LAB - AND Operator

In this lab, you will learn how to use the SQL Server `AND` operator to combine multiple Boolean expressions.

The `AND` is a logical operator that allows you to combine two Boolean expressions. It returns `TRUE` only when both expressions evaluate to `TRUE`.

The following illustrates the syntax of the `AND` operator:

```
boolean_expression AND boolean_expression
```

The `boolean_expression` is any valid Boolean expression that evaluates to `TRUE`, `FALSE`, and `UNKNOWN`.

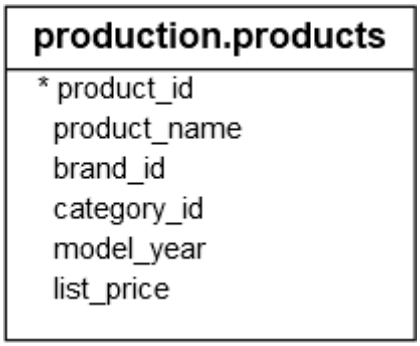
The following table shows the result when you combine `TRUE`, `FALSE`, and `UNKNOWN` values using the `AND` operator:

	TRUE	**FALSE**	**UNKNOWN**
TRUE	TRUE	FALSE	UNKNOWN
FALSE	FALSE	FALSE	FALSE
UNKNOWN	UNKNOWN	FALSE	UNKNOWN

When you use more than one logical operator in an expression, SQL Server always evaluates the `AND` operators first. However, you can change the order of evaluation by using parentheses.

SQL Server `AND` operator examples

See the following `products` table



A) Using `AND` operator example

The following example finds the products where the category identification number is one and the list price is greater than 400:

```
SELECT
    *
FROM
    production.products
WHERE
    category_id = 1
AND list_price > 400
ORDER BY
    list_price DESC;
```

Here is the result:

product_id	product_name	brand_id	category_id	model_year	list_price
98	Electra Straight 8 3i (20-inch) - Boy's - 2017	1	1	2017	489.99
100	Electra Townie 3i EQ (20-inch) - Boys' - 2017	1	1	2017	489.99
280	Trek Superfly 24 - 2017/2018	9	1	2018	489.99

B) Using multiple `AND` operators example

The following statement finds the products that meet all the following conditions: category id is 1, the list price is greater than 400, and the brand id is 1:

```
SELECT
    *
FROM
    production.products
WHERE
    category_id = 1
AND list_price > 400
AND brand_id = 1
ORDER BY
    list_price DESC;
```

The result is as follows:

product_id	product_name	brand_id	category_id	model_year	list_price
98	Electra Straight 8 3i (20-inch) - Boy's - 2017	1	1	2017	489.99
100	Electra Townie 3i EQ (20-inch) - Boys' - 2017	1	1	2017	489.99

C) Using the `AND` operator with other logical operators

See the following query example:

```
SELECT
    *
FROM
    production.products
WHERE
    brand_id = 1
OR brand_id = 2
AND list_price > 1000
ORDER BY
    brand_id DESC;
```

The following shows the result:

product_id	product_name	brand_id	category_id	model_year	list_price
41	Haro Shift R3 - 2017	2	6	2017	1469.99
46	Haro SR 1.3 - 2017	2	6	2017	1409.99
64	Electra Townie Original 7D - 2017	1	3	2017	489.99
70	Electra Amsterdam Original 3i - 2015/2017	1	3	2017	659.99
74	Electra Cruiser Lux 1 - 2017	1	3	2017	439.99
75	Electra Cruiser Lux Fat Tire 1 Ladies - 2017	1	3	2017	599.99
76	Electra Girl's Hawaii 1 16" - 2017	1	3	2017	299.99
77	Electra Glam Punk 3i Ladies' - 2017	1	3	2017	799.99
81	Electra Amsterdam Fashion 7i Ladies' - 2017	1	3	2017	1099.99

In this example, we used both `OR` and `AND` operators in the condition. As always, SQL Server evaluated the `AND` operator first. Therefore, the query retrieved the products whose brand id is two and list price is greater than 1,000 or the products whose brand id is one.

To get the product whose brand id is one or two and list price is larger than 1,000, you use parentheses as follows:

```
SELECT
    *
FROM
    production.products
WHERE
    (brand_id = 1 OR brand_id = 2)
AND list_price > 1000
ORDER BY
    brand_id;
```

The query returned the following result:

product_id	product_name	brand_id	category_id	model_year	list_price
81	Electra Amsterdam Fashion 7i Ladies' - 2017	1	3	2017	1099.99
191	Electra Loft Go! 8i - 2018	1	5	2018	2799.99
192	Electra Townie Go! 8i - 2017/2018	1	5	2018	2599.99
195	Electra Townie Go! 8i Ladies' - 2018	1	5	2018	2599.99
198	Electra Townie Commute Go! - 2018	1	5	2018	2999.99
199	Electra Townie Commute Go! Ladies' - 2018	1	5	2018	2999.99
250	Electra Townie Go! 8i - 2017/2018	1	3	2018	2599.99
251	Electra Townie Commute Go! - 2018	1	3	2018	2999.99
252	Electra Townie Commute Go! Ladies' - 2018	1	3	2018	2999.99
253	Electra Townie Go! 8i Ladies' - 2018	1	3	2018	2599.99
258	Electra Amsterdam Royal 8i - 2017/2018	1	3	2018	1259.90
259	Electra Amsterdam Royal 8i Ladies - 2018	1	3	2018	1199.99
303	Electra Townie Go! 8i - 2017/2018	1	2	2018	2599.99
41	Haro Shift R3 - 2017	2	6	2017	1469.99
46	Haro SR 1.3 - 2017	2	6	2017	1409.99

In this lab, you have learned how to use the SQL Server `AND` operator to combine two Boolean expressions.