

Operators for filtering dates and numbers

Previously, you examined operators like less than (<) or greater than (>) and explored how they can be used in filtering numeric and date and time data types. This reading summarizes what you learned and provides new examples of using operators in filters.

Numbers, dates, and times in cybersecurity

Security analysts work with more than just **string data**, or data consisting of an ordered sequence of characters.

They also frequently work with **numeric data**, or data consisting of numbers. A few examples of numeric data that you might encounter in your work as a security analyst include:

- the number of login attempts
- the count of a specific type of log entry
- the volume of data being sent from a source
- the volume of data being sent to a destination

You'll also encounter **date and time data**, or data representing a date and/or time. As a first example, logs will generally timestamp every record. Other time and date data might include:

- login dates
- login times
- dates for patches
- the duration of a connection

Comparison operators

In SQL, filtering numeric and date and time data often involves operators. You can use the following operators in your filters to make sure you return only the rows you need:

operator	use
<	less than
>	greater than
=	equal to
<=	less than or equal to
>=	greater than or equal to
<>	not equal to

Note: You can also use `!=` as an alternative operator for not equal to.

Incorporating operators into filters

These comparison operators are used in the *WHERE* clause at the end of a query. The following query uses the `>` operator to filter the *birthdate* column. You can run this query to explore its output:

```
SELECT firstname, lastname, birthdate
FROM employees
WHERE birthdate > '1970-01-01';
```

+-----+-----+-----+		
FirstName LastName BirthDate		
+-----+-----+-----+		
Jane	Peacock	1973-08-29 00:00:00
Michael	Mitchell	1973-07-01 00:00:00
Robert	King	1970-05-29 00:00:00
+-----+-----+-----+		

This query returns the first and last names of employees born after, but not on, '1970-01-01' (or January 1, 1970). If you were to use the `>=` operator instead, the results would also include results on exactly '1970-01-01'.

In other words, the `>` operator is exclusive and the `>=` operator is inclusive. An **exclusive operator** is an operator that does not include the value of comparison. An **inclusive operator** is an operator that includes the value of comparison.

BETWEEN

Another operator used for numeric data as well as date and time data is the *BETWEEN* operator. *BETWEEN* filters for numbers or dates within a range. For example, if you want to find the first and last names of all employees hired between January 1, 2002 and January 1, 2003, you can use the *BETWEEN* operator as follows:

```
SELECT firstname, lastname, hiredate
FROM employees
WHERE hiredate BETWEEN '2002-01-01' AND '2003-01-01';
```

```
+-----+-----+-----+
| FirstName | LastName | HireDate      |
+-----+-----+-----+
| Andrew   | Adams    | 2002-08-14 00:00:00 |
| Nancy    | Edwards  | 2002-05-01 00:00:00 |
| Jane     | Peacock  | 2002-04-01 00:00:00 |
+-----+-----+-----+
```

Note: The *BETWEEN* operator is inclusive. This means records with a *hiredate* of January 1, 2002 or January 1, 2003 are included in the results of the previous query.

Key takeaways

Operators are important when filtering numeric and date and time data. These include exclusive operators such as $<$ and inclusive operators such as \leq . The *BETWEEN* operator, another inclusive operator, helps you return the data you need within a range.

Revision #1

Created 10 July 2023 12:09:56 by naruzkurai

Updated 10 July 2023 12:11:45 by naruzkurai