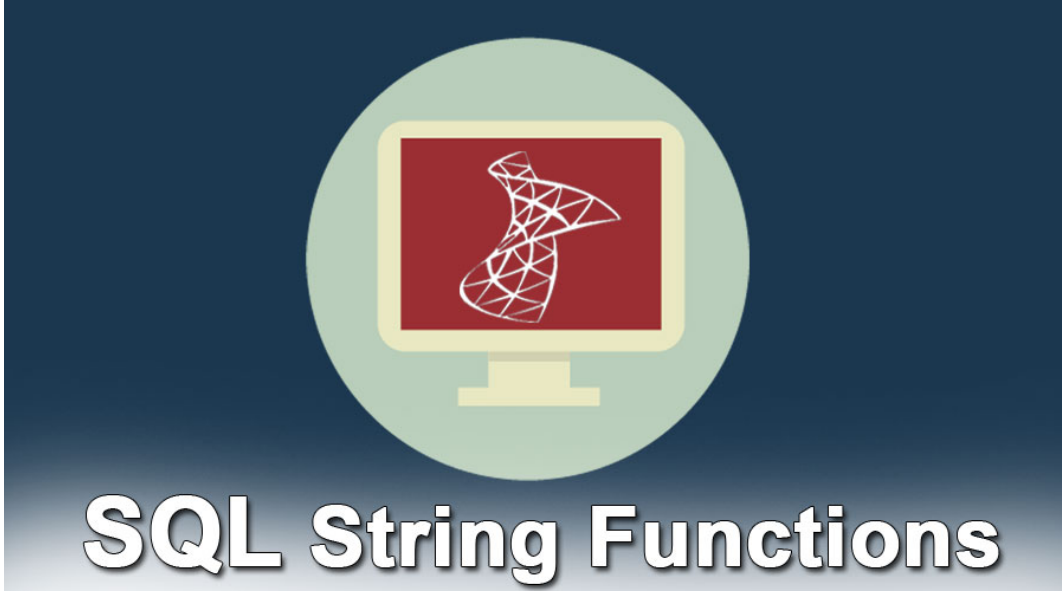


SQL Server String Fonksiyonları



Yazılımın her alanında olduğu gibi [SQL Server](#) üzerinde de en sık yapılan işlemler string işlemleridir. Bu yazımızda [SQL Server](#) içerisinde kullanılan string fonksiyonlarını işliyoruz olacağız.

ASCII

Parametre olarak aldığı stringin ilk harfinin ASCII karşılığını verir.

```
SELECT ASCII('ABC') AS A, ASCII('B') AS B,  
ASCII('a') AS a, ASCII('b') AS b,  
ASCII(1) AS [1], ASCII(2) AS [2];
```

| | A | B | a | b | 1 | 2 |
|---|----|----|----|----|----|----|
| 1 | 65 | 66 | 97 | 98 | 49 | 50 |

CHAR

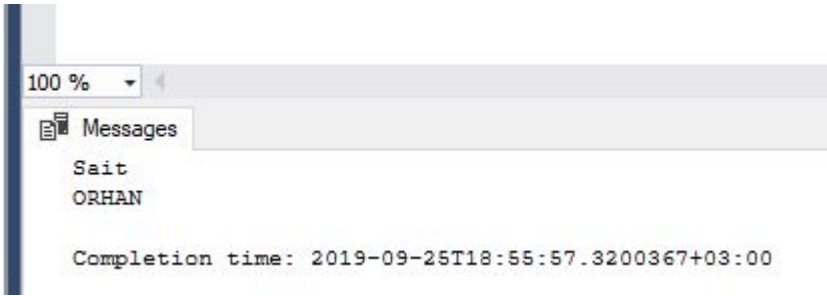
ASCII fonksiyonunun tersine parametre olarak aldığı int değerinin karakter karşılığını vermektedir.

```
SELECT CHAR(65), CHAR(66), CHAR(97), CHAR(98), CHAR(49),
CHAR(50)
```

| | (No column name) | (No column name) | (No column name) | (No column name) | (No column name) | (No column name) |
|---|------------------|------------------|------------------|------------------|------------------|------------------|
| 1 | A | B | a | b | 1 | 2 |

CHAR fonksiyonu kullanılarak yeni satır, tab, satır başı gibi kontrol karakterleri de insert dileyebilir. Örneğin aşağıdaki örnekte kişinin adı ve soyadı araya enter karakteri eklenerek alt alta yazıyor.

