# the Master Course

{CODENATION}

# Backend Development

## Introduction to MySQL

{CODENATION}

{CODENATION}

# Learning Objectives

To set up a cloud-based MySQL Add-On database.

To use MySQL Workbench to interact with a database.

To use SQL to perform CRUD operations on a database.

# What is MySQL?

It is a RDBMS for interacting with a SQL based database.

**Relational Database Management System.**

RDBMS is the basis for SQL, and for all modern database systems such as MS SQL Server, IBM DB2, Oracle, MySQL, and Microsoft Access.

The data is held in tables.

A table is a collection of related data entries held in rows.

{CODENATION}

# What is SQL?

SQL, or Structured Query Language, is a database language used for interacting with Relational Databases.

SQL allows you to write data queries and perform operations on relational databases.

MySQL is a database system that runs on a server.

The data in MySQL is stored in database objects called tables.

A table is a collection of related data entries and it consists of columns and rows.

| CustomerID | CustomerName | ContactName | Address | City | PostalCode | Country |
|---|---|---|---|---|---|---|
| 1 | Alfreds Futterkiste | Maria Anders | Obere Str. 57 | Berlin | 12209 | Germany |
| 2 | Ana Trujillo Emparedados y helados | Ana Trujillo | Avda. de la Constitución 2222 | México D.F. | 5021 | Mexico |
| 3 | Antonio Moreno Taquería | Antonio Moreno | Mataderos 2312 | México D.F. | 5023 | Mexico |
| 4 | Around the Horn | Thomas Hardy | 120 Hanover Sq. | London | WA1 1DP | UK |
| 5 | Berglunds snabbköp | Christina Berglund | Berguvsvägen 8 | Luleå | S-958 22 | Sweden |
| 6 | Blauer See Delikatessen | Hanna Moos | Forsterstr. 57 | Mannheim | 68306 | Germany |

**Tables are broken up into columns called fields.**

**The fields in this Customers table are: CustomerID, CustomerName, ContactName, Address, City, PostalCode and Country.**

{CODENATION}

A database contains one or more tables.

Each table is identified by a name (e.g. "Customers" or "Orders").

Tables contain records (rows) of data.

The biggest differences between SQL and NoSQL (MongoDB), are forced structure, entry relationships and database computation.

SQL data is stored in tables, which can be connected by relationships.

# What are SQL Relationships?

SQL Relationships are common fields that exist in two or more tables.

Usually Primary or Foreign Keys, SQL relationships allow two or more sets of data to be called upon based on their connection.

| AuthorID | AuthorName | Publisher |
|---|---|---|
| 1 | Stephen King | Penguin |
| 2 | J.R.R Tolkien | Harper Collins |
| 3 | Jane Austin | Penguin |
| 4 | Ian Felming | Hachette |
| 5 | Margaret Atwood | Macmillan |
| 6 | Emily Brontë | Harper Collins |

| BookId | AuthorId | bookName |
|---|---|---|
| 1 | 1 | The Shining |
| 2 | 3 | Emma |
| 3 | 2 | The Hobbit |
| 4 | 6 | Wuthering Heights |
| 5 | 1 | It |
| 6 | 3 | Pride and Prejudice |

**Primary Keys are usually an identifying key in a table.**

**They must be unique and can never be null.**

**Foreign Keys reference a Primary Key in another table.**

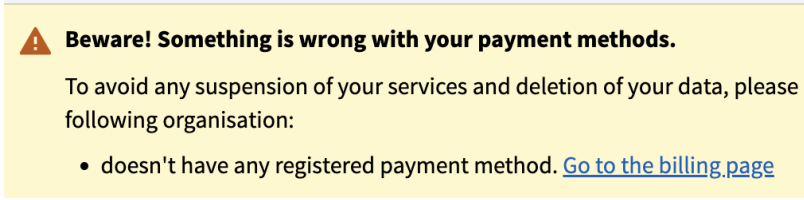
# But first, we need a MySQL database.

clever cloud

Clever-Cloud is a cloud service used to create databases (and other things)

Visit https://clever-cloud.com and sign up, for free.

