıdır. Karakter sayısı daha kısa veri girersek char gibi boşlukla tamamlanmaz. Kaç karakter veri girilirse boyutu ona göre değişir. ((kaydedilenVerininKarakterSayisi x 1 bayt) + 2 bayt) olarak yer kaplar. **varchar(MAX)** şeklinde kullanırsak maksimum 8000 karakter değil de maksimum**2,147,483,647** karakter veri girilebilir.

**text:** varchar(max) ile aynı özelliktedir. maksimum 2,147,483,647 karakter veri girilebilir. microsoft bu veri tipini gelecek versiyonlarda kaldıracağı için kullanılması önerilmez. yerine**varchar(MAX)** kullanabilirsiniz.

**Not:** Eğer oluşturmuş olduğumuz veritabanın dil seçeneği (collation) türkçe (Turkish\_Cl\_AS gibi) ise unicode olmayan bu veri tiplerinde (char,varchar) de türkçe karakter saklayabiliriz. Fakat dil seçeneği farklıysa türkçe karakter gönderdiğimizde seçtiğimiz dil seçeneğine göre türkçe karakter kaydedilmeyebilir. (ı’lar i; ş’ler s’ye çevrilebilir). Böyle durumlarda ya veritabanı özelliklerinden dil seçeneğini (collation)’ı türkçeye çevirmeliyiz ya da unicode karakter veri tiplerini(nchar,nvarchar) kullanmalıyız.

**5. Unicode Karakter Veri Tipleri**

**nchar:** char ile kullanımı aynıdır. char’dan farklı olarak unicode karakterleri de saklayabilir. nchar(n) şeklinde kullanılır ve n değeri 1 ile 4000 arasındadır. char’ın iki katı kadar (n x 2 bayt) yer kaplar.

**nvarchar:** varchar ile kullanımı aynıdır. varchar’dan farklı olarak unicode karakterleri de saklayabilir. nvarchar(n) şeklinde kullanılır ve n değeri 1 ile 4000 arasındadır. varchar’ın iki katı kadar ((kaydedilenVerininKarakterSayisi x 2 bayt) + 2 bayt) olarak yer kaplar.

**ntext:** nvarchar(max) ile aynı özelliktedir. maksimum 1,073,741,823 karakter veri girilebilir. microsoft bu veri tipini gelecek versiyonlarda kaldıracağı için kullanılması önerilmez. yerine**nvarchar(MAX)** kullanabilirsiniz.

**Not:** Eğer veritabanımızdaki kayıtlarda birden fazla dil kullanılacaksa veya veritabanının dil seçeneğinden (collation) farklı bir dil ile kayıt yapılacaksa unicode karakter veri tipleri kullanılmalıdır.

**6. Binary Veri Tipleri (Binary Strings)**

**binary:** dosyaları(binary data) saklamak için kulanılır. binary(n) şeklinde n değeri 1 ile 8000 arasında değer alır. n bayt kadar yer kaplar.

**varbinary:** dosyaları(binary data) saklamak için kullanılır. binary’den farklı olarak boyutu kaydedilen dosyanın boyutuna göre değişir. varbinary(n) şeklinde n değeri 1 ile 8000 arasında değer alır. varbinary(MAX) olarak kullanıldığında maksimum 2,147,483,647 bayt (2 GB) büyüklüğünde dosya kaydedilebilir.

**image:** dosyaları(binary data) saklamak için kullanılır. maksimum 2,147,483,647 bayt (2 GB) büyüklüğünde dosya saklayabilir. microsoft, sql server’ın gelecek versiyonlarında image veri tipini kaldırmayı düşündüğü için yerine varbinary(MAX) kullanabilirsiniz.

**7. Diğer Veri Tipleri (Other Data Types)**

**uniqueidentifier:** 6F9619FF-8B86-D011-B42D-00C04FC964FF gibi bir GUID tipindeki verileri tutar. C#’taki karşılığı GUid’tir. Guid, harf ve sayılardan oluşan eşsiz bir datadır. İki Guid’in birbiri ile eşit olmayacağı garantidir.

**timestamp:** girilen kaydın (satırın) versiyon numarası gibidir (0x00000000000007D3). her satırda değişir ve satır güncellendiğinde de değişir. Tabloda sadece bir alan timestamp olabilir. tarih ve saat tutmak için kullanılmaz.

**xml:** <ROOT><a>111</a></ROOT> şeklinde xml dataları saklar ve tablo ve satırlarda sorgu yapmadan, query() metoduyla, kaydettiğimiz xml’in içindeki datalara ulaşabiliriz.

**sql\_variant:** farklı veri tiplerindeki değişkenleri kaydedebileceğimiz bir alan sağlar. maksimum boyutu 8016 bayt’tır. varchar(MAX), nvarchar(MAX), varbinary(MAX), text, ntext, image, timestamp, sql\_variant, hierarchyid, geography, geometry ve kullanıcı tarafından oluşturulan veri tipleri haricinde diğer bütün veri tiplerini saklayabilir.

.

**Sql Server 2008′de olup da Sql Server 2005′te yer almayan veri tipleri:**  
date  
time  
datetime2  
datetimeoffset  
hierarchyid