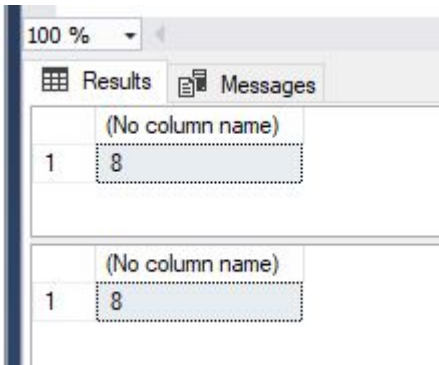
```
DECLARE @N NVARCHAR(40)
SET @N = 'Sait' + CHAR(13) + 'ORHAN'
PRINT @N
```



CHARINDEX

Verilen iki parametreden ilk verilen değeri ikinci parametre içerisinde arar ve bulunduğu ilk indeksi verir. Üçüncü parametre de isteğe bağlı ve verilmesi durumunda verilen indeksten sonrası için arar. Aranılan değer bulunmaz ise 0 değerini döner.

```
select CHARINDEX('HAN', 'Sait ORHAN')  
select CHARINDEX('HAN', 'Sait ORHAN', 7)
```



CONCAT

Parametre olarak aldığı string değerlerini birleştirir.

```
SELECT CONCAT ( 'Bugün ', 'Günlerden ', 25, '/', '09' ) AS
```

Tarih;

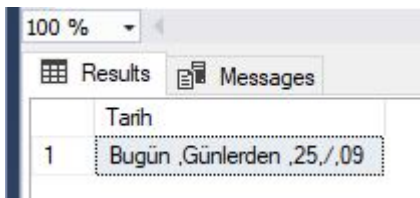


| | Tarih |
|---|-----------------------|
| 1 | Bugün Günlerden 25/09 |

CONCAT_WS

Concat fonksiyonu ile aynı şekilde çalışır ancak fark olarak stringleri birleştirirken araya ilk parametreyi koyar. Örneğin aşağıda yukarıdaki örneği “,” ile birleştiriyor.

```
SELECT CONCAT_ws (',', 'Bugün ', 'Günlerden ', 25, '/', '09' )  
AS Tarih;
```



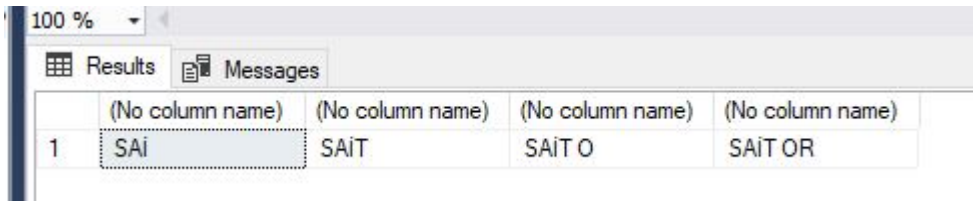
| | Tarih |
|---|--------------------------|
| 1 | Bugün ,Günlerden ,25/,09 |

LEFT

İki parametre alır. İlk parametre bir string değer alır, ikinci parametre olarak bir int değer alır. Sonuç olarak int değeri kadar harfi stringin solundan döndürür.

```
SELECT LEFT('SAİT ORHAN', 3), LEFT('SAİT ORHAN', 4),
```

```
LEFT('SAİT ORHAN', 6), LEFT('SAİT ORHAN', 7)
```



The screenshot shows a SQL query result grid with a zoom level of 100%. The grid has two tabs: 'Results' and 'Messages'. The 'Results' tab is active and displays a table with four columns, each labeled '(No column name)'. The first row contains the values 'SAİ', 'SAİT', 'SAİT O', and 'SAİT OR'.