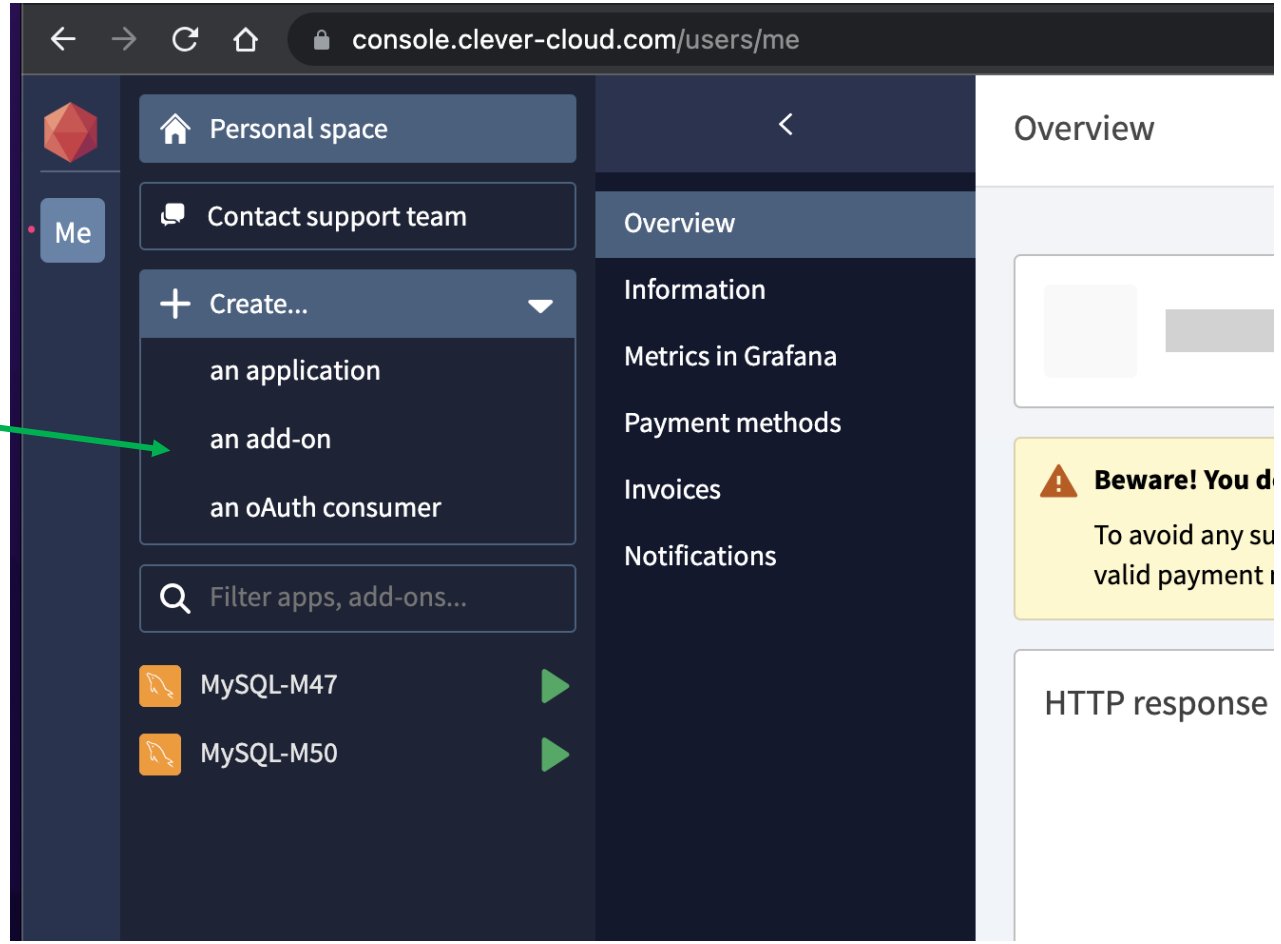
Please **don't enter payment info** or worry about the nag!

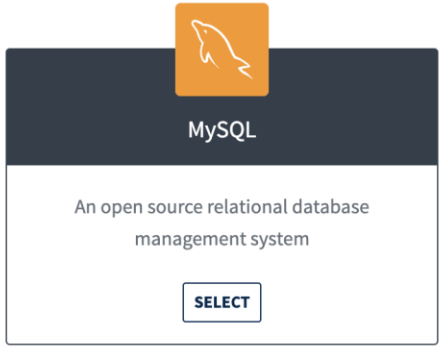


# Clever Cloud - Database

We can create up to 5 free add-ons (databases)

Click ‘Create’, then an add-on.




Select MySQL

# Clever Cloud - Database

You will then see a big list of MySQL kinds.

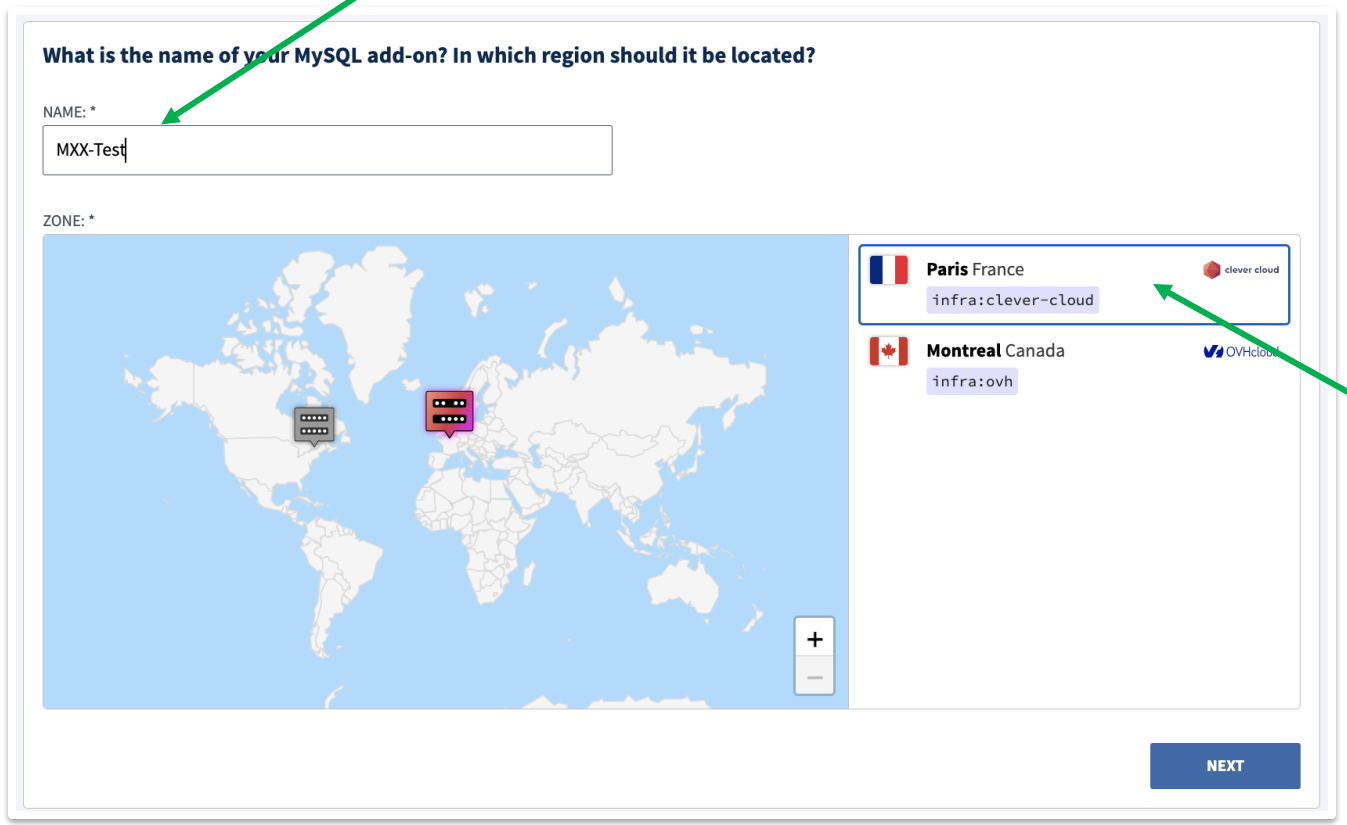
Select the free ‘dev’ option.


