# Configuring goorm FOR mysql

***Credits:*** *these notes were originally prepared by a colleague, Ian Tomey, and modified by myself.*

These notes expand upon the Goorm introduction given last week, to show you how to setup a MySQL database server environment. They also explain how to setup mysqlview, a very basic MySQL database viewer and editor.

You will only have to do this once. Once installed and configured, your container will run MySQL without any manual intervention needed.

## Create Container

First step is to create a container on Goorm. We covered this last week and hopefully you have a container set up already. Please see the week 1 notes at <https://nwcourses.github.io/COM518/topic1.html>.

## Configuring MySQL

At the command prompt in Goorm, enter mysql-ctl start.

This will start the MySQL database server as a background process.

Text

Description automatically generated

Note that the root (admin) database user should not be used for security reasons, so we will go into MySQL and create a new user. So enter mysql at the command prompt to start the MySQL client:

Text

Description automatically generated

From the MySQL command prompt, we create the new user. We then create a database for the new user to use and grant all privileges on that database to the new user.

Enter the queries exactly as below. Change the username and password to whatever you want it to be. This example is creating the database user ‘myuser’ with a password of ‘mypassword’, creating database ‘waddb’, and granting user ‘myuser’ access to the database ‘waddb’:

CREATE USER 'myuser' IDENTIFIED BY 'mypassword';

CREATE DATABASE waddb;  
GRANT ALL PRIVILEGES ON waddb.\* TO 'myuser';  
FLUSH PRIVILEGES;

exit;

The

exit;

exits the MySQL client and returns you to the command prompt.

## Installing mysqlview (basic Mysql web frontend)

mysqlview is a very basic web-based viewer and editor for a MySQL database. We need to install it to our container. It is Node- and Express-based. It will run on port 3100, rather than port 3000 (we will consider this further later on), to allow it to run at the same time as your own applications.

To install mysqlview, please enter the following from the container’s command prompt, to clone it from GitHub, enter the folder it’s installed to, and install its dependencies:

git clone <https://github.com/nickw1/mysqlview.git>

cd mysqlview

npm install

You should then install pm2, a process manager which allows you to run Node applications in the background:

npm install pm2

Then start mysqlview with

npx pm2 start app.mjs

To show that it’s started, you can enter:

npx pm2 list

## ACCESSING mysqlview

In Goorm, flick down the preview tab on the top right and select 'Set Running URL and Port'

Graphical user interface, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

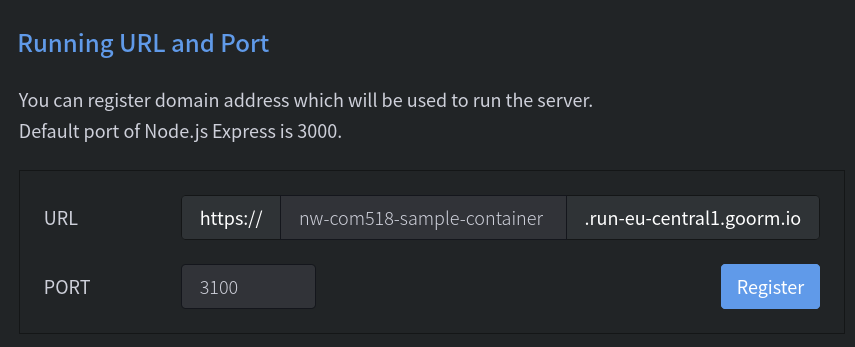
This page shows allows you to register a URL on the internet that exposes an additional port into your container. Note that as we saw last week, Goorm sets up one URL for you by default, which exposes your Node application on port 3000. However, mysqlview is running on a different port, port 3100, so we need to setup a separate url to expose Port 3100 of our container.

