

More on MySQL

This section covers some essential MySQL features and operators that help you write more powerful and flexible queries.

1. Logical Operators

Logical operators are used to combine multiple conditions in a `WHERE` clause.

Operator	Description	Example
<code>AND</code>	All conditions must be true	<code>salary > 50000 AND gender = 'Male'</code>
<code>OR</code>	At least one condition is true	<code>gender = 'Male' OR gender = 'Other'</code>
<code>NOT</code>	Reverses a condition	<code>NOT gender = 'Female'</code>

2. Add a Column to an Existing Table

Use `ALTER TABLE` to add a column:

```
ALTER TABLE users
ADD COLUMN city VARCHAR(100);
```

This adds a new column named `city` to the `users` table.

3. Wildcard Operators

Wildcards are used with the `LIKE` operator for pattern matching in text.

Wildcard	Description	Example
<code>%</code>	Matches any sequence	<code>WHERE name LIKE 'A%'</code> (starts with A)
<code>_</code>	Matches a single character	<code>WHERE name LIKE '_a%'</code> (second letter is 'a')

4. LIMIT with OFFSET

`LIMIT` is used to limit the number of rows returned. `OFFSET` skips a number of rows before starting to return rows.

```
SELECT * FROM users
ORDER BY id
LIMIT 5 OFFSET 10;
```

This skips the first 10 rows and returns the next 5.

Alternative syntax:

```
SELECT * FROM users
ORDER BY id
LIMIT 10, 5;
```

This also skips 10 and returns 5 (syntax: `LIMIT offset, count`).

5. DISTINCT Keyword

`DISTINCT` is used to return **only unique values**.

```
SELECT DISTINCT gender FROM users;
```

Returns a list of unique gender values from the `users` table.

6. TRUNCATE Keyword

`TRUNCATE` removes **all rows** from a table, but **keeps the table structure**.

```
TRUNCATE TABLE users;
```

- Faster than `DELETE FROM users`
 - Cannot be rolled back (unless in a transaction-safe environment)
-

7. CHANGE vs MODIFY Column

Both `CHANGE` and `MODIFY` are used to alter existing columns in a table, but they work slightly differently.

CHANGE: Rename and change datatype

```
ALTER TABLE users  
CHANGE COLUMN city location VARCHAR(150);
```

This renames `city` to `location` and changes its type.

MODIFY: Only change datatype

```
ALTER TABLE users  
MODIFY COLUMN salary BIGINT;
```

This changes only the datatype of `salary`.