There is a max connection limit of 5, and a size limit of 10mb – but it will be fine for our needs.

Click ‘NEXT’ at the bottom right corner.

# Clever Cloud - Database

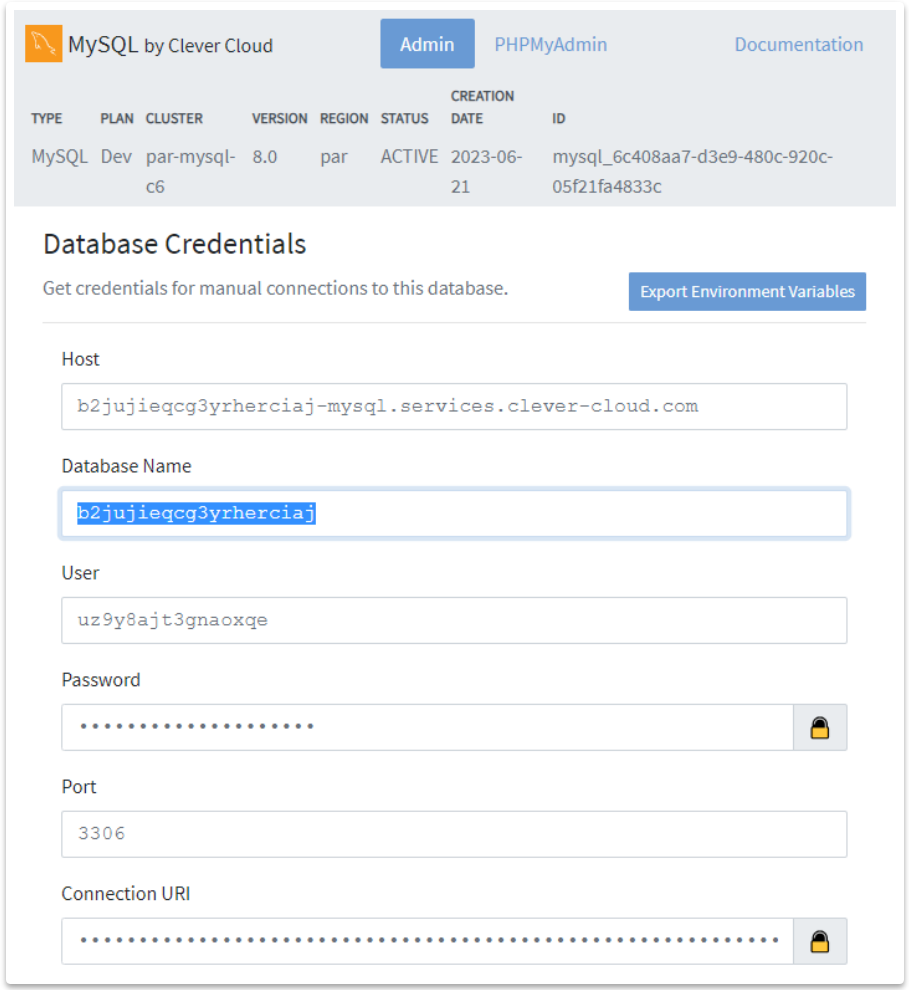
Give your MySQL instance a suitable name.




