Introduction
to
ANSI SQL
(supplement)

Tim Hartley and Associates, LLC
Specialists in Database Technology
72 Crane Hill Road
Mansfield, CT 06268
860 450-0921
Working with String Functions

Although the ANSI SQL:1999 standard has defined several scalar or individual functions, each vendor tends to supply their own functions or provides slightly different implementations of the standard functions.

In this supplement, we examine some useful string functions, using the Microsoft SQL Server implementation to demonstrate their potential use.

The specific functions we consider are:

- CHARINDEX
- LEN
- SUBSTRING
- LEFT
- RIGHT
- PATINDEX
- LOWER
- UPPER
Working with String Functions

CHARINDEX

Returns the starting position of the specified expression in a character string. If the expression is not found within the character string, CHARINDEX returns 0.

Syntax:

```
CHARINDEX (expression1, expression2 [ , start_location ])
```

Arguments:

expression1
An expression containing the sequence of characters to be found.

eexpression2
An expression, usually a column searched for the specified sequence.

start_location
Character position to start searching for expression1 in expression2. If start_location is not given, is a negative number, or is zero, the search starts at the beginning of expression2.

Return Type:

int

Example:

```
SELECT CNO, CNAME,
       CHARINDEX (' ', CNAME) AS "FIRST SPACE"
FROM COURSE
```
Working with String Functions

LEN

Returns the number of characters, rather than the number of bytes, of the
given string expression, excluding trailing blanks.

Syntax:

LEN (string_expression)

Arguments:

string_expression
  The string expression to be evaluated.

Return Type:

int

Example:

SELECT CNO, CNAME,  
     LEN(CNAME)       
FROM COURSE
Working with String Functions

SUBSTRING

Returns a portion of a character string.

Syntax:

    SUBSTRING (expression, start, length)

Arguments:

    expression
    A character string, a column, or an expression that includes a column.

    start
    An integer that specifies where the substring begins.

    length
    An integer that specifies the length of the substring (the number of characters or bytes to return).

Example:

    SELECT SUBSTRING(CNAME, 6, 5) AS "NO FIRST WORD"
    FROM COURSE
    WHERE CNO = 'C11'
Working with String Functions

LEFT

Returns the part of a character string starting at a specified number of characters from the left side of the string.

Syntax:

    LEFT (character_expression, integer_expression)

Arguments:

    character_expression
        A constant, variable, or column of character data.

    integer_expression
        A positive whole number indicating the number of characters to extract.

Return Type:

    varchar

Example:

    SELECT CNO,
        LEFT(CNAME,5) as "FIRST 5 CHARACTERS"
    FROM COURSE
Working with String Functions

RIGHT

Returns the part of a character string starting a specified number of integer_expression characters from the right side of the string.

Syntax:

    RIGHT (character_expression, integer_expression)

Arguments:

    character_expression
    A constant, variable, or column of character data.

    integer_expression
    A number indicating the starting position of the extracted string.

Return Type:

    varchar

Example:

    SELECT CNO,
           RIGHT(CNAME,5) AS "LAST 5 CHARACTERS"
    FROM COURSE
Working with String Functions

Query objective: Locate the position of the first space within a character string column.

```
SELECT CNO, CNAME,
       CHARINDEX(' ', CNAME) AS "FIRST SPACE"
FROM COURSE
```

Query objective: Extract that portion of the string following the first word of a multi-word character string, i.e., a character string containing an embedded space.

Use position information obtained from the previous query to extract a the string.

```
SELECT SUBSTRING(CNAME, 6, 5) AS "NO FIRST WORD"
FROM COURSE
WHERE CNO = 'C11'
```

The first attempt at substring extraction was not quite correct. The desired substring actually begins in the position immediately following the space. Identify the actual starting position of the desired substring.

```
SELECT CNO, CNAME,
       CHARINDEX(' ', CNAME) + 1 AS "AFTER SPACE"
FROM COURSE
```

Use the position identified in the previous query as the starting position of the substring.

```
SELECT SUBSTRING(CNAME, 7, 5) AS "NO FIRST WORD"
FROM COURSE
WHERE CNO = 'C11'
```
Working with String Functions

Query objective: Determine the length of a character string located in a table column.

```sql
SELECT CNO, CNAME,
       LEN(CNAME) AS "LENGTH OF NAME"
FROM COURSE
```

Query objective: Determine the length of a character string located in a table column, the starting position of the first embedded space, and the number of characters that follow the space.

```sql
SELECT CNO, CNAME,
       LEN(CNAME) AS "LENGTH OF NAME",
       CHARINDEX(' ', CNAME) AS "FIRST SPACE",
       LEN(CNAME)-CHARINDEX(' ', CNAME) AS "THE REST"
FROM COURSE
```

