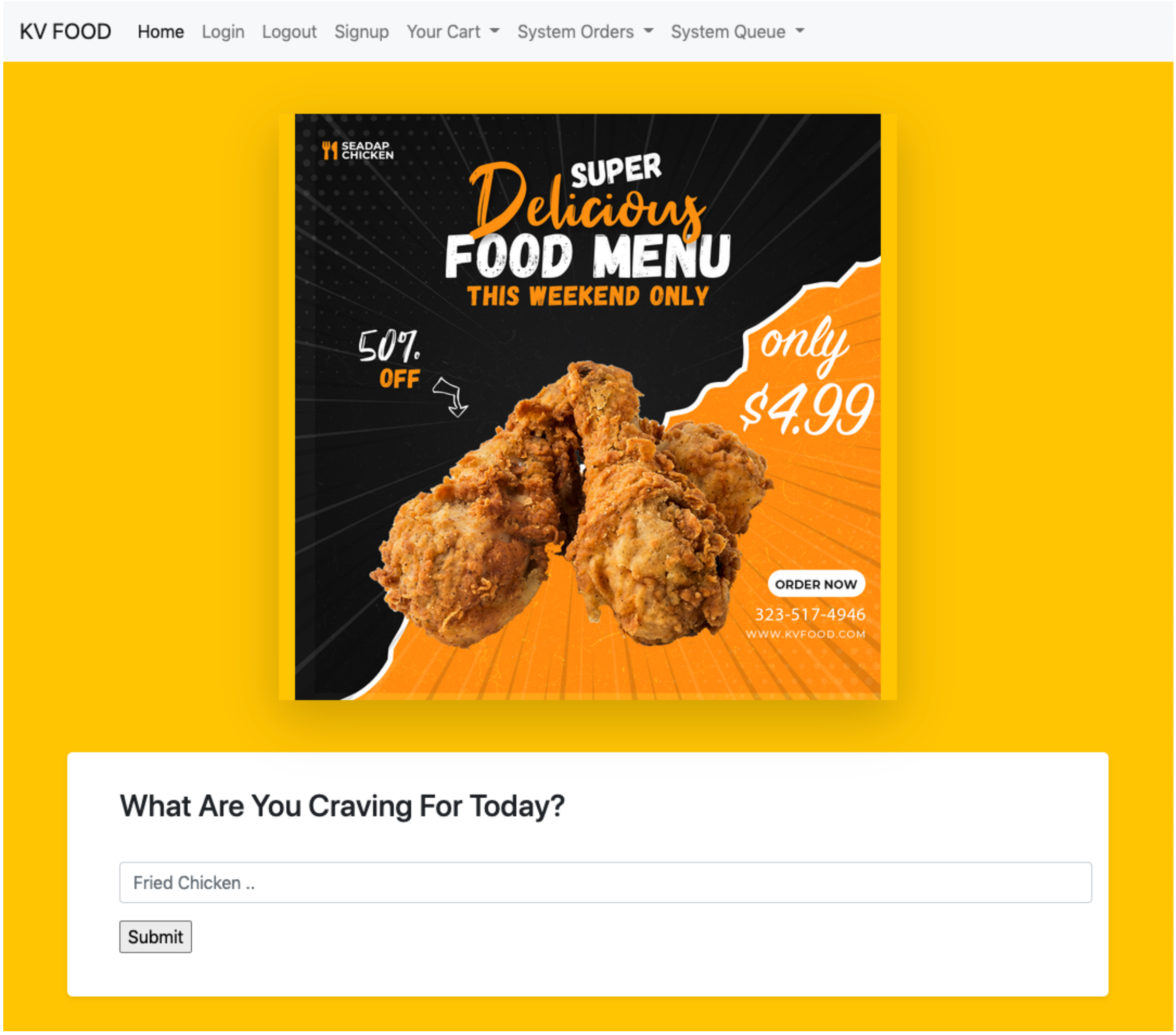
Instructions on

Navigating the Backend Server for KV Food Service



What Are You Craving For Today?

1. Set up MySQL Using docker. - Part A

First and foremost, you need to have docker installed on your computer. You can download a copy of docker depending on your system configuration here:

https://docs.docker.com/get-docker/

Once installed, head over to your terminal and issue the following command

```
docker run --name KV-sql -p 3306 -e MYSQL_ROOT_password=password -d
mysql:latest
```

This line of code would run the latest mysql image that is on your system and runs the image in a container. This container is also known as a *docker container*. If a mysql image is not found on the computer, docker automatically pulls off and downloads the latest mysql image that is found on docker hub. Home to docker's repositories of images.

