MariaDB (Mac Setup)

Download Technologies:

- MariaDB: https://mariadb.com/resources/blog/installing-mariadb-10-1-16-on-mac-os-x-with-homebrew/
- DataGrip: https://www.jetbrains.com/datagrip/

Errors downloading/connecting to server/starting server with MariaDB on Mac:

- Refer to the Top Second Answer by Daniel Viglione in this forum: https://stackoverflow.com/questions/51189634/installing-mariadb-with-mysql-on-mac
    - We need to remove MySQL and keep MariaDB

Setup Steps:

Server Setup

1. Download the mentioned technologies and handle any errors.
2. Once MariaDB is installed on your machine, start a server in one terminal:

```
mysqld --port=3306
```

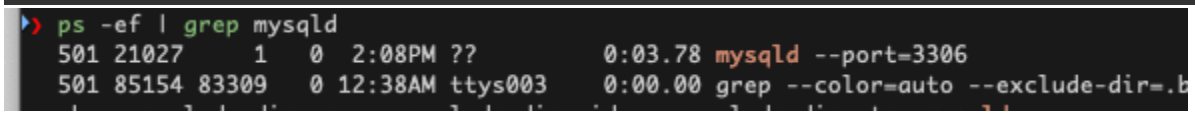
Or use this command if you don't want to specify your ports, but it automatically starts at 3306 or whatever free port if you don't specify:

```
mysql.server start
```

If the machine has an error starting the server session and is unable to run, it may be because the port 3306 is being used. Kill the session happening on port 3306 by following the steps:

- Find the process running on that specific port (3306), enter the command in the terminal:

```
ps -ef | grep mysqld
```



- Identify the process/row where ***mysqld*** and the correct port number (**3306**) exists
- Identify the **PID** of the row (the second column). In this example, the <**PID**> is **21027**.
- Kill the process on the port with the following command:

```
sudo kill -SIGTERM <PID>
```

- You should now be able to run the command ***mysqld --port=3306*** and the server should start after these steps.

3. Enter the server session in another terminal:

```
sudo mysql -u root
```

This will prompt for a password. The password should be your machine/laptop password since we are using root.

Security Privileges

4. Create the following two MySql users. Their passwords should be `cecs491.` Enter the following commands in the same server session to do so:

```
CREATE USER 'HAGSJP.WeCasa.SqlUser'@localhost IDENTIFIED BY 'cecs491';
```

```
CREATE USER 'HAGSJP.WeCasa.SysAdmin@localhost IDENTIFIED BY 'cecs491';
```

The two previous commands should generate this table when you run the SELECT statement after:

```
MariaDB [(none)]> select user,host,password from mysql.user;
+-------------------------+-------------------+-------------------------------------------+
| User                    | Host              | Password                                  |
+-------------------------+-------------------+-------------------------------------------+
| mariadb.sys             | localhost         |                                           |
| root                    | localhost         | invalid                                   |
| githel                  | localhost         | invalid                                   |
|                         | localhost         |                                           |
|                         | macbook-pro-53.lan |                                           |
| HAGSJP.WeCasa.SqlUser   | localhost         | *6F38F5F03BB7940E67A423B952D349D86B3E2CE1 |
| HAGSJP.WeCasa.SysAdmin | localhost         | *6F38F5F03BB7940E67A423B952D349D86B3E2CE1 |
```

5. Grant the following privileges for the two new users just created in the same terminal session (the privileges differ between the SqlUser and SysAdmin):

```
GRANT EXECUTE, PROCESS, SELECT, ALTER, ALTER ROUTINE, CREATE, CREATE ROUTINE,
CREATE TABLESPACE, CREATE TEMPORARY TABLES, CREATE VIEW, DELETE, DROP, EVENT,
INDEX, INSERT, REFERENCES, TRIGGER, UPDATE on *.* TO
'HAGSJP.WeCasa.SqlUser'@'localhost' IDENTIFIED BY 'cecs491' WITH GRANT
OPTION;
```

```
GRANT SHOW DATABASES, SHOW VIEW, EXECUTE, PROCESS, SELECT, ALTER, ALTER
ROUTINE, CREATE, CREATE ROUTINE, CREATE TABLESPACE, CREATE TEMPORARY TABLES,
CREATE VIEW, DELETE, DROP, EVENT, INDEX, INSERT, REFERENCES, TRIGGER, UPDATE
on *.* TO 'HAGSJP.WeCasa.SysAdmin@'localhost' IDENTIFIED BY 'cecs491' WITH
```

