

Relational DB & SQL - C11

[Dashboard](#) / [Courses](#) / [Miscellaneous](#) / [RDB & SQL - C11](#) / [Functions](#) / [Other Functions & Expressions](#)

Other Functions & Expressions

To do: Go through the activity to the end

COALESCE, NULLIF and ISNUMERIC Expressions

`COALESCE(expression [, ...n])`

Evaluates the arguments in order and returns the current value of the first expression that initially doesn't evaluate to NULL. For example,

```
1 SELECT COALESCE(NULL, NULL, 'third_value', 'fourth_value');
```

returns the **third value** because the third value is the first value that isn't null.

Tips:

If all arguments are `NULL`, `COALESCE` returns `NULL`. At least one of the null values must be a typed `NULL`.

Comparing `COALESCE` and `CASE`

The `COALESCE` expression is a syntactic shortcut for the `CASE` expression. That is, the code `COALESCE(expression1,...n)` is rewritten by the query optimizer as the following `CASE` expression:

```
1 CASE
2 WHEN (expression1 IS NOT NULL) then expression1
3 WHEN (expression2 IS NOT NULL) then expression2
4 ...
5 ELSE expressionN
6 END
```

Examples:

query:

```
1 SELECT COALESCE(Null, Null, 1, 3) AS col
```

result:

```
1 col
2 -----
3 1
```

query:

```
1 SELECT COALESCE(Null, Null, 'William', Null) AS col
```

result:

```
1 col
2 -----
3 William
```

NULLIF(expression, expression)

Returns a null value if the two specified expressions are equal. For example,

```
1 SELECT NULLIF(4,4) AS Same, NULLIF(5,7) AS Different;
```

returns **NULL** for the first column (4 and 4) because the two input values are the same. The second column returns the first value (5) because the two input values are different.

Examples:

query:

```
1 SELECT NULLIF(1, 3) AS col
```

result:

```
1 col
2 -----
3 1
```

query:

```
1 SELECT NULLIF('2021-01-01', '2021-01-01') AS col
```

result:

```
1 col
2 -----
3 NULL
```

Tips:

You can use the **NULLIF()** function to find the product whose price does not change. If all arguments are **NULL**, **COALESCE** returns **NULL**. At least one of the null values must be a typed **NULL**.

ISNUMERIC(expression)

Determines whether an expression is a valid numeric type.

Returns 1 when the input expression evaluates to a valid numeric data type; otherwise it **returns 0**. Valid numeric data types are: **bigint**, **int**, **smallint**, **tinyint**, **bit**, **decimal**, **numeric**, **float**, **real**, **money**, **smallmoney**.

query:

1	SELECT ISNUMERIC ('William') AS col	
---	-------------------------------------	--

result:

1	col	
2	---	
3	0	

query:

1	SELECT ISNUMERIC (123.455) AS col	
---	-----------------------------------	--

result:

1	col	
2	---	
3	1	

Previous

Next

You have completed 50% of the lesson

50%

Jump to...



[Reset user tour on this page](#)



© 2021 Copyright: Clarusway.com