Choose **Paris**!
(it is closer to us than Canada)

click NEXT

# Clever Cloud - Database




You will be shown your new server information…

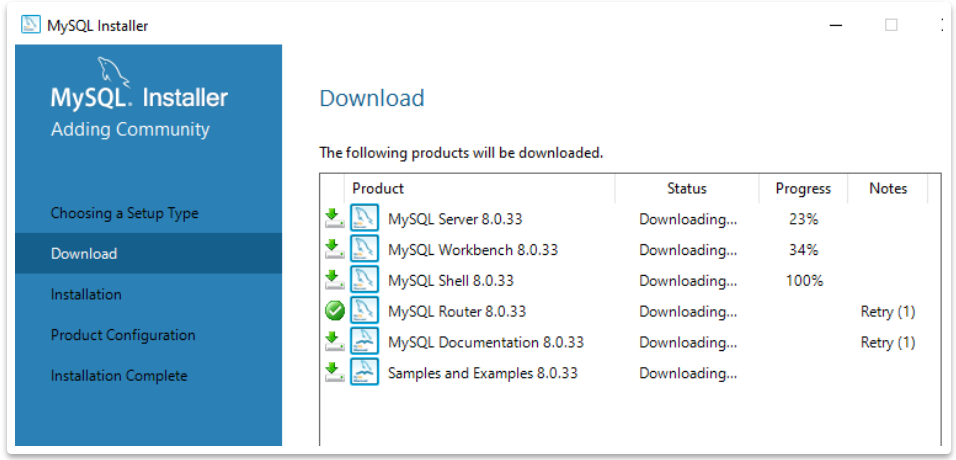
Keep this Browser Tab open as we will need these details, shortly.

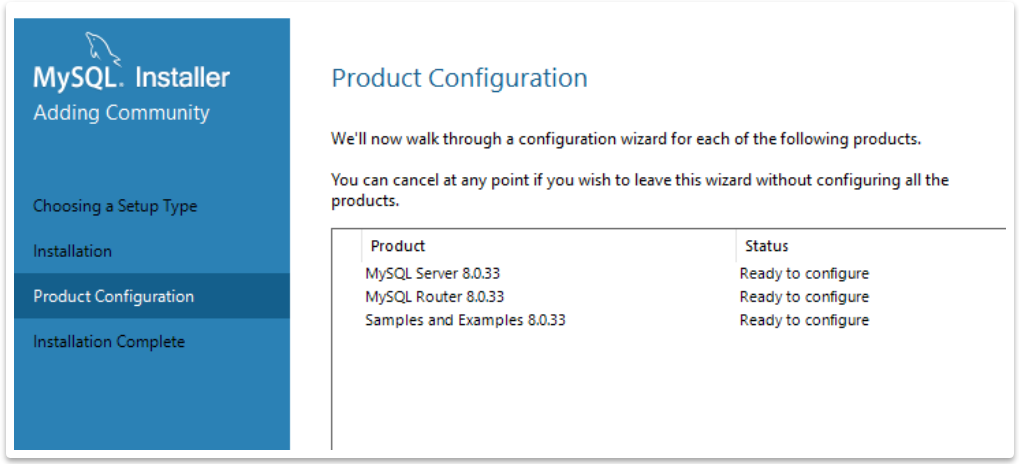
It’s time to install MySQL Workbench …

# MySQL Workbench

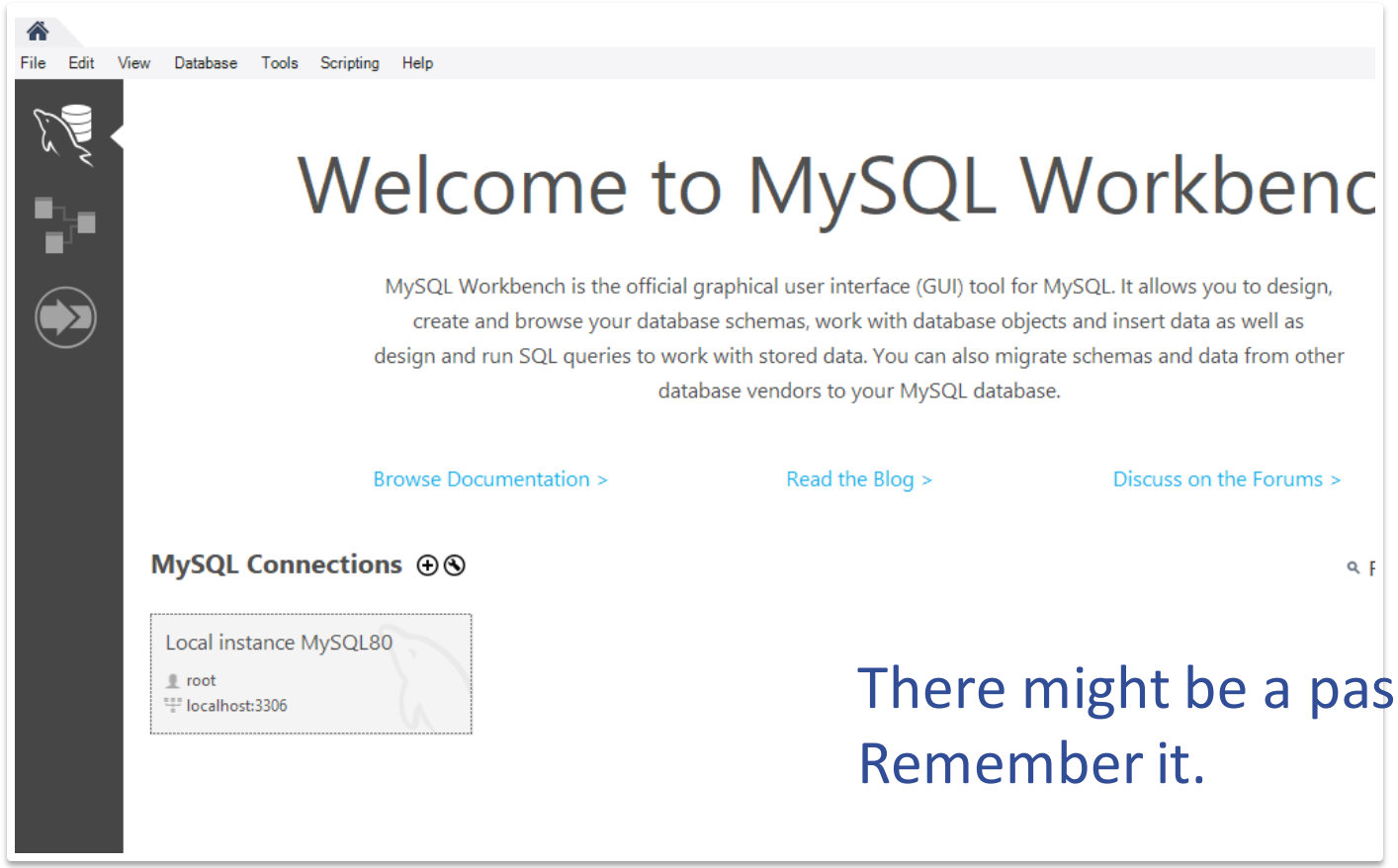
MySQL Workbench is a visual database design tool that integrates SQL development, administration, database design, creation and maintenance into a single integrated development environment for the MySQL database system.