Because we specified the --name flag, we're giving the docker container a name; in this case , it'll be called *KV-sql*

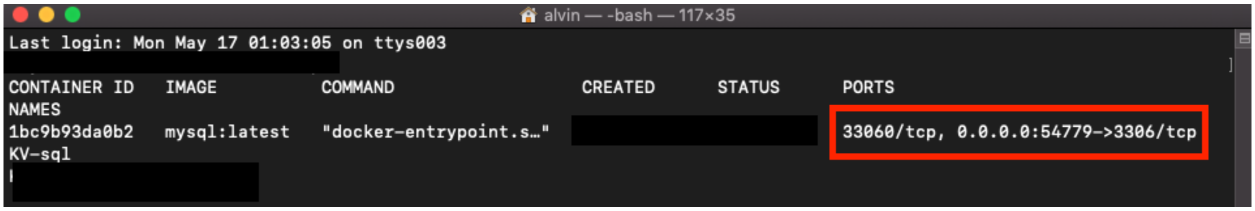
With the -p flag, we're telling mysql to expose it's ports so we can connect to it at a later time.

Please change the MYSQL_ROOT_password=your_desired_password if you like.

Once that is done, issue the following command.

```
docker ps
```

This command will reveal all the containers that are **currently** running.



```
Last login: Mon May 17 01:03:05 on ttys003
[redacted]
CONTAINER ID   IMAGE          COMMAND                  CREATED       STATUS      PORTS
NAMES
1bc9b93da0b2   mysql:latest   "docker-entrypoint.s…"   [redacted]                33060/tcp, 0.0.0.0:54779->3306/tcp
KV-sql
[redacted]
```

Not only does it show you your container's name, it also reveals the important information regarding it's ports. Notice we now have a set of foregin numbers, 54779

As this set of numbers will vary across different computers, it is important to take note of **your own** numbers. Write them down if you have to.

Next, you'll have to set the export path:

```
export PATH=$PATH:/usr/local/mysql/bin
```

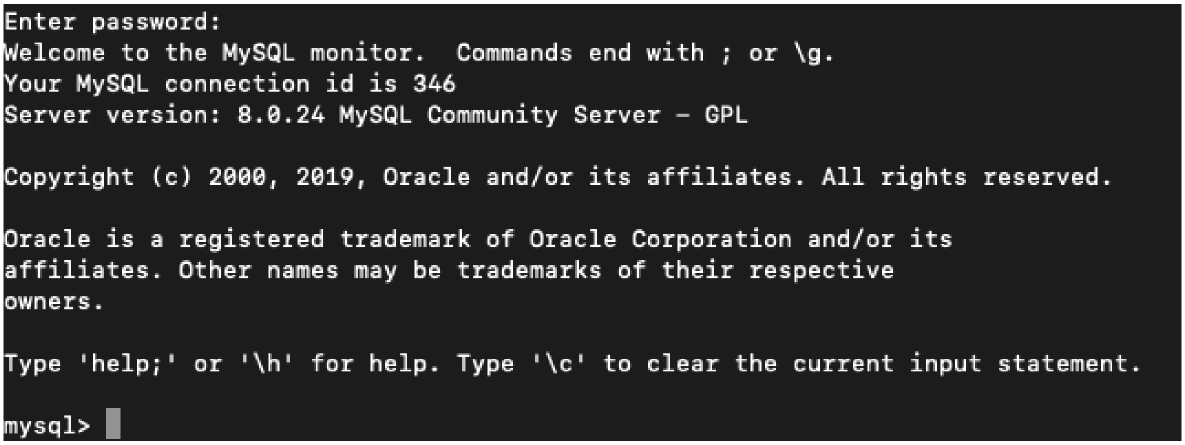
Followed by:

```
mysql -P 54779 --protocol=tcp -u root -p
```

Please change 54779 to your own set of numbers.

This line of code would prompt you for your mySQL password that you've set earlier on. If you copy pasted the codes, the password is *password*.

Press Enter after you've typed them in and you should be prompted with mySQL's welcome message.



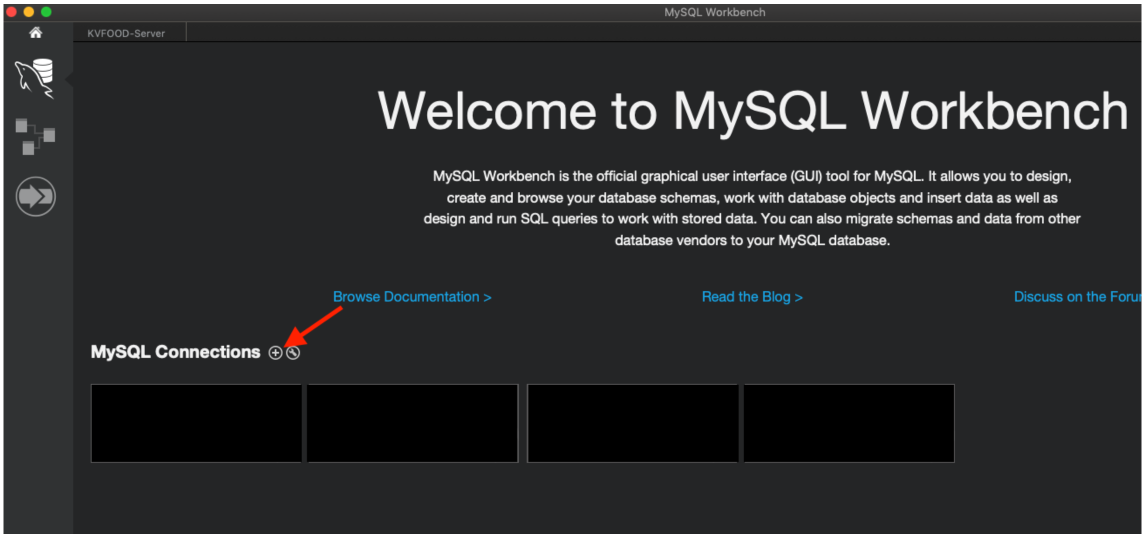
```
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 346
Server version: 8.0.24 MySQL Community Server - GPL

Copyright (c) 2000, 2019, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> 
```

