Cast( ) & Convert( ) Functions

A note by
Shah Md. Arshad Rahman Ziban

Cast( ) Function

The Cast() function is used to convert an expression of one data type to another.

Syntax:

```
CAST(expression AS target_data_type)
```

Example:

```
-- Convert a decimal number to an integer
SELECT CAST(33.50 AS INT);
-- Output: 33

-- Convert a character ('A') to a variable character (VARCHAR) with a maximum length of 10
SELECT CAST('A' AS VARCHAR(10));
-- Output: 'A'

-- Convert a string representing a date ('2017-11-02') to a DATE type
SELECT CAST('2017-11-02' AS DATE);
-- Output: 2017-11-02 (as a DATE)

-- Convert a string ('Hello kitty') to a fixed-length character (CHAR) with a length of 15
SELECT CAST('Hello kitty' AS CHAR(15));
-- Output: 'Hello kitty   ' (padded with spaces to make it 15 characters)

-- Convert a string representing a date ('2017-08-25') to a DATETIME type
SELECT CAST('2017-08-25' AS DATETIME);
-- Output: 2017-08-25 00:00:00 (DATETIME with default time)
```

Convert( ) Function

The Convert() function is similar to Cast() but offers more flexibility, especially with date formats.

Syntax:

```
SELECT CONVERT(target_data_type, expression)
```

Example:

```
-- Convert a decimal number (25.02) to an integer
SELECT CONVERT(INT, 25.02);
-- Output: 25

-- Convert a string ('2019-03-22') to a DATETIME type
SELECT CONVERT(DATETIME, '2019-03-22');
-- Output: 2019-03-22 00:00:00.000

-- Convert a string ('Naima') to a VARCHAR(20) and alias it as 'Name'
SELECT CONVERT(VARCHAR(20), 'Naima') AS Name FROM test;
-- Output: 'Naima' (as VARCHAR)
```

Convert function can be used for datetime syntax:

Syntax:

```
SELECT CONVERT(target_data_type, expression[, style])
```

101: mm/dd/yyyy (US)
102: yyyy.mm.dd (ANSI)
103: dd/mm/yyyy (British)
104: dd.mm.yyyy (German)
105: dd-mm-yyyy (Italian)

Example:

```
SELECT EmpName,
    CONVERT(VARCHAR, JoiningDate, 103) AS JoiningDate
FROM EmpDetail;
```

Date Functions

```
-- 1. DAY(): Returns the day number of the month for a given date
SELECT DAY('2019-01-11') AS DayNumber;
-- Output: 11

-- 2. MONTH(): Returns the month number of the year for a given date
SELECT MONTH('2019-01-11') AS MonthNumber;
-- Output: 1

-- 3. YEAR(): Returns the year number for a given date
SELECT YEAR('2019-01-11') AS YearNumber;
-- Output: 2019

-- 4. ISDATE(): Checks if the given value is a valid date
SELECT ISDATE('2019/3/22') AS IsValidDate;
-- Output: 1
SELECT ISDATE('Students') AS IsValidDate;
-- Output: 0

-- 5. DATENAME(): Returns a string representing a part of the given date
SELECT DATENAME(MONTH, '2019-03-22') AS MonthName;
-- Output: 'March'

-- 6. DATEADD(): Adds a specified number to a date part and returns the new date
SELECT DATEADD(DAY, 5, '2019-01-11') AS NewDate;
-- Output: '2019-01-16'

-- 7. DATEDIFF(): Counts the specified date part boundaries between two dates
SELECT DATEDIFF(DAY, '2019-01-11', '2019-01-16') AS DaysDifference,
    DATEDIFF(MONTH, '2019-01-11', '2020-01-16') AS MonthsDifference,
    DATEDIFF(YEAR, '2019-01-11', '2022-01-16') AS YearsDifference;
-- Output: DaysDifference: 5, MonthsDifference: 12, YearsDifference: 3

-- 8. SYSDATETIME(): Returns the current system date and time
SELECT SYSDATETIME() AS CurrentDateTime;
-- Output: (e.g., '2024-11-17 14:11:19.464')
```