Sounds Great! Let's download it:
https://dev.mysql.com/downloads/windows/installer/8.0.html
Don't bother signing-up to Oracle…



# MySQL Workbench




There might be a password for you to enter. Remember it.

Click ‘Next’ for most other options, until you finally see this screen…

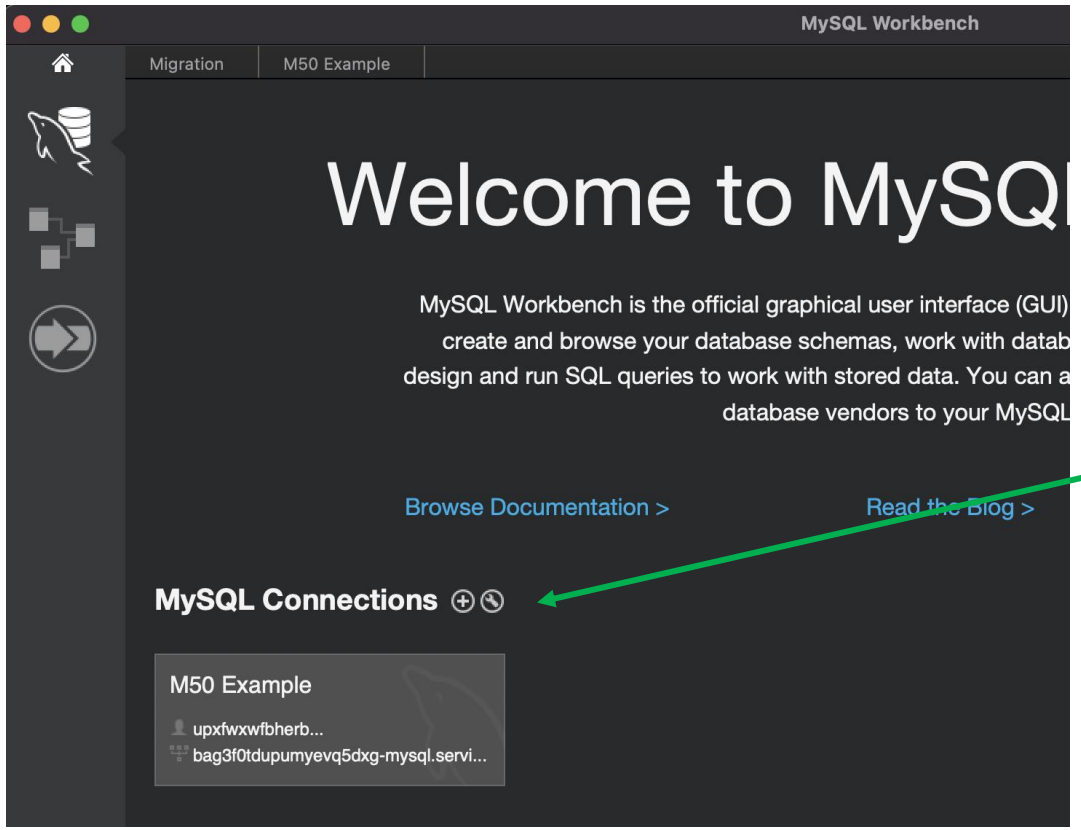
# All okay?

Let's move on to connecting this database up to our MySQL Workbench…

# Clever Cloud ←→ MySQL Workbench

So, our relational database now sits on the cloud (**Clever Cloud)**.
We need a way of connecting to it, and interacting with it, like we did with **MongoDB**.