| | (No column name) | (No column name) | (No column name) | (No column name) |
|---|------------------|------------------|------------------|------------------|
| 1 | SAİ | SAİT | SAİT O | SAİT OR |

LEN

Parametre olarak aldığı string değerın uzunluğunu döndürür.

```
SELECT LEN('SAİT ORHAN')
-----Sonuç
-- 10
```

LOWER

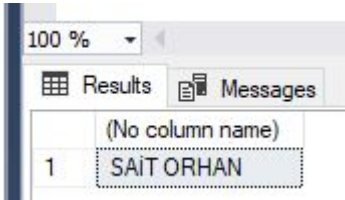
Parametre olarak aldığı stringin tümünü küçük harfe çevirerek döndürür.

```
SELECT lower('SAİT ORHAN')
-----Sonuç
-- sait orhan
```

LTRIM

Parametre olarak aldığı stringin solunda olan bütün boşlukları kaldırır.

```
SELECT LTRIM(' SAİT ORHAN ')
```



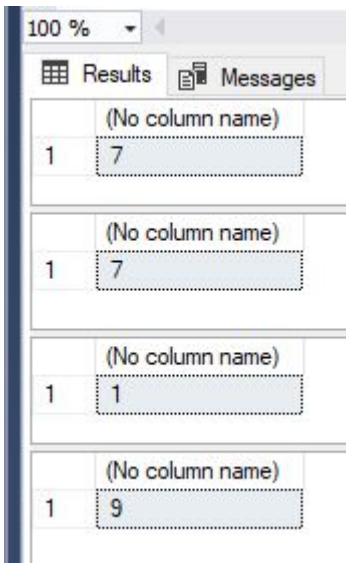
A screenshot of a SQL Server Results window. The window title is '100 %'. It has two tabs: 'Results' and 'Messages'. The 'Results' tab is active, showing a table with one row and one column. The column header is '(No column name)' and the value is 'SAİT ORHAN'.

| | (No column name) |
|---|------------------|
| 1 | SAİT ORHAN |

PATINDEX

İlk parametrede aldığı paterni ikinci parametrede verilen strin içerisinde arar ve bulunduğu ilk indeki döndürür. Bulamazsa 0 döner.

```
select PATINDEX('%R_A%', 'SAİT ORHAN')  
select PATINDEX('%R%', 'SAİT ORHAN')  
select PATINDEX('SA%', 'SAİT ORHAN')  
select PATINDEX('%AN', 'SAİT ORHAN')
```



A screenshot of a SQL Server Results window showing four separate query results. Each result is a table with one row and one column. The column header is '(No column name)'. The values are 7, 7, 1, and 9 respectively.

| | (No column name) |
|---|------------------|
| 1 | 7 |

| | (No column name) |
|---|------------------|
| 1 | 7 |

| | (No column name) |
|---|------------------|
| 1 | 1 |

| | (No column name) |
|---|------------------|
| 1 | 9 |

REPLACE

Üç parametre alır. İkinci parametrede verilen string değeri birinci parametrede verilen string içerisinde arar ve üçüncü parametrede verilen değer ile değiştirir. Örneğin aşağıdaki örnekte "sailorhan1" string değerinde geçen "1"leri "2" olarak değiştiriyor.

```
SELECT REPLACE('sailorhan1', '1', '2')  
--Sonuç  
-- sai2orhan2
```

REPLICATE

Birinci parametrede verilen string değerini ikinci parametrede verilen sayı kadar tekrar ederek döner.

```
SELECT REPLICATE('12', 3)  
--Sonuç  
-- 121212
```

REVERSE

Verilen string değeri ters çevirir.

```
SELECT REVERSE('123456789')  
--Sonuç  
-- 987654321
```

RIGHT

LEFT fonksiyonu ile aynı çalışır. Farkı metni sağdan alıyor olmasıdır.