Query objective: Extract that portion of the string following the first word of a multi-word character string, i.e., a character string containing an embedded space.

```sql
SELECT CNO, CNAME,
       SUBSTRING(CNAME,
                  CHARINDEX(' ', CNAME)+1,
                  LEN(CNAME)-CHARINDEX(' ', CNAME))
       AS "NO FIRST WORD"
FROM COURSE
```
Working with String Functions

Query objective: Extract the first five characters of a character string stored in a table column.

```
SELECT CNO,
       LEFT(CNAME, 5) as "FIRST 5 CHARACTERS"
FROM COURSE
```

Query objective: Extract the last five characters of a character string stored in a table column.

```
SELECT CNO,
       RIGHT(CNAME, 5) AS "LAST 5 CHARACTERS"
FROM COURSE
```

Query objective: Extract that portion of the string following the first word of a multi-word character string, i.e., a character string containing an embedded space.

```
SELECT CNO,
       RIGHT(CNAME, CHARINDEX (' ', CNAME) - 1)
FROM COURSE
```

The first attempt did not quite work because there were some strings that did not have an embedded space. The CHARINDEX function returned 0 for these strings, resulting in a negative value for the length argument of the RIGHT function.

```
SELECT CNO,
       RIGHT(CNAME, CHARINDEX (' ', CNAME) - 1)
FROM COURSE
WHERE CHARINDEX (' ', CNAME) <> 0
```
Working with String Functions

PATINDEX

Returns the starting position of the first occurrence of a pattern in a specified character expression. Zero is returned if the pattern is not found.

Syntax:

\texttt{PATINDEX ('\%pattern\%', expression)}

Arguments:

- \texttt{pattern}
  A literal string. Wildcard characters can be used in the pattern.
  The \% character must precede and follow the pattern string (except when searching for first or last characters).

- \texttt{expression}
  An expression that is searched for the specified pattern.

Return Type:

\texttt{int}

Examples:

\begin{verbatim}
SELECT CNO, CDESCP, 
    PATINDEX ('%MA%', CDESCP) 
FROM COURSE

SELECT CNO, CDESCP, 
    PATINDEX ('%M%A%', CDESCP) 
FROM COURSE
\end{verbatim}
Working with String Functions

LOWER

Returns a character expression after converting uppercase character data to lowercase.

Syntax:

   LOWER(character_expression)

Arguments:

   character_expression
   A constant, variable, or column of character data.

Return Type:

   varchar

Example:

   SELECT CNO,
       LOWER(CDESCP) AS "LOWERCASE"
   FROM COURSE
Working with String Functions

UPPER

Returns a character expression with lowercase character data converted to uppercase.

Syntax:

    UPPER(character_expression)

Arguments:

    character_expression
    A constant, variable, or column of character data.

Return Type:

    varchar

Example:

    SELECT CNO,
            UPPER(CDESCP) AS "UPPERCASE"
    FROM COURSE
Workshop: Using Strings

1. Using the CHARINDEX function, identify the staff members who have an embedded space in their name. Show the name and the position of the space.

2. Using the CHARINDEX, SUBSTRING, and LEN functions, identify the staff members who have an embedded space in their name. Show only the last name, i.e., that portion of the character string following the embedded space.

3. Using the CHARINDEX and LEFT functions, identify the staff members who have an embedded space in their name. Show only the first name, i.e., that portion of the character string preceding the embedded space.

4. Using the CHARINDEX, SUBSTRING, LEN, and LEFT functions together with the concatenation operator (+), identify the staff members who have an embedded space in their name. Show the last name followed by the first name. Separate the two names using a comma and space. The output format should be: “KISS, HANK”
Workshop Solutions

1. SELECT ENAME, CHARINDEX(' ', ENAME) AS "LOCATION OF SPACE"
   FROM STAFF
   WHERE CHARINDEX(' ', ENAME) <> 0

2. SELECT SUBSTRING(ENAME, CHARINDEX(' ', ENAME)+1, LEN(ENAME)-CHARINDEX(' ', ENAME)) AS "LAST NAME"
   FROM STAFF
   WHERE CHARINDEX(' ', ENAME) <> 0

3. SELECT LEFT(ENAME, CHARINDEX(' ', ENAME)-1) AS "FIRST NAME"
   FROM STAFF
   WHERE CHARINDEX(' ', ENAME) <> 0

4. SELECT SUBSTRING(ENAME, CHARINDEX(' ', ENAME)+1, LEN(ENAME)-CHARINDEX(' ', ENAME)) + ', ' + LEFT(ENAME, CHARINDEX(' ', ENAME)-1) AS "LAST, FIRST"
   FROM STAFF
   WHERE CHARINDEX(' ', ENAME) <> 0