**AGGREGATE FUNCTIONS**

**Introduction**

We’ve learned how to write queries to retrieve information from the database. Now, we are going to learn how to perform calculations using SQL.

Calculations performed on multiple rows of a table are called **aggregates**.

In this lesson, we have given you a table named fake\_apps which is made up of fake mobile applications data.

Here is a quick preview of some important aggregates that we will cover in the next five exercises:

* [COUNT()](https://www.codecademy.com/resources/docs/sql/aggregate-functions/count?page_ref=catalog): count the number of rows
* [SUM()](https://www.codecademy.com/resources/docs/sql/aggregate-functions/sum?page_ref=catalog): the sum of the values in a column
* [MAX()](https://www.codecademy.com/resources/docs/sql/aggregate-functions/max?page_ref=catalog)/[MIN()](https://www.codecademy.com/resources/docs/sql/aggregate-functions/min?page_ref=catalog): the largest/smallest value
* [AVG()](https://www.codecademy.com/resources/docs/sql/aggregate-functions/avg?page_ref=catalog): the average of the values in a column
* [ROUND()](https://www.codecademy.com/resources/docs/sql/commands/round?page_ref=catalog): round the values in the column

Let’s get started!

**Instructions**

**1.**

Before getting started, take a look at the data in the fake\_apps table.

In the code editor, type the following:

SELECT \*  
FROM fake\_apps;

What are the column names?

Checkpoint 2 Passed

Hint

The column names are id, name, category, downloads, and price.

**aggregate-functions.sqlite**

SELECT \*

FROM fake\_apps;



**Count**

The fastest way to calculate how many rows are in a table is to use the COUNT() function.

[COUNT()](https://www.codecademy.com/resources/docs/sql/aggregate-functions/count?page_ref=catalog) is a function that takes the name of a column as an argument and counts the number of non-empty values in that column.

SELECT COUNT(\*)  
FROM table\_name;

Here, we want to count every row, so we pass \* as an argument inside the parenthesis.

**Instructions**

**1.**

Let’s count how many apps are in the table.

In the code editor, run:

SELECT COUNT(\*)   
FROM fake\_apps;

Checkpoint 2 Passed

Hint

There are 200 apps.

Common errors:

* Missing parenthesis.
* Missing ;.

**2.**

Add a WHERE clause in the previous query to count how many *free* apps are in the table.

Checkpoint 3 Passed

Hint

Remember the WHERE statement?

The following code should go inside the previous query, before the semicolon:

SELECT COUNT(\*)   
FROM fake\_apps  
WHERE price = 0;

* WHERE indicates we want to only include rows where the following condition is true.
* price = 0 is the condition.

There are 73 free apps in the table.

**count.sqlite**

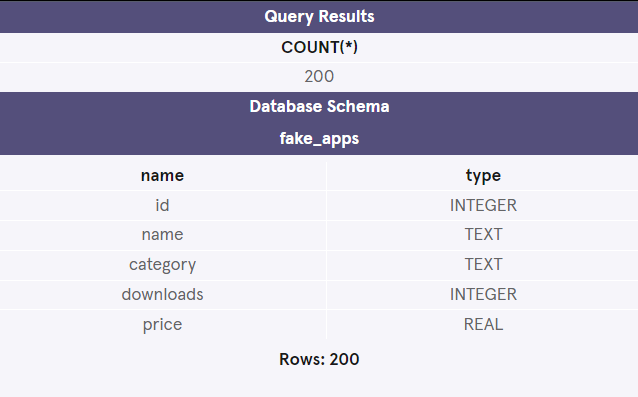
/\*SELECT COUNT(\*)

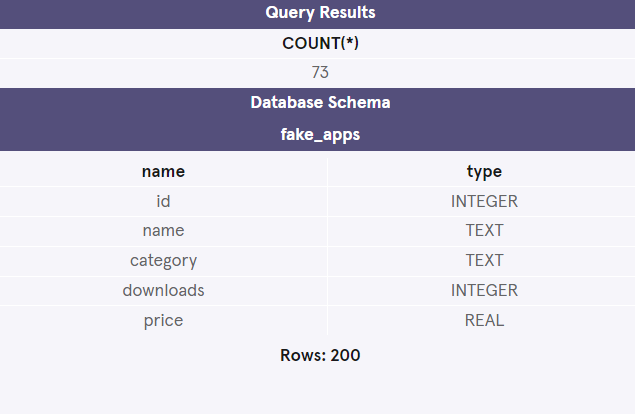
FROM fake\_apps\*/

SELECT COUNT(\*)

FROM fake\_apps

WHERE price = 0.0;

****

****

**Sum**

SQL makes it easy to add all values in a particular column using SUM().

