



Select

```
SELECT column_name FROM table_name;
```

To select all use the * operator.

```
SELECT * FROM table_name
```

Where

Used to filter based on content. If it's text, quotation marks should be used.

Example

```
SELECT column FROM table WHERE column_name='value';
```

You can use the NOT operator to exclude something

```
SELECT column FROM table WHERE NOT column_name='value';
```

Operator than can be used with WHERE:

Operator	Description
=	Equal
<	Less than
>	Greater than
<=	Less than or equal
>=	Greater than or equal

If you need to look for a blank value, simply write two single quotation marks("")

BETWEEN

Can search between a numerical range.

```
SELECT column FROM table WHERE column_name BETWEEN x AND y;
```



LIKE

Can search for patterns. Use it in combination with the % character.

```
SELECT column FROM table WHERE column_name LIKE 'value%'
```

IN

Used to specify multiple values for a column

```
SELECT column FROM table WHERE column_name IN (value1, value2);
```

AND

Used to select as a condition different columns to filter.

```
SELECT column FROM table WHERE column_name1 AND column_name2;
```

OR

Used with different columns or different values within the same column.

```
SELECT column FROM table WHERE column_name1 OR column_name2;
```

```
SELECT column FROM table WHERE column_name1='value1' or  
column_name1='value_number2';
```

Order by

To organize the data in a specific order, you can use ORDER BY. This lets you sort the information in either ascending or descending order. By default, it will sort in ascending order ASC (from smallest to largest), but if you want it to go in the opposite direction, you can use the keyword DESC (from largest to smallest)

Example:

```
SELECT column_name FROM table_name ORDER BY column_name ASC;
```

If you want to sort by multiple columns, you can separate them with a comma.



Update

The update statement in SQL allows us to update values in a certain column.

- Use it adding conditions with the WHERE clause. Same rules apply.
- It's a destructive statement because it modifies data. Caution!

Example:

```
UPDATE table_name SET column_name1='new value1', column_name2='new value2' WHERE condition;
```

After the update statement you may want to use a simple select statement to see what you did.

Alter

With this statement you can alter the table by adding or removing columns, changing the name or the datatype.

Add

Add columns to your table. You need to type the name and the data type

```
ALTER TABLE table_name ADD newcolumn_name datatype;
```

Datatypes and the keyword for them are different on every SQL dialect. Check documentation for more information:

[SQLite](#)

[MySQL](#)

Drop

Delete columns

```
ALTER TABLE table_name DROP COLUMN column_name;
```

Rename

Rename the name of any column

```
ALTER TABLE table_name RENAME COLUMN old_column_name TO newcolumn_name;
```