Now that we've verified that we can connect mySQL, go ahead and fire up your mySQL workbench. The mySQL workbench is a nicely made GUI that allows you to make quick changes on a mySQL database such as the one that we've just created.

If you don't have the program installed, you can download one here for your operating system.

https://dev.mysql.com/downloads/workbench/

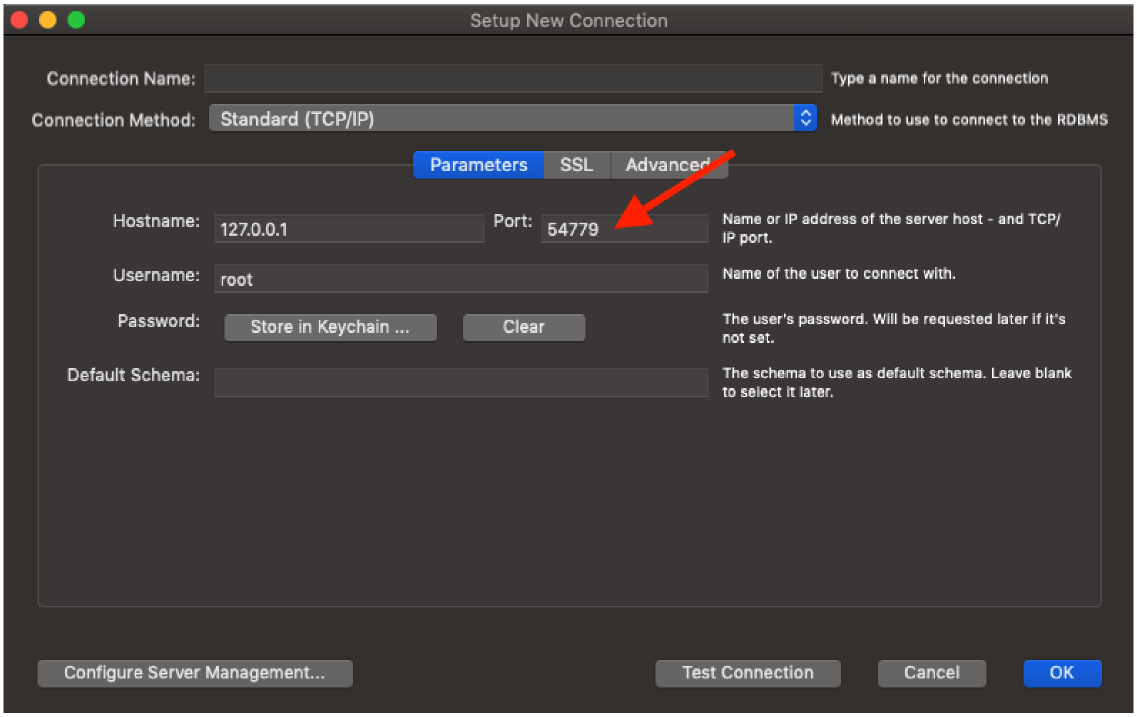
Once your workbench is opened, you'll be greeted with this screen:



Welcome to MySQL Workbench

MySQL Workbench is the official graphical user interface (GUI) tool for MySQL. It allows you to design, create and browse your database schemas, work with database objects and insert data as well as design and run SQL queries to work with stored data. You can also migrate schemas and data from other database vendors to your MySQL database.

Click on the + sign and go ahead and change the port number to your own. In this case, we'll be changing it to 54779.



Connection Name: Type a name for the connection

Connection Method: Standard (TCP/IP) Method to use to connect to the RDBMS

Hostname: 127.0.0.1 Port: 54779 Name or IP address of the server host - and TCP/IP port.