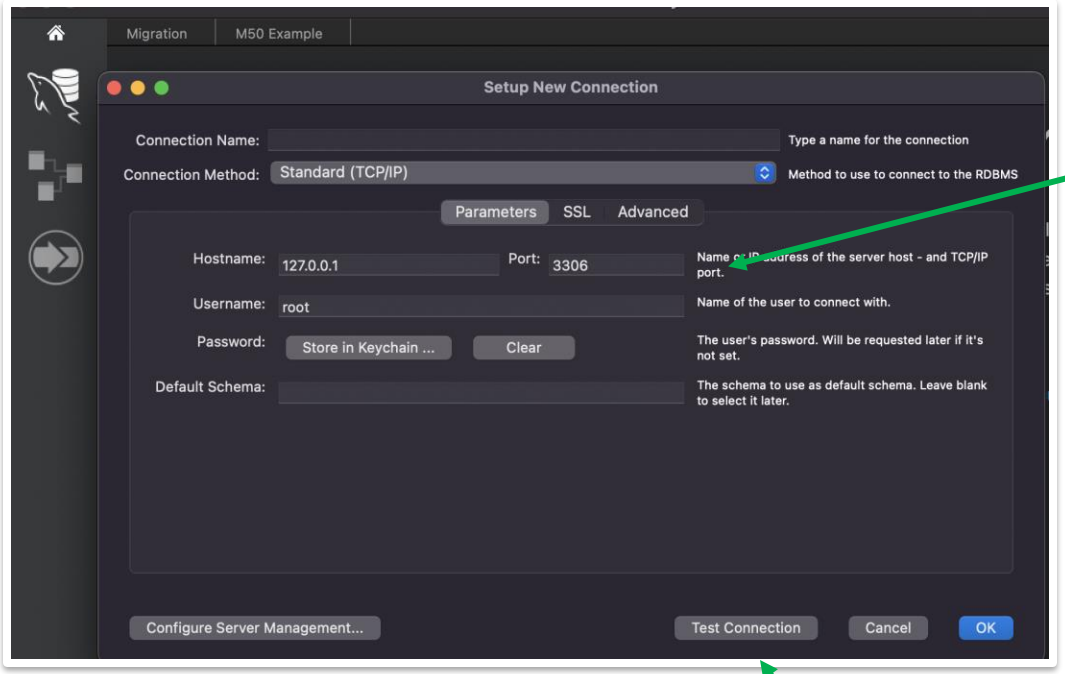
**MySQL Workbench** does this.

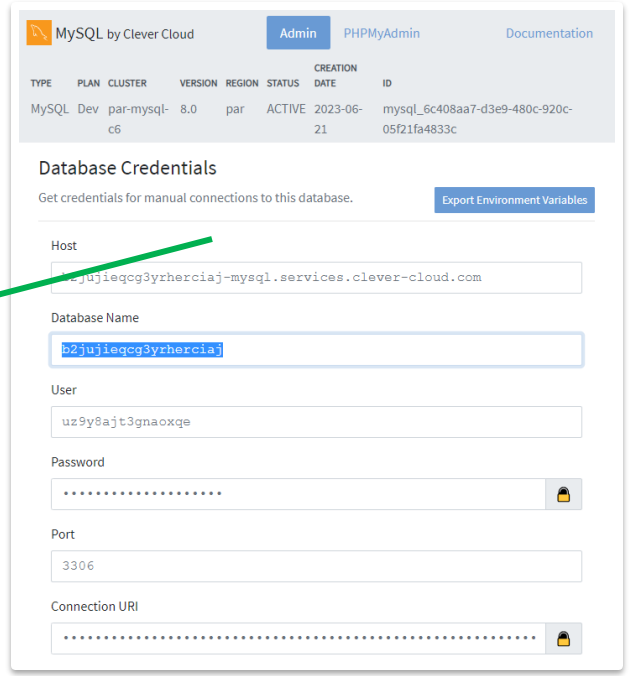



Click the small 'Circle-Plus' next to MySQL Connections

# Clever Cloud ←→ MySQL Workbench

Copy the Clever Cloud Database Credentials across




Test your connection!

