Operators for Adv. Querying

Introduction to SQL operators

Operators allow you to further refine querying tasks. These are the operators that we'll cover:

- Common SQL Logical Operators
  - The AND operator
  - The NOT operator
  - The OR and IN operators
  - The BETWEEN operator
  - The LIKE operator and the % wildcard
- Arithmetic operators
- Comparison operators

Operators for Adv. Querying

The AND operator

In the previous lecture you learned that the WHERE clause allows you to limit your queries based on a specific value. The query below limits the result to only return records where the name is equal to 'Wally'.

```
SELECT * FROM employees WHERE name = 'Wally'
```

But what if we had more than one Wally in the database? You could refine your search even further by using the AND operator as follows:

```
SELECT * FROM employees WHERE name = 'Wally' AND username = 'wwebmaster'
```

In this case, even if two Wally's were listed in our database, we can assume that they don't have the same username. In this situation, the query returns one result.

Operators for Adv. Querying

The NOT operator

As you just saw with the AND operator, the WHERE clause can be combined with other operators including the NOT operator. The NOT operator displays a record if the condition(s) is NOT TRUE.

```
SELECT * FROM employees WHERE NOT name = 'Wally'
```

In this case, every record is returned from the employees table where the name is not equal to Wally.

Operators for Adv. Querying

The OR and IN operator

If you still wanted to limit your search results but include multiple possibilities in the query, you could rely on the OR operator:

```
SELECT * FROM employees WHERE name = 'Wally' OR name = 'Wilbur'
```

employeeid	name	username	password	email	roleid
1	Wally	wwebmaster	password	wally@vectacorp.com	1
2	Wilbur	wfounder	password	wilbur@vectacorp.com	2

Operators for Adv. Querying

The OR and IN operator

The IN operator allows you to specify multiple values in a WHERE clause similar to OR. The IN operator however, is a shorthand for multiple OR conditions.

```
SELECT * FROM employees WHERE name IN ('Wally', 'Wilbur')
```

In this case, the same result is returned as was the case with the OR operator.

Operators for Adv. Querying

The BETWEEN operator

The BETWEEN operator selects values within a given range. The values can be numbers, text, or dates:

```
SELECT * FROM tickets WHERE ticketdate BETWEEN #07/04/2018# AND #07/10/2018#
```

In this case, any support tickets in the tickets table submitted between July 4th and July 10th 2018 are returned.

Operators for Adv. Querying

The LIKE operator and % wildcard

One of the most common operations on any website is to provide your users with the ability to search for the content that they want. When implementing a custom search within your website, you'll want to use the LIKE operator within your SQL queries. The LIKE operator, in conjunction with the % wildcard, allows you to find data in a table that matches a partial criteria:

```
SELECT * FROM employees WHERE name LIKE 'Wally'
```

employeeid	name	username	password	email	roleid
1	Wally	wwebmaster	password	wally@vectacorp.com	1

Operators for Adv. Querying

The LIKE operator and % wildcard

In a search scenario however, a user might enter a partial term and would expect to see results based on that term. If that were the case, you'd want to use the % wildcard like so:

```
SELECT * FROM employees WHERE name LIKE '%a%'
```

In this case, the % wildcard instructs the query to return any name that contains an 'a' in the value. So the

employeeid	name	username	password	email	roleid
1	Wally	wwebmaster	password	wally@vectacorp.com	1
3	Tina	ttechie	abc123	tina@vectacorp.com	1
4	Agnes	aaccountant	12345	agnes@vectacorp.com	2
5	Damon	ddeveloper	ispeakbinary	damon@vectacorp.com	1

Operators for Adv. Querying

Arithmetic operators

Operator	Description
+	Use when adding fields or values.
-	Used when subtracting fields or values.
*	Use when multiplying fields or values.
/	Used when dividing fields or values.

Operators for Adv. Querying

Comparison operators

Operator	Description
=	Used to determine equality.
>	The greater-than operator is used in WHERE clauses to determine whether a first value is greater than the second, such as this: `SELECT * FROM employees WHERE employeeid > 10` The result returns all the employee ids after 10.
<	The less-than operator is used in WHERE clauses to determine whether a first value is less than the second, such as this: `SELECT * FROM employees WHERE employeeid < 10` The result returns employee ids 1–9.

Operators for Adv. Querying

Comparison operators

Operator	Description
>=	The greater-than-or-equal-to operator is used in WHERE clauses to determine whether a first value is greater than or equal to the second, such as this: `SELECT * FROM employees WHERE employeeid >= 10` The result returns employee ids of 10 and greater.
<=	The less-than-or-equal-to operator is used in WHERE clauses to determine whether a first value is less than or equal to the second, such as this: `SELECT * FROM employees WHERE employeeid <= 10` The result returns all the employee ids between 1 and 10.
<>, !=	When comparing values, use these keywords to check and make sure that one value is not equal to a second value.