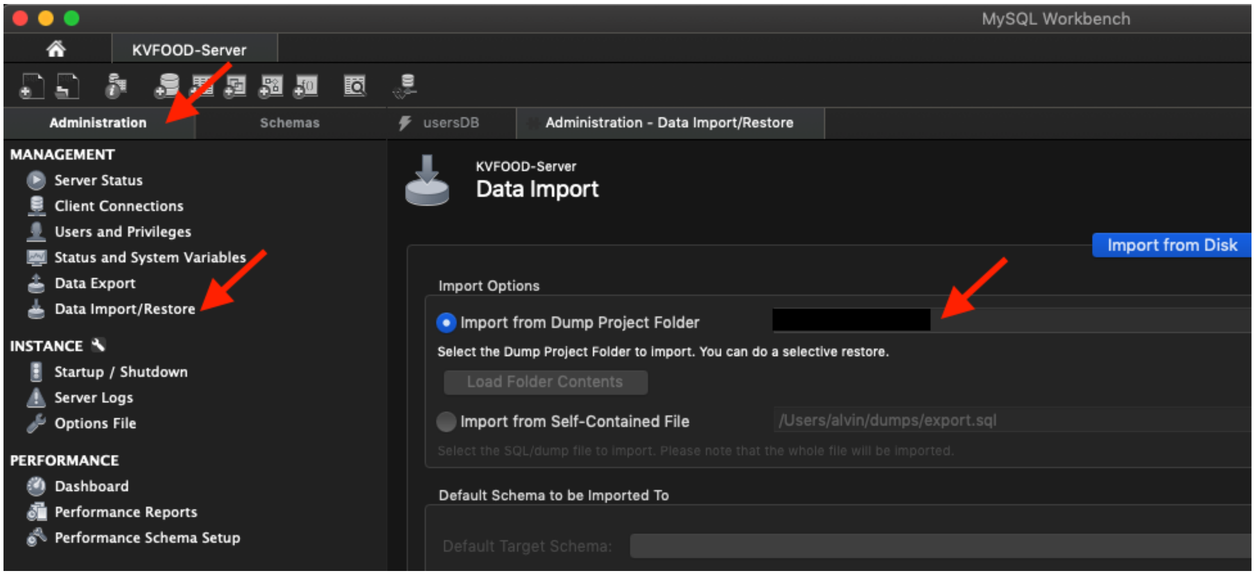
Username: root Name of the user to connect with.

Password: The user's password. Will be requested later if it's not set.

Default Schema: The schema to use as default schema. Leave blank to select it later.

Now that we have the program setup, we'll import some of the test data into mysql.

Click on the Administration Tab, then on Data Import/Restore and then click on the path name to locate a .sql file that should accompany the project's source code.



● Import from Dump Project Folder [redacted]

Select the Dump Project Folder to import. You can do a selective restore.

○ Import from Self-Contained File /Users/alvin/dumps/export.sql

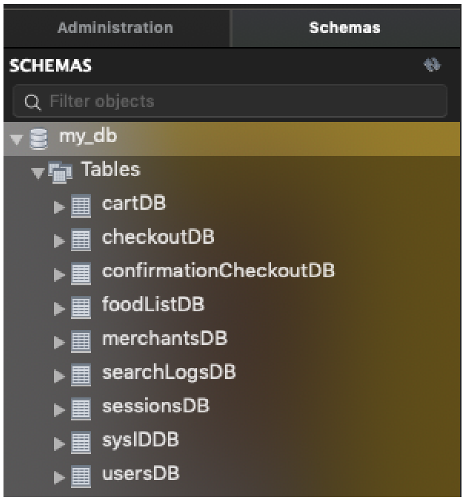
Select the SQL/dump file to import. Please note that the whole file will be imported.

The filename should be KVFoodServer-17May-2021.sql

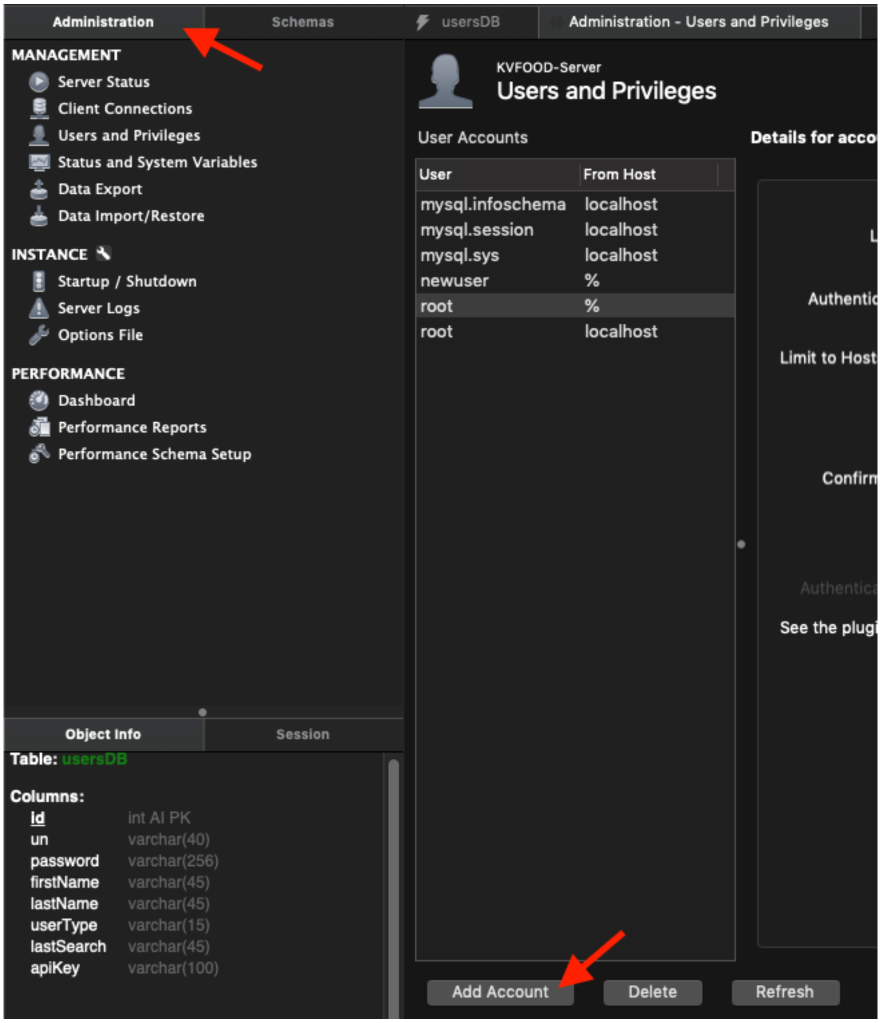
After the import has been done, you are now ready to start testing the server!

