

# Theory: Basic SELECT statement

🕒 16 minutes

0 / 5 problems solved

Skip this topic

Start practicing

4116 users solved this topic. Latest completion was 30 minutes ago.

## §1. Introduction

You’ve already written your first simple SQL queries, for example:

```
1 | SELECT 'Hello, World!';
```

and

```
1 | SELECT 2.3*4+5;
```

Both of these queries are examples of basic **SELECT statements**. In this topic, we’ll further discuss this first building block for the majority of SQL queries. Yes, you can select more data!

## §2. Tuple

In a **SELECT statement**, you can specify more than one value separated by a comma. For example, the query below selects a string literal, a numeric literal, and an arithmetic expression:

```
1 | SELECT 'Alice', 170, 170*0.393701;
```

Such a set of values (or attributes, fields) is called a **tuple** (record, row). Actually, the result of Hello World query is also a row with a single attribute.

## §3. Alias

In a query, you may specify a name (**alias**) for each attribute of a tuple. To do so, you should use a keyword **AS** followed by a name for the value. If the attribute alias consists of several words or matches an SQL keyword, it should be stated in double-quotes. The query below illustrates the application of aliases:

```
1 | SELECT
2 |     'Alice' AS name,
3 |     170 AS height_in_centimeters,
4 |     170*0.393701 AS "height in inches"
5 | ;
```

The query evaluation result is a tuple with three attributes: “name”, “height\_in\_centimeters”, and “height in inches”. It’s a good idea to specify human-readable aliases; otherwise, your data management system may generate some for you.

As you remember, SQL is designed to process data organized in tables. Actually, the result of the example query is also a table with column names specified in aliases and consisting of only one row.

## §4. Code readability

SQL is case insensitive (keywords may be written in any case), so **SELECT**, **select**, **SeLeCt**, and **seLEct** are all valid. However, it is most common to type keywords in uppercase to emphasize them and improve code readability.

We also recommend using indentation, which is leaving a bit of free space to visually separate the line. There are different code style guides: use your intuition and follow with your team’s preferences.

Compare the following formatting options of the same query. Which one would you prefer to read?

Current topic:

[Basic SELECT statement](#) ...

Topic depends on:

✗ [Expressions](#) ...

Topic is required for:

[SELECT FROM statement](#) ...

Table of contents:

[1 Basic SELECT statement](#)

[§1. Introduction](#)

[§2. Tuple](#)

[§3. Alias](#)

[§4. Code readability](#)

[§5. Conclusion](#)

[Feedback & Comments](#)

```
1 | SELECT 'Bob' AS name, 160 AS "height in centimeters", 160*0.393701 AS "height in i
2 | nches";
3 |
4 | SELECT
5 |     'Bob' AS "name",
6 |     160 AS "height in centimeters",
7 |     160*0.393701 AS "height in inches"
8 | ;
9 |
10 | SELECT
11 |     'Bob'          AS "name",
12 |     160            AS "height in centimeters",
13 |     160*0.393701 AS "height in inches"
14 | ;
```

The third version is much better: you can easily see the number of attributes, their values, and their names. The version in the middle is intermediate in terms of code readability but easier to support than the third one (manual indentation by blanks gets broken each time one changes the code). Of course, in practice, you can choose the code formatting style that your team prefers.

## \$5. Conclusion

Here is a template for a basic SELECT statement: SELECT keyword, list of values to extract with optional aliases for them, and a semicolon to indicate the end of the statement:

```
1 | SELECT val1 [AS name1], ..., valN [AS nameN];
```

Congratulations! Now you know how to select a row with several named attributes of different types. Let’s practice, shall we?

 Report a typo

327 users liked this theory. 6 didn't like it. What about you?



Start practicing

[Comments \(4\)](#)[Hints \(0\)](#)[Useful links \(0\)](#)[Show discussion](#)