

(T7)討論 StringFunction

CourseGUID: e48417fc-9db5-4e99-822c-706c5ccef6cc

(T7)討論 StringFunction

## 0. What to learn

1. ASCII(Char), CHAR(INT)
2. LTRIM/RTRIM/LTRIM(RTRIM)/LOWER/UPPER/REVERSE/LEN
3. LEFT/RIGHT/CHARINDEX/SUBSTRING/REPLICATE/SPACE/PATINDEX/REPLACE/STUFF

## 0. What to learn

What to learn

- ASCII(Char)
- CHAR(INT)
- RTRIM(char(n))
- LTRIM(char(n))
- LTRIM(RTRIM(char(n)))
- LOWER(char(n))
- UPPER(char(n))
- REVERSE(char(n))
- LEN(char(n))
- LEFT( character\_expression , integer\_expression )
- RIGHT( character\_expression , integer\_expression )
- CHARINDEX( expressionToFind , expressionToSearch [ , start\_location ] )
- SUBSTRING( expression , StartIndex , length )
- REPLICATE( string\_expression , integer\_expression )
- REPLACE( string\_expression , string\_pattern , string\_replacement )
- SPACE( integer\_expression )
- PATINDEX( '%pattern%' , expression )
- STUFF( character\_expression , start , length , replaceWith\_expression )

## 1. ASCII(Char), CHAR(INT)

```
--=====
--T007_01_ASCII(Char), CHAR(INT)
--=====
--T007_01_01
--ASCII(Char)
PRINT ASCII('A');
PRINT ASCII('BC');
PRINT ASCII('a');
/*
1.
ASCII(Char)
Convert char to ASCII int.
1.1.
--PRINT  ASCII('A');
output
--65
65+26-1=90 which is ASCII code of Z
```

```

1.2.
--PRINT ASCII('BC');
output
--66
Returns the ASCII code of FIRST character.
1.3.
--PRINT ASCII('a');
output
--97
97+26-1=122 which is ASCII code of z
*/
=====
--T007_01_02
--CHAR(INT)
PRINT CHAR(65);
PRINT CHAR(97);
DECLARE @Number INT;
SET @Number = 65;
WHILE ( @Number <= 90 )
    BEGIN
        PRINT CHAR(@Number);
        SET @Number = @Number + 1;
    END;
DECLARE @Number2 INT;
SET @Number2 = 97;
WHILE ( @Number2 <= 122 )
    BEGIN
        PRINT CHAR(@Number2);
        SET @Number2 = @Number2 + 1;
    END;
DECLARE @Number3 INT;
SET @Number3 = 1;
WHILE ( @Number3 <= 255 )
    BEGIN
        PRINT CHAR(@Number3);
        SET @Number3 = @Number3 + 1;
    END;
/*
1.
CHAR(INT)
Convert ASCII int to char.
1.1.
--PRINT CHAR(65);
output
--A
65+26-1=90 which is ASCII code of Z
Thus, 65~90 is A to Z. (Upper case)
1.2.
--PRINT CHAR(97);
--a
97+26-1=122 which is ASCII code of Z
Thus, 97~122 is a to z. (Lower case)
2.
1~255 is the ASCII code of all char
*/
=====