If the import has not been successful, please ensure you have a schema selected first.

In the schemas tab, you will find a total of 9 tables with all the data pre populated because of what you've imported.



Once this is done, we're just left with the setting up of users as well as



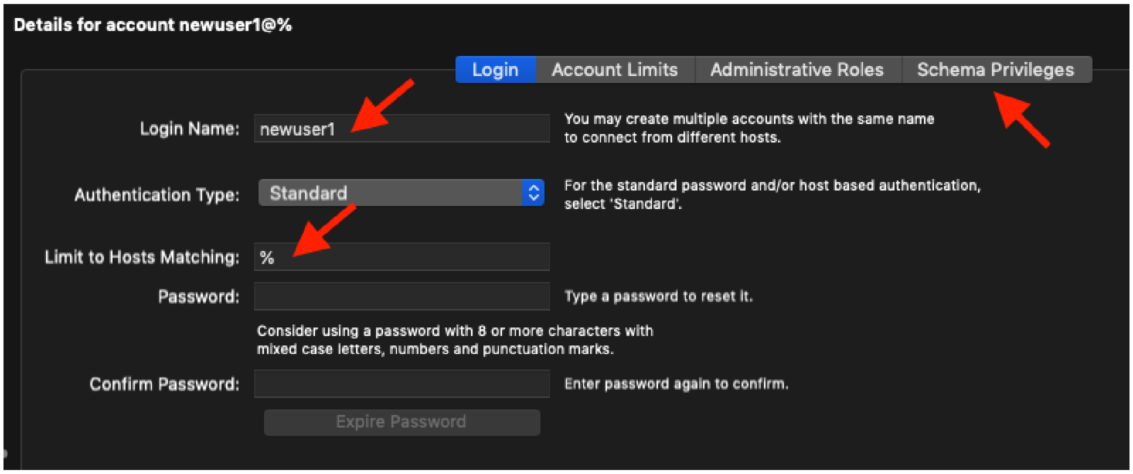
User	From Host
mysql.infoschema	localhost
mysql.session	localhost
mysql.sys	localhost
newuser	%
root	%
root	localhost

Table: usersDB

Columns:

id	int AI PK
un	varchar(40)
password	varchar(256)
firstName	varchar(45)
lastName	varchar(45)
userType	varchar(15)
lastSearch	varchar(45)
apiKey	varchar(100)

Heading to the Administration tab again, click on Add Account.



Details for account newuser1@%

Login Name: newuser1

You may create multiple accounts with the same name to connect from different hosts.

Authentication Type: Standard

For the standard password and/or host based authentication, select 'Standard'.

Limit to Hosts Matching: %

Password:

Type a password to reset it.

Consider using a password with 8 or more characters with mixed case letters, numbers and punctuation marks.

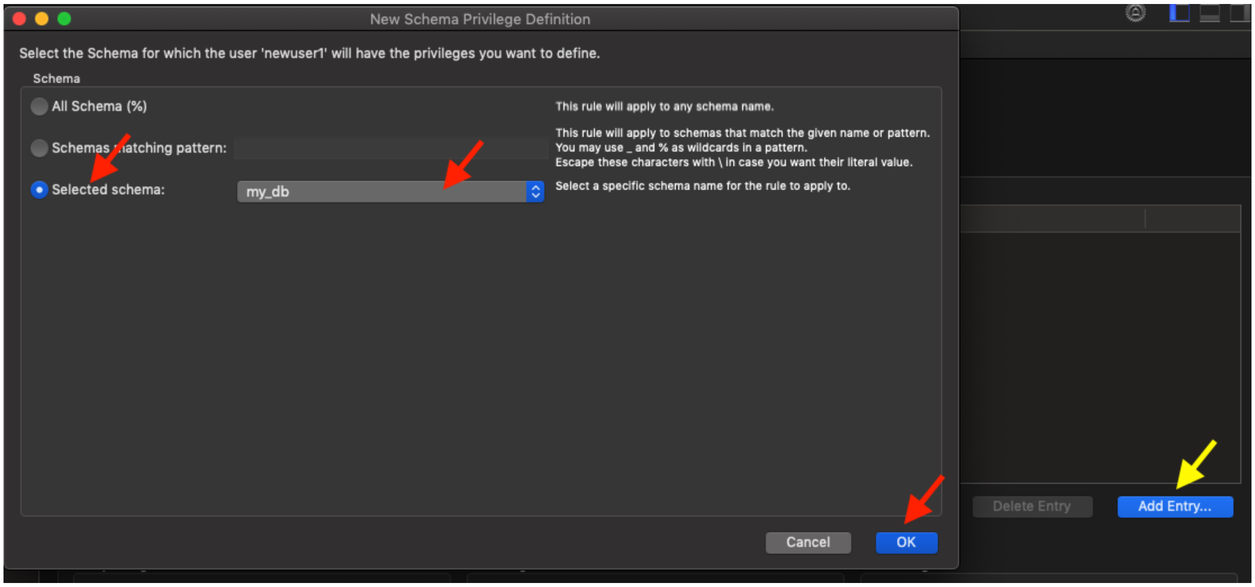
Confirm Password:

Enter password again to confirm.

Type in your desired login username. We've called in newuser1 in this instance. Remember to type % to ensure the user is granted permissions to interact with your mysql database.

Then, proceed to type in your desired Password; a password that you will remember as we will need it very shortly.

Once you're done, head over to the *Schema Privileges* tab.



New Schema Privilege Definition

Select the Schema for which the user 'newuser1' will have the privileges you want to define.

Schema

○ All Schema (%)

This rule will apply to any schema name.

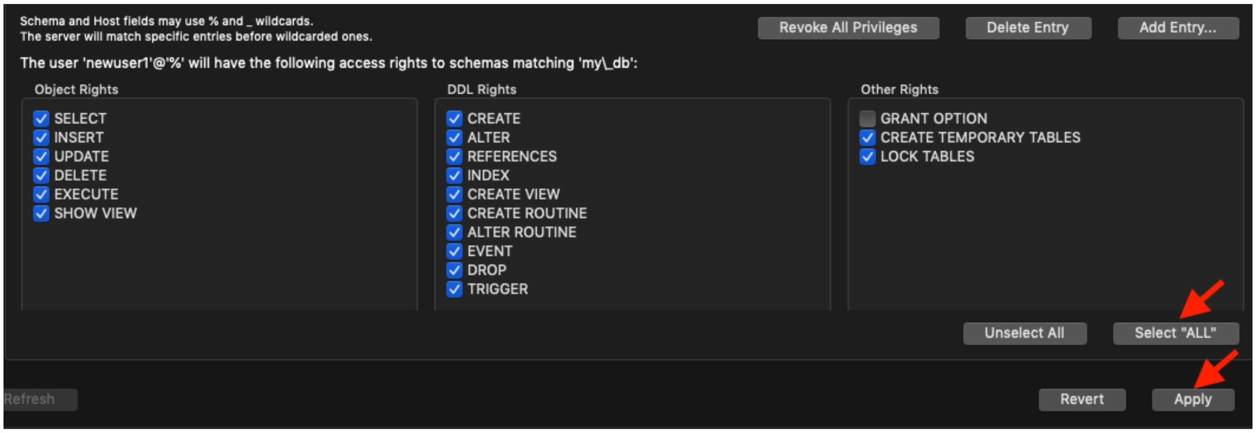
○ Schemas matching pattern:

This rule will apply to schemas that match the given name or pattern.
You may use _ and % as wildcards in a pattern.
Escape these characters with \ in case you want their literal value.

● Selected schema: my_db

Select a specific schema name for the rule to apply to.

Click on Add Entry (yellow arrow) and you'll be greeted with a new popup. Click on Selected schema and choose my_db and then click on the OK button.



Schema and Host fields may use % and _ wildcards.
The server will match specific entries before wildcarded ones.

The user 'newuser1'@'%' will have the following access rights to schemas matching 'my_db':

Object Rights

- ☑ SELECT
- ☑ INSERT
- ☑ UPDATE
- ☑ DELETE
- ☑ EXECUTE
- ☑ SHOW VIEW

DDL Rights

- ☑ CREATE
- ☑ ALTER
- ☑ REFERENCES
- ☑ INDEX
- ☑ CREATE VIEW
- ☑ CREATE ROUTINE
- ☑ ALTER ROUTINE
- ☑ EVENT
- ☑ DROP
- ☑ TRIGGER

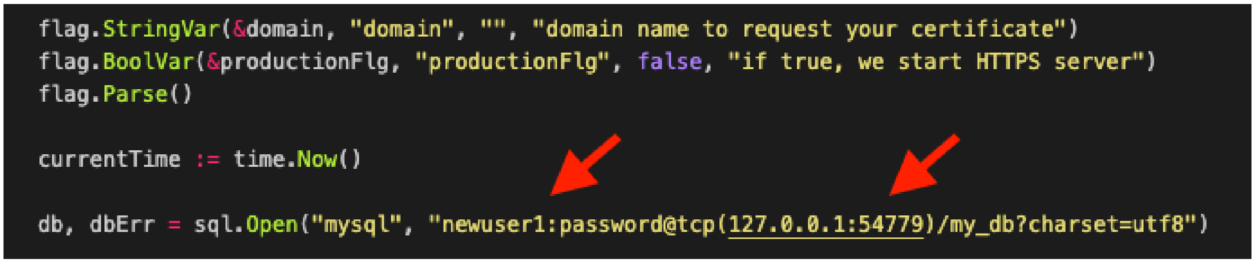
Other Rights

- □ GRANT OPTION
- ☑ CREATE TEMPORARY TABLES
- ☑ LOCK TABLES

Proceed with granting this user privileges. Click on Select All, or customize what you'll want the user to receive access to. Once you're done, click on Apply!