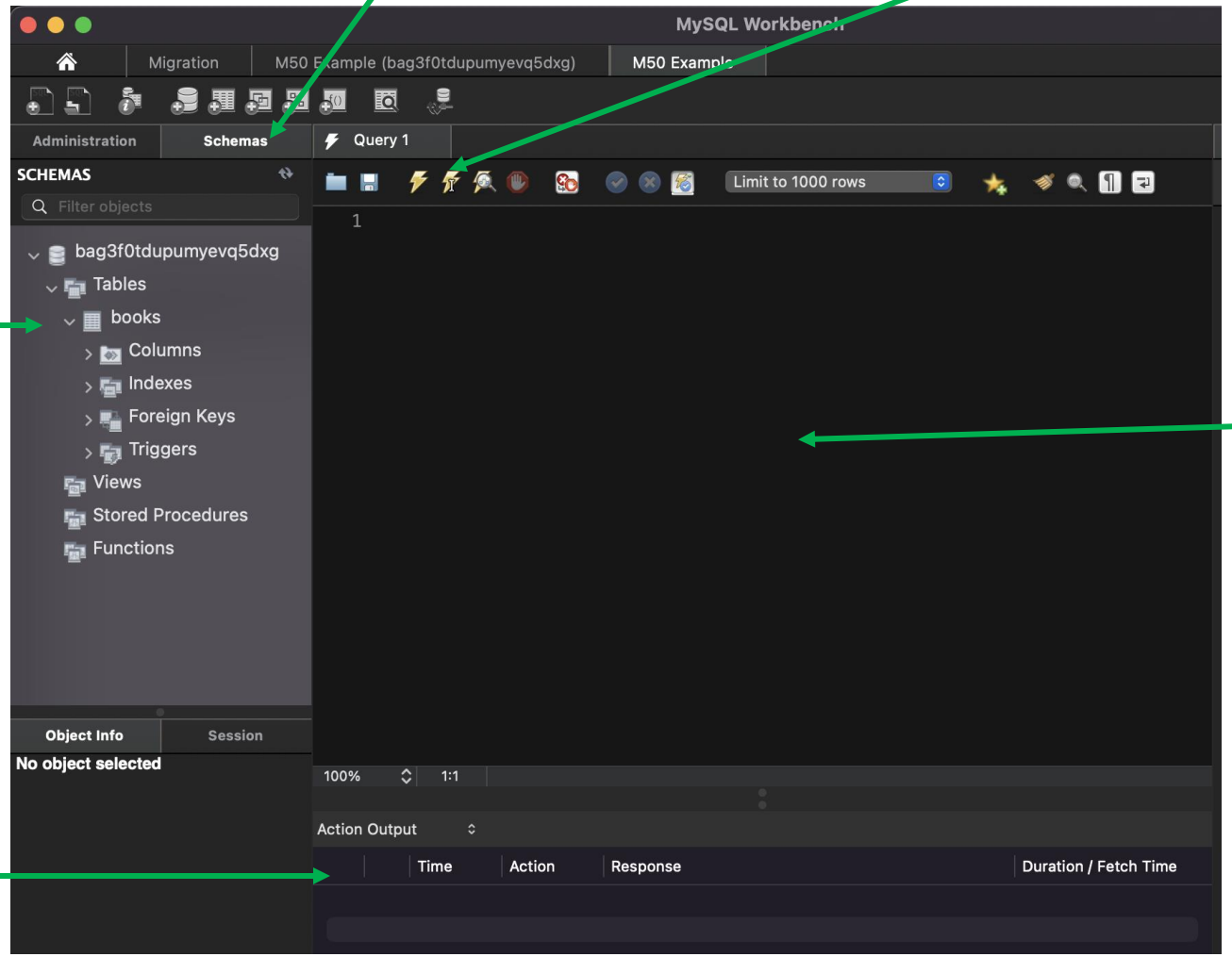
**Well Done!**

You have connected your database to MySQL Workbench

# MySQL Workbench

View your Schema (database)

Run your query!




Your tables, etc

Query Window

Query Output Window

# Great!

We have connected our database up to our Workbench and we can now do things with this empty database ...

# But, what is a database?

# What is a database?

It is a collection of organised data, information and records.

- A bank needs to store the information relating to customer accounts
- A hospital needs to keep data about patients and the medication dispensed
- A university needs to maintain records of its students
- A library needs to keep its book stock updated
- A business website must store its user's credentials and purchase history

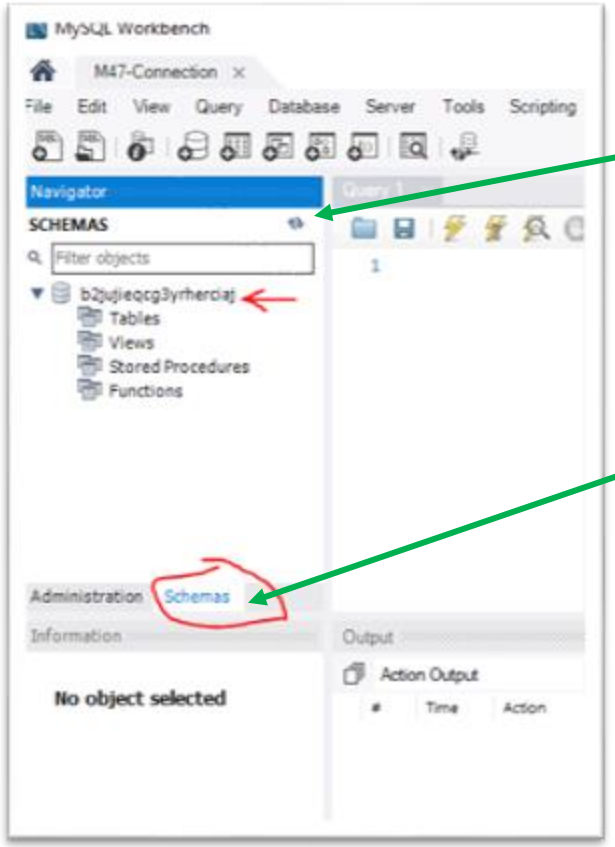
# What is a database?

It needs a database tool like MySQL Workbench to help organise and maintain the data in it.

As a full-stack junior developer, you should know how to use the tools in order to make the database efficient and accurate.

# MySQL Workbench

Click on the **Connection** to Open the SQL Editor.




If you don't see the database, then try the **Refresh** button.

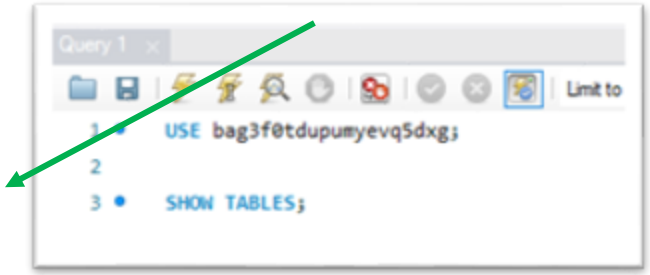
Select **Schemas** to see your database listed above.