```

## 2. LTRIM/RTRIM/LTRIM(RTRIM)/LOWER/UPPER/REVERSE/LEN

```
--=====
--T007_02_LTRIM/RTRIM/LTRIM(RTRIM)/LOWER/UPPER/REVERSE/LEN
--=====
--=====
--T007_02_01
--LTRIM(char(n))
PRINT LTRIM(' LTRIM(char(n))');
/*
1.
LTRIM(char(n))
Reference:
https://docs.microsoft.com/en-us/sql/t-sql/functions/ltrim-transact-sql
Returns a character expression after it removes leading blanks.
2.
output
--LTRIM(char(n))
*/
--=====
--T007_02_02
--RTRIM(char(n))
PRINT RTRIM('RTRIM(char(n)) ');
/*
1.
RTRIM(char(n))
Reference:
https://docs.microsoft.com/en-us/sql/t-sql/functions/rtrim-transact-sql
Returns a character string after truncating all trailing spaces.
2.
output
--RTRIM(char(n))
*/
--=====
--T007_02_03
--LTRIM(RTRIM(char(n)))
PRINT LTRIM(RTRIM(' LTRIM(RTRIM(char(n))) '));
/*
1.
LTRIM(RTRIM(char(n)))
Returns a character string after truncating all trailing spaces and leading blanks.
2.
output
--LTRIM(RTRIM(char(n)))
*/
--=====
--T007_02_04
--LOWER(char(n))
PRINT LOWER('LOWER(char(n))');
/*
1.
LOWER(char(n))
Reference:
https://docs.microsoft.com/en-us/sql/t-sql/functions/lower-transact-sql
Returns a character expression after converting uppercase character data to lowercase.
2.
output
--lower(char(n))
*/
--=====
--T007_02_05
```

```
--UPPER(char(n))
PRINT UPPER('upper(char(n))');
/*
1.
UPPER(char(n))
Reference:
https://docs.microsoft.com/en-us/sql/t-sql/functions/upper-transact-sql
Returns a character expression with lowercase character data converted to uppercase.
2.
output
--UPPER(CHAR(N))
*/
=====
--T007_02_06
--REVERSE(char(n))
PRINT REVERSE('ABCDEFGHIJKLMNOPQRSTUVWXYZ');
/*
1.
REVERSE(char(n))
Reference:
https://docs.microsoft.com/en-us/sql/t-sql/functions/reverse-transact-sql
Returns the reverse order of a string value.
2.
output
--ZYXWVUTSRQPONMLKJIHGFEDCBA
*/

=====
--T007_02_07
--LEN(char(n))
PRINT LEN('123456789');
PRINT LEN('1 2 3 4 5 6 7 8 9');
PRINT LEN('1 2 3 4 5 6 7 8 9 ');
PRINT LEN(' 1 2 3 4 5 6 7 8 9 ');
/*
1.
LEN(char(n))
Reference:
https://docs.microsoft.com/en-us/sql/t-sql/functions/len-transact-sql
Returns the number of characters of the specified string expression, excluding trailing blanks.
2.
2.1.
--PRINT LEN('123456789');
output
--9
2.2.
--PRINT LEN('1 2 3 4 5 6 7 8 9');
output
--17 = 9 numbers + 8 space
2.3.
--PRINT LEN('1 2 3 4 5 6 7 8 9 ');
output
--17 = 9 numbers + 8 space, ignore 2 trailing blanks
2.4.
--PRINT LEN(' 1 2 3 4 5 6 7 8 9 ');
output
--19 = 2 leading spaces + 9 numbers + 8 space, ignore 2 trailing blanks
*/
```

```
=====
```