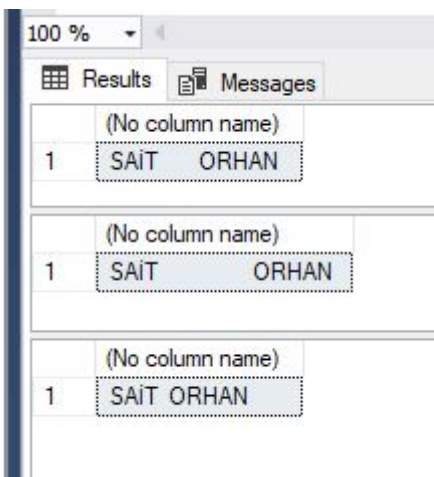
RTRIM

LTRIM fonksiyonu ile aynı çalışır. Farkı stringin sağındaki boşlukları siliyor olmasıdır.

SPACE

Parametre olarak aldığı sayı değeri kadar boşluk döndürür.

```
SELECT 'SAİT' + SPACE(7) + 'ORHAN'  
SELECT 'SAİT' + SPACE(16) + 'ORHAN'  
SELECT 'SAİT' + SPACE(2) + 'ORHAN'
```



The screenshot shows a SQL Server query results window with three rows of output. Each row has a column header '(No column name)' and a row number '1'. The first row shows 'SAİT' followed by 7 spaces and 'ORHAN'. The second row shows 'SAİT' followed by 16 spaces and 'ORHAN'. The third row shows 'SAİT' followed by 2 spaces and 'ORHAN'.

| | (No column name) |
|---|------------------|
| 1 | SAİT ORHAN |
| 1 | SAİT ORHAN |
| 1 | SAİT ORHAN |

STRING_AGG

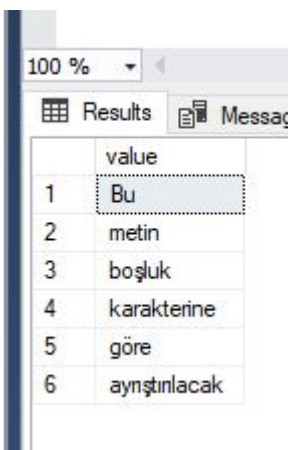
Select sorgusunda birinci parametre olarak kendisine verilen kolon değerini ikinci parametrede verilen karakter ile bölerek gösterir. Aşağıdaki örnekte Code alanını arasına “,” koyarak gösteriyor

```
SELECT STRING_AGG(Code, ',') from DocsTypes
--Sonuç
-- EN,FR,KA,LS,OR,PL,PR,PS,SE,ST,TL,TS
```

STRING_SPLIT

Birinci parametrede verilen metni ikinci parametrede verilen karaktere göre ayrıştırır. Geriye value adında kolon barındıran ve satırlarında karaktere göre ayrıştırma sonucunu tutan bir tablo döner.

```
SELECT * from STRING_SPLIT('Bu metin boşluk karakterine göre ayrıştırılacak', ' ')
```



| | value |
|---|----------------|
| 1 | Bu |
| 2 | metin |
| 3 | boşluk |
| 4 | karakterine |
| 5 | göre |
| 6 | ayrıştırılacak |

SUBSTRING

String içerisinden belli bir bölümü döndürür. Üç parametre alır. İlk parametrede işlem yapılacak string değeri, ikinci parametrede alınacak alt parçanın başladığı index değeri ve son parametre olarak da alınacak alt parçanın uzunluğunu alır.

```
SELECT SUBSTRING('Bu metin boşluk karakterine göre ayrıştırılacak', 1, 4)
SELECT SUBSTRING('Bu metin boşluk karakterine göre ayrıştırılacak', 5, 10)
SELECT SUBSTRING('Bu metin boşluk karakterine göre ayrıştırılacak', 25, 1000)
```

```
--Sonuç
-- Bu m
-- etin boşlu
-- ine göre ayrıştırılacak
```

TRIM

LTRIM ve RTRIM ile aynı çalışır. Farkı stringin hem sağ hem solundan boşlukları siliyor olmasıdır.

UPPER

LOWER ile aynı çalışır. Verilen stringin tamamını büyük harfe çevirir.