The Output window at the bottom tells you things about the success/failure/status of things to do with your SQL query or connection, etc:

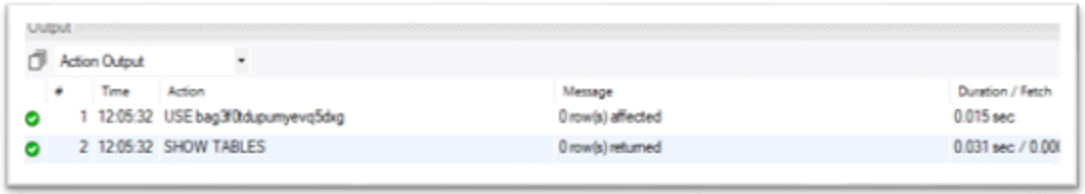
# MySQL Workbench

Let's write our first query in the top 'Query' window, to show all tables in our database.

Execute the query




The results should look like this




We write all SQL code in uppercase letters. SQL was developed in the 1970s when the popular programming languages like COBOL used ALL CAPS, and the convention must have stuck. It does make all commands easier to spot!

CODE ALONG

```
3    -- SHOW TABLES;
```

You can use 2 dashes for comments

# SQL has data types, too ...

### Strings

CHAR()
VARCHAR()
TEXT()
TINYTEXT
LONGTEXT
etc

### Numbers

BIT()
INT()
TINYINT()
BIGINT()
FLOAT()
DECIMAL()
etc

### Other

DATE
TIME
DATETIME()
IMAGE
etc

### More info...

https://www.w3schools.com/sql/sql_datatypes.asp

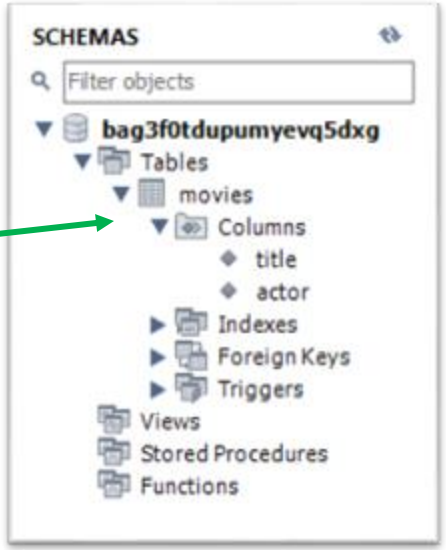
# Let's create a new database table...

```
Query 1
Limit to 1000 rows
1  USE bag3f0tdupumyevq5dxg;
2
3  CREATE TABLE movies (
4      title VARCHAR(255) NOT NULL UNIQUE,
5      actor VARCHAR(255) DEFAULT 'Not specified'
6  );
```

**UNIQUE** = No similar string in another cell for this field

**DEFAULT** = If nothing exists then use this... (This is helpful for the user)

**Run the query to see your new table with its columns.**




CODE ALONG

# Let's insert some data...

INSERT is the CREATE aspect of the CRUD operations

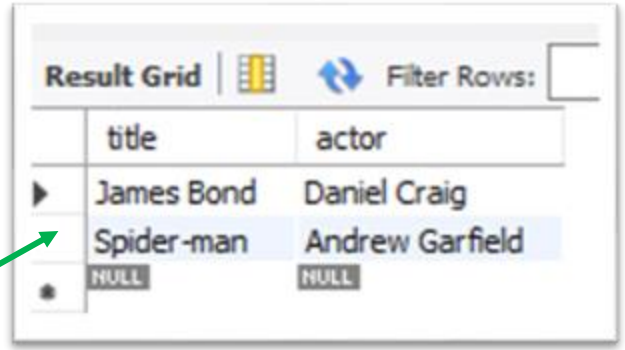
These two separate SQL commands will add two rows to the movies table:

```
8 •   INSERT INTO movies VALUES ('Spider-man', 'Andrew Garfield');
9 •   INSERT INTO movies VALUES ('James Bond', 'Daniel Craig');
```

We can check to see the new rows by SELECTING the table:

```
11 •   SELECT * FROM movies;
```

SELECT is the READ aspect of the CRUD operations




Run the query to see two new rows in the movies table.
(Remember to comment out the previous lines)

CODE ALONG

# Let's insert some data...

Now, if we want to see the DEFAULT response to a SQL statement then try this:

```
14 •   INSERT INTO movies (title) VALUES ('John Wick');
```

| title | actor |
|---|---|
| Andy Dufresne | Tim Robbins |
| Dr Strange | Benedict Cumberbatch |
| James Bond | Daniel Craig |
| John Wick | Not specified |
| Neo | Keanu Reeves |
| Spider-man | Andrew Garfield |
| NULL | NULL |

We are telling it to insert the text value into the column, but without an actor value. (It will use the default value)

Default values are very user-friendly!

CODE ALONG

# All sorted ?

```
11 •    SELECT * FROM movies;
```


This SELECT statement, showed us the data in our table. It auto-sorted the rows by the PRIMARY KEY (title) but we can order results in different ways...

```
14 •    SELECT * FROM movies ORDER BY actor DESC;
```


CODE ALONG

Try this:

```
14 •    SELECT * FROM movies WHERE title = 'Neo';
```

# Activity


Add 5 more films and actors into your movies table. Check that they were successfully added by displaying the table with a SELECT statement..

Add relevant comments to your SQL statements.

# Stretch

Do some research and implement the **UPDATE** and **DELETE** statements into your simple movie table. I recommend that you visit **w3schools.com** for documentation on syntax, etc, but of course other website tutorials are available.
(Can you bypass the safety feature within Workbench that prevents DELETE?)

{CODENATION}

# Learning Objectives

To set up a cloud-based MySQL Add-On database.

To use MySQL Workbench to interact with a database.

To use SQL to perform CRUD operations on a database.