### 3. LEFT/RIGHT/CHARINDEX/SUBSTRING/REPLICATE/SPACE/PATINDEX/REPLACE/STUFF

```
-----
--T007_03_LEFT/RIGHT/CHARINDEX/SUBSTRING/REPLICATE/SPACE/PATINDEX/REPLACE/STUFF
-----
--T007_03_01
--LEFT( character_expression , integer_expression )
PRINT LEFT('ABCDE', 3);
/*
1.
LEFT( character_expression , integer_expression )
Reference:
https://docs.microsoft.com/en-us/sql/t-sql/functions/left-transact-sql
Returns the left part of a character string with the specified number of characters.
2.
--PRINT LEFT('ABCDE', 3);
output
--ABC
*/
-----
--T007_03_02
--RIGHT( character_expression , integer_expression )
PRINT RIGHT('ABCDE', 3);
/*
1.
RIGHT( character_expression , integer_expression )
Reference:
https://docs.microsoft.com/en-us/sql/t-sql/functions/right-transact-sql
Returns the right part of a character string with the specified number of characters.
2.
--PRINT RIGHT('ABCDE', 3);
output
--CDE
*/
-----
--T007_03_03
--CHARINDEX( expressionToFind , expressionToSearch [ , start_location ] )
PRINT CHARINDEX('5', '123456789', 1);
/*
1.
CHARINDEX( expressionToFind , expressionToSearch [ , start_location ] )
Reference:
https://docs.microsoft.com/en-us/sql/t-sql/functions/charindex-transact-sql
Searches an expression for another expression and returns its starting position if found.
Index starts from 1, not from 0.
2.
--PRINT CHARINDEX('5', '123456789', 1);
Output
--5
Index starts from 1, not from 0.
Start from index 1 and search '5' in '123456789'
This will return 5 which means index of '5' is 5
*/
-----
--T007_03_04
--SUBSTRING( expression , StartIndex , length )
PRINT SUBSTRING('123456789', 6, 1);
```

```

PRINT SUBSTRING('123456789', 6, 3);
PRINT SUBSTRING('123456789', 6, 4);
PRINT SUBSTRING('123456789', 6, 10);
/*
1.
SUBSTRING( expression ,StartIndex , length )
Reference:
https://docs.microsoft.com/en-us/sql/t-sql/functions/substring-transact-sql
Returns part of a character, binary, text, or image expression in SQL Server.
--SUBSTRING ( char(n) ,StartIndex , Length )
index start from 1, not 0
In char(n) string, from StartIndex, then take the Length of chars as sub-string. Return the sub-string.
2.
2.1.
--PRINT SUBSTRING('123456789', 6, 1);
output
--6
2.2.
--PRINT SUBSTRING('123456789', 6, 3);
output
--678
2.3.
--PRINT SUBSTRING('123456789', 6, 4);
output
--6789
2.4.
--PRINT SUBSTRING('123456789', 6, 10);
output
--6789
*/

=====
--T007_03_05
--CHARINDEX/LEN
DECLARE @Email NVARCHAR(100);
SET @Email = '123456@789.com';
--Get the email domain name
PRINT CHARINDEX('@', @Email);
--7
PRINT ( CHARINDEX('@', @Email) + 1 );
--8
PRINT LEN(@Email);
--14
PRINT ( LEN(@Email) - CHARINDEX('@', @Email) );
--7
PRINT SUBSTRING(@Email, ( CHARINDEX('@', @Email) + 1 ),
( LEN(@Email) - CHARINDEX('@', @Email) ));
/*
1.
----SET @Email = '123456@789.com';
--PRINT CHARINDEX('@', @Email); -- Output 7
--PRINT ( CHARINDEX('@', @Email) + 1 ); --Output 8
--PRINT LEN(@Email); --Output 14
--PRINT ( LEN(@Email) - CHARINDEX('@', @Email) ); --output 7
--PRINT SUBSTRING(@Email, ( CHARINDEX('@', @Email) + 1 ),
-- ( LEN(@Email) - CHARINDEX('@', @Email) ));
----PRINT SUBSTRING(@Email, 8, 7);
----In '123456@789.com' string, from StartIndex 8, then take the 8 chars as the sub-string.
*/

=====
--T007_03_06
--REPLICATE( string_expression ,integer_expression )