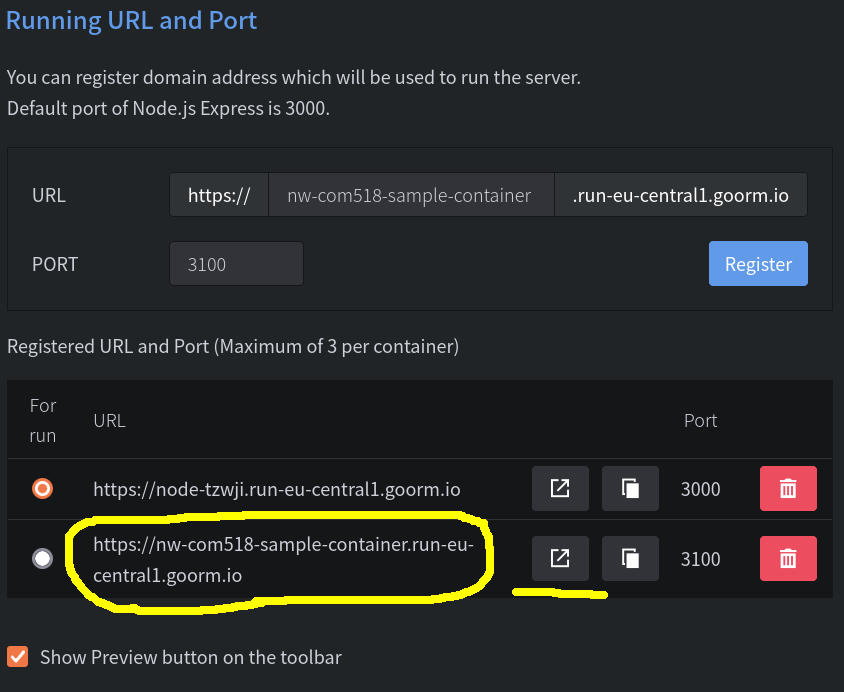
As we saw last week, a URL is already created with the port of 3000 (default for a node project).

Graphical user interface, application

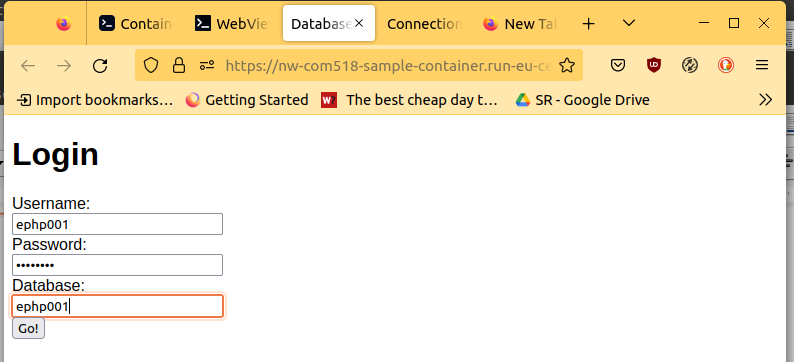
Description automatically generated

But we also need to communicate with the mysqlview webserver on port 3100. So, from the “Set running URL and port” screen, set up a URL on port 3100 as shown below (choose the part before .run-eu-central1.goorm.io, note this has to be unique as it’s a public URL, so maybe choose a nickname personal to you) and hit register.

Launch the URL with the highlighted button in your browser.



It should give us the default mysqlview page as below. Once you have a username, password and database, you should be able to login.



## Setting services to auto run at startup

We can configure our Goorm container to automatically run the Apache web server and the MySQL server at startup. To do this, access the dashboard (the page which first appears when you login to Goorm) and select your container.

Graphical user interface, text, application

Description automatically generated

Click the three dots in the top right and then “Go to settings”.

Edit the init script editable text area to start up MySQL by adding the following to it:

mysql-ctl start

npx pm2 start /workspace/YourContainerName/mysqlview/app.mjs

This will automatically startup MySQL when the container boots. Now whenever yourself *or somebody who have shared the container with who has root level access* the services should be started with the container.