[SUM()](https://www.codecademy.com/resources/docs/sql/aggregate-functions/sum?page_ref=catalog) is a function that takes the name of a column as an argument and returns the sum of all the values in that column.

What is the total number of downloads for all of the apps combined?

SELECT SUM(downloads)  
FROM fake\_apps;

This adds all values in the downloads column.

**Instructions**

**1.**

Let’s find out the answer!

In the code editor, type:

SELECT SUM(downloads)  
FROM fake\_apps;

Checkpoint 2 Passed

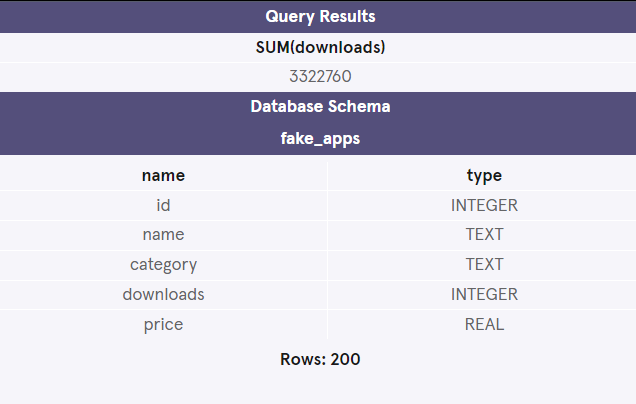
Hint

There are 3,322,760 total downloads.

**sum.sqlite**

SELECT SUM(downloads)

FROM fake\_apps;

****

**Max / Min**

The MAX() and MIN() functions return the highest and lowest values in a column, respectively.

How many downloads does the most popular app have?

SELECT MAX(downloads)  
FROM fake\_apps;

The most popular app has 31,090 downloads!

[MAX()](https://www.codecademy.com/resources/docs/sql/aggregate-functions/max?page_ref=catalog) takes the name of a column as an argument and returns the largest value in that column. Here, we returned the largest value in the downloads column.

[MIN()](https://www.codecademy.com/resources/docs/sql/aggregate-functions/min?page_ref=catalog) works the same way but it does the exact opposite; it returns the smallest value.

**Instructions**

**1.**

What is the least number of times an app has been downloaded?

In the code editor, type:

SELECT MIN(downloads)  
FROM fake\_apps;

Checkpoint 2 Passed

Hint

1,387 downloads.

**2.**

Delete the previous query.

Write a new query that returns the price of the most expensive app.

Checkpoint 3 Passed

Hint

SELECT MAX(price)  
FROM fake\_apps;

$14.99 is the price of the most expensive app.

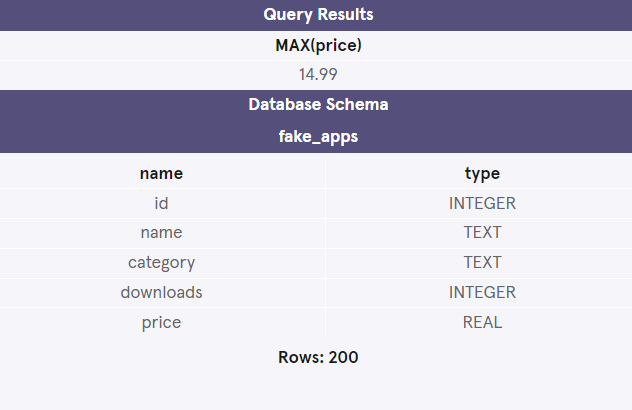
**max.sqlite**

/\*SELECT MIN(downloads)

FROM fake\_apps;\*/

SELECT MAX(price)

FROM fake\_apps;

****