**SQL Server da Temporary (Geçici) Tablo Kullanmak**

SQL Server developerların işini çok kolaylaştıran ve kullanımı kolay temporary(geçici) tablo yapısını destekler. Geçici tablolar çalışma anında oluşturulur ayrıca normal bir tabloda yapabileceğimiz tüm işlemleri temp tablolarda da yapabiliriz. Temp Tablolar tempdb veritabanında saklanırlar.

Bu makalemiz de temp tabloların kullanımı ile ilgili giriş seviyesinde hızlı bir balış atacağız.

Temp Tablo Tipleri

SQL Server iki farklı temp tablo tipi sağlar.

* Local Temp Tablo
* Global Temp Tablo

Local Temp Tablo

Local Temp tablolar sadece tabloyu oluşturan kullanıcının, oluşturmak için kullandığı connection üzerin de geçerlidir. Kullanıcı bağlantıyı kapattığı anda otomatik olarak silinir. Local Temp tablo tanımlamak için tablo isminin önüne # işareti koymamız yeterlidir.

Global Temp Table

Global olarak temp tablo tanımlamak için ## karakterlerini tablo isminden önce eklememiz yeterlidir. Global temp tablolar bir connection içinde tanımlandığı anda, normal tablolar gibi tüm kullanıcılar ve connectionlar içinde kullanılabilir hale gelirler. Global tabloyu oluşturduğunuz connection kapatılana kadar bu tablo var olmaya devam eder.

SQL Serverda Temp Tablo Oluşturmak

Yukarıda anlattığımız iki farklı tipteki temp tabloların nasıl oluşturulup kullanılacağını aşağıdaki örneklere bakarak anlamaya çalışalım

Local Temp Tablo

[?](http://www.yazilimmutfagi.com/10143/veritabani/sql-server/sql-server-da-temporary-gecici-tablo-kullanmak.aspx)

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | CREATE TABLE #LocalTempTablo  (      ID int,      Ad varchar(20),      Soyad varchar(20)  ) |

Yukarıdaki sorgu ile tempdb içinde bir geçici(temp) tablo oluşturmuş olduk, normal bir tabloya insert/select yapıyormuş gibi temp tabloya da aşağıdaki gibi insert/select yapabiliriz.

INSERT

[?](http://www.yazilimmutfagi.com/10143/veritabani/sql-server/sql-server-da-temporary-gecici-tablo-kullanmak.aspx)

|  |  |
| --- | --- |
| 1 | insert into #LocalTempTablo values ( 1, 'Sabri','KUNT'); |

SELECT

[?](http://www.yazilimmutfagi.com/10143/veritabani/sql-server/sql-server-da-temporary-gecici-tablo-kullanmak.aspx)

|  |  |
| --- | --- |
| 1 | select \* from #LocalTempTablo |

Select Sonucu

[?](http://www.yazilimmutfagi.com/10143/veritabani/sql-server/sql-server-da-temporary-gecici-tablo-kullanmak.aspx)

|  |  |
| --- | --- |
| 1  2  3  4  5 | ID          Ad                   Soyad  ----------- -------------------- --------------------  1           Sabri                KUNT    (1 row(s) affected) |

Yukarıki scriptleri çalıştırdıktan başka bir sorgu penceresinde select veya insert sorgusu çalıştırırsanız aşağıdaki hatayı alırsınız.

[?](http://www.yazilimmutfagi.com/10143/veritabani/sql-server/sql-server-da-temporary-gecici-tablo-kullanmak.aspx)

|  |  |
| --- | --- |
| 1  2 | Msg 208, Level 16, State 0, Line 1  Invalid object name '#LocalTempTablo' |

Yukarıdaki hatanın sebebi, oluşturduğumuz temp tablonun sadece oluşturulduğu connection içinde geçerli olmasıdır.

Global Temp Tablo

Global bir temp tablo oluşturmak için iki adet # karakteri kullanıyoruz (##). Global temp tablolar oluşturulduğu connection aktif olduğu sürece tüm kullanıcılar ve connectionlarda kullanılabilir demiştir. Bir örnekle daha iyi anlayalım.

[?](http://www.yazilimmutfagi.com/10143/veritabani/sql-server/sql-server-da-temporary-gecici-tablo-kullanmak.aspx)

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | CREATE TABLE ##GlobalTempTablo  (      ID int,      Ad varchar(20),      Soyad varchar(20)  ) |

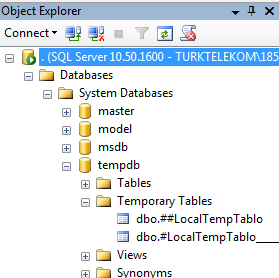
INSERT ve SELECT Sorgusu

[?](http://www.yazilimmutfagi.com/10143/veritabani/sql-server/sql-server-da-temporary-gecici-tablo-kullanmak.aspx)

|  |  |
| --- | --- |
| 1  2  3 | insert into ##GlobalTempTablo values ( 1, 'Sabri','KUNT');    select \* from ##GlobalTempTablo |

Yukarıdaki select ve insert sorgusunu istediğiniz herhangi bir sorgu ekranında çalıştırabilirsiniz. İlk oluşturduğunuz connection açık olduğu sürece her yerden bu tablolara erişim yapılabilir. Bunu kendinizde deneyerek görebilirsiniz.

SQL Server Temp tabloları aşağıda da görüldüğü gibi tempdb veritabanın da tutar.



Temp Tablolar Nerede ve Ne Amaçla Kullanılır

Büyük veriler ile çalışırken bazen bir sorgu sonucunda dönen kayıtları başka bir sorguda join işlemine tabi tutarız. Temp tablo kullanmak yerine iç içe select yazabiliriz fakat bu büyük kayıtlarda bazen performans sıkıntısı ve komplex sqllere sebeb olur. Bu yüzden belirli bir sorgu sonucunu temp bir tabloya atar ve bu tabloyuda başka bir sorguda kullanabiliriz. Bununla ilgili olarak aşağıdaki örneği inceleyebilirsiniz.

[?](http://www.yazilimmutfagi.com/10143/veritabani/sql-server/sql-server-da-temporary-gecici-tablo-kullanmak.aspx)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | select  top 100 CustomerKey,FirstName,LastName  into #temp  from dbo.DimCustomer    select        ProductKey,OrderDateKey        ,s.CustomerKey,SalesOrderNumber        ,UnitPrice from FactInternetSales s  inner join #temp t on s.CustomerKey = t.CustomerKey |

Alternatif Yöntem : Tablo Değişkeni

Temp tablo oluşturmak için alternatif diğer yöntem tablo değişkeni tanımlamaktır. Nasıl tanımlandığı ve kullanıldığını aşağıda görebilirsiniz.

[?](http://www.yazilimmutfagi.com/10143/veritabani/sql-server/sql-server-da-temporary-gecici-tablo-kullanmak.aspx)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | Declare @TempTableVariable TABLE(      ID int,      Ad varchar(20),      Soyad varchar(20)  )    insert into @TempTableVariable values ( 1, 'Sabri','KUNT');  insert into @TempTableVariable values ( 1, 'Ali','RÜZGAR');    select \* from @TempTableVariable |

Ne zaman Temp Tablo Ne zaman Tablo Değişkeni Kullanmalıyız

* Eğer 100 kayıtdan daha az kayıt ile çalışacaksanız tablo değişkeni kullanın, diğer türlü temp tablo kullanmalısınız. Bunun sebebi SQL server tablo değişkeni için istatistik oluşturmaz.
* Eğer index oluşturmanız gerekiyor ise temp tablo kullanın.