1. Set up MySQL in main.go file - Part B

Part B's relatively easy. Locate the main.go file, open it in your favorite editor and look for these lines of code:



```
flag.StringVar(&domain, "domain", "", "domain name to request your certificate")
flag.BoolVar(&productionFlg, "productionFlg", false, "if true, we start HTTPS server")
flag.Parse()

currentTime := time.Now()

db, dbErr = sql.Open("mysql", "newuser1:password@tcp(127.0.0.1:54779)/my_db?charset=utf8")
```

Change your id, password as well as the port number.

For example, if your id = newuser1 , password = newuser1password, port = 12345, your line of code will look like this.

```
sql.Open("mysql",
"newuser1:newuser1password@tcp(127.0.0.1:12345)/my_db?charset=utf8")
```

Save the file and we're ready to get started!

2. Launch the Back-End Server

CD into the directory where you unzipped the file and the following commands in your terminal:

```
go run .
```

You can use a client such as chrome to access the server at your local address:

localhost

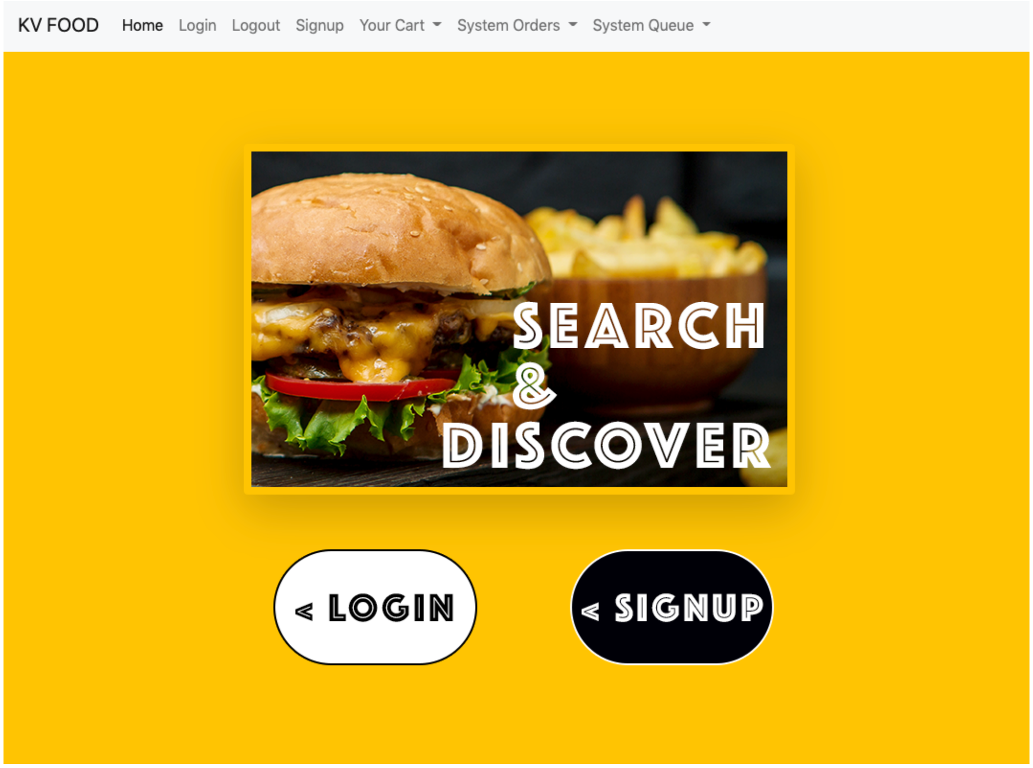
There is no need for a http:// or https:// right in front of the link. It is also important to note that the server is listening at **port 80**