```

```

PRINT REPLICATE('ReplicateMe', 3);
/*
1.
REPLICATE( string_expression ,integer_expression )
Reference:
https://docs.microsoft.com/en-us/sql/t-sql/functions/replicate-transact-sql
Repeats a string value a specified number of times.
2.
--PRINT REPLICATE('ReplicateMe', 3);
Output
--ReplicateMeReplicateMeReplicateMe
Repeats a string value 'ReplicateMe' 3 times.
*/
=====
--T007_03_07
--SPACE( integer_expression )
PRINT 'SPACE(1)' + SPACE(1) + 'SPACE(2)' + SPACE(2) + 'SPACE(3)' + SPACE(3)
+ 'SPACE(4)' + SPACE(4) + 'SPACE(5)' + SPACE(5) + 'END';
/*
1.
SPACE( integer_expression )
Reference:
https://docs.microsoft.com/en-us/sql/t-sql/functions/space-transact-sql
Returns a string of repeated spaces.
2.
--PRINT 'SPACE(1)' + SPACE(1) + 'SPACE(2)' + SPACE(2) + 'SPACE(3)' + SPACE(3)
--      + 'SPACE(4)' + SPACE(4) + 'SPACE(5)' + SPACE(5) + 'END'
output
--SPACE(1) SPACE(2) SPACE(3) SPACE(4) SPACE(5) END
*/
=====
--T007_03_08
--PATINDEX( '%pattern%' , expression )
PRINT PATINDEX('%ter%', 'interesting data'); --3
PRINT PATINDEX('%en_ure%', 'please ensure the door is locked'); --8
PRINT PATINDEX('%ein%', 'Das ist ein Test' COLLATE Latin1_General_BIN); --9
PRINT PATINDEX( '@789.com%', '123456@789.com') --7
/*
1.
PATINDEX( '%pattern%' , expression )
Reference:
https://docs.microsoft.com/en-us/sql/t-sql/functions/patindex-transact-sql
Returns the starting position of the first occurrence of a pattern in a specified expression,
or zeros if the pattern is not found, on all valid text and character data types.
2.
2.1.
--PRINT PATINDEX('%ter%', 'interesting data'); --output 3
2.2.
--PRINT PATINDEX('%en_ure%', 'please ensure the door is locked'); --output 8
% means several chars
_ means one chars
%en_ure% match 'ensure' which is at index 8
Index start from 1, not from 0
2.3.
--PRINT PATINDEX('%ein%', 'Das ist ein Test' COLLATE Latin1_General_BIN); --output 9
uses the COLLATE function to explicitly specify the collation of the expression that is searched.
2.4.
--PRINT PATINDEX( '@789.com%', '123456@789.com') --output 7
*/
=====
--T007_03_09
--REPLACE( string_expression , string_pattern , string_replacement )
PRINT REPLACE('123456', '234', 'bcd'); --1bcd56
PRINT REPLACE('123456@789.com', 'com', 'net'); --123456@789.net
/*

```

1.  
REPLACE( string\_expression , string\_pattern , string\_replacement )  
Reference:  
<https://docs.microsoft.com/en-us/sql/t-sql/functions/replace-transact-sql>  
Replaces all occurrences of a specified string value with another string value.

2.  
2.1.  
PRINT REPLACE('123456', '234', 'bcd'); --output 1bcd56  
In '123456', replace '234' by 'bcd'

2.2.  
PRINT REPLACE('123456@789.com', 'com', 'net'); --output 123456@789.net  
In 123456@789.com, replace 'com' by 'net'  
\*/

```
--=====
--T007_03_10
--STUFF( character_expression , start , length , replaceWith_expression )
PRINT STUFF('123456789', 2, 3, '*****'); --1*****56789
/*
```

1.  
STUFF( character\_expression , startIndex , length , replaceWith\_expression )  
Reference:  
<https://docs.microsoft.com/en-us/sql/t-sql/functions/stuff-transact-sql>  
The STUFF function inserts a string into another string.  
It deletes a specified length of characters in the first string  
at the start position and then  
inserts the second string into the first string at the start position.

2.  
PRINT STUFF('123456789', 2, 3, '\*\*\*\*\*'); --output 1\*\*\*\*\*56789  
Index start from 1 not from 0.  
From Index 2, delete 3 chars means delete 234  
then insert '\*\*\*\*\*' into index 2  
\*/