If localhost address doesn't work, please use

localhost:80

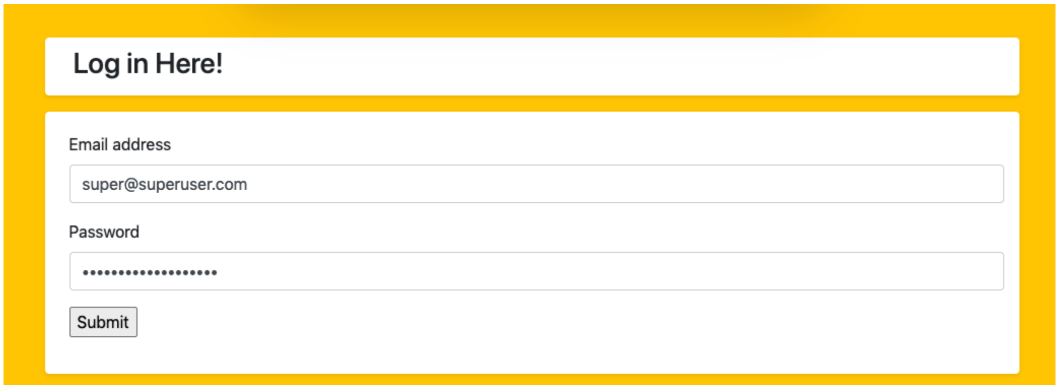
3. **If you start searching, you'll get redirected and get greeted with options**

Since this is all about navigating the backend server, choose login instead.



On the login page, you can use the following id and password:

Id : super@superuser.com
Password: super@superuser.com



Log in Here!

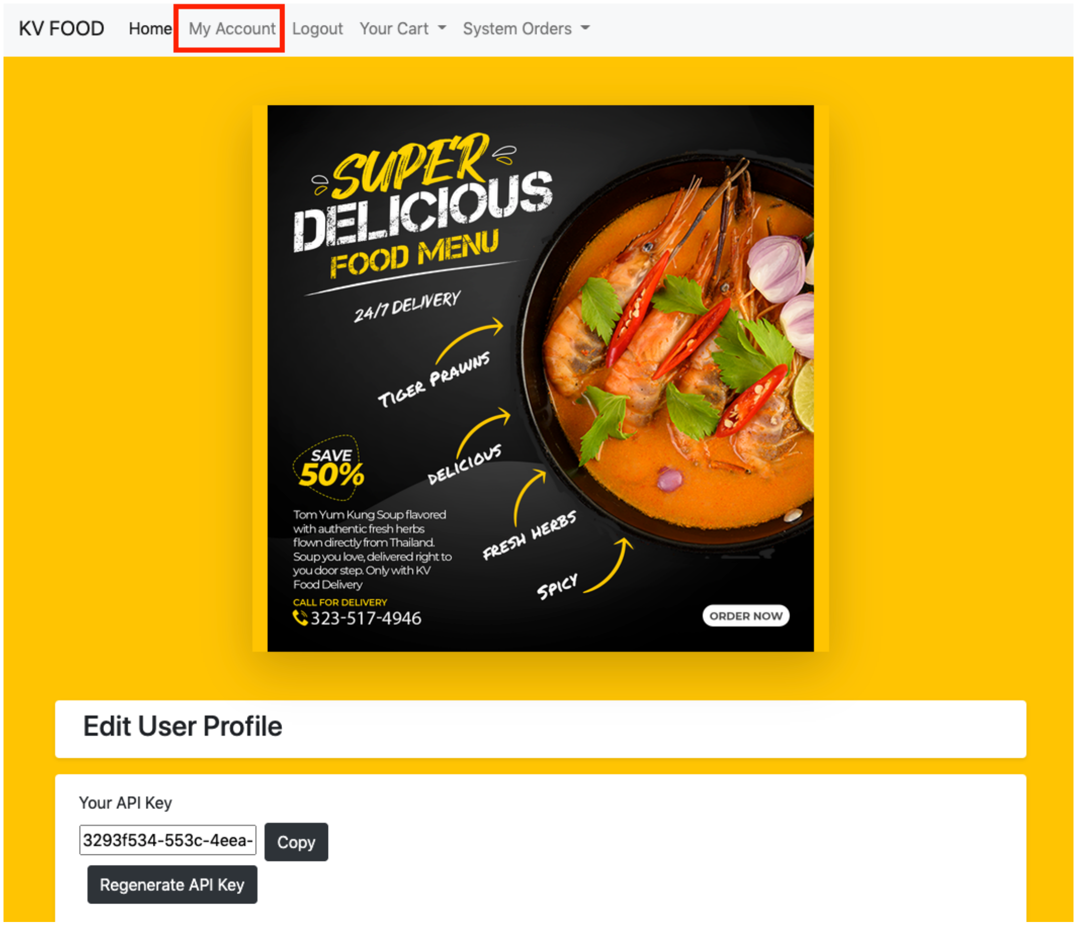
Email address
super@superuser.com

Password
••••••••••••••••••

4. Locating API information

Depending on the type of account privileges, users will be given API keys.

This API key can be located in the *My Account* section.



Edit User Profile

Your API Key

3293f534-553c-4eea-

On this page, it is important to note that the *Regenerate API key* is capable of generating a new API access key for the user.

In other words, if a user has a client program opened and stores the older API key, pressing this button **will immediately** render the client application useless. However, a revalidation of the API key will solve all issues. This is to allow users to revoke their API keys easily yet securely since a login is required before a user can regenerate their API key.

This marks the end of the instructions manual. If you're looking to learn more about how to make orders, updating of cart etc, then please proceed to read the other instructions manual that doesn't involve the backend server, nor the client.

We hope you've enjoyed trying and using the web app as much as I've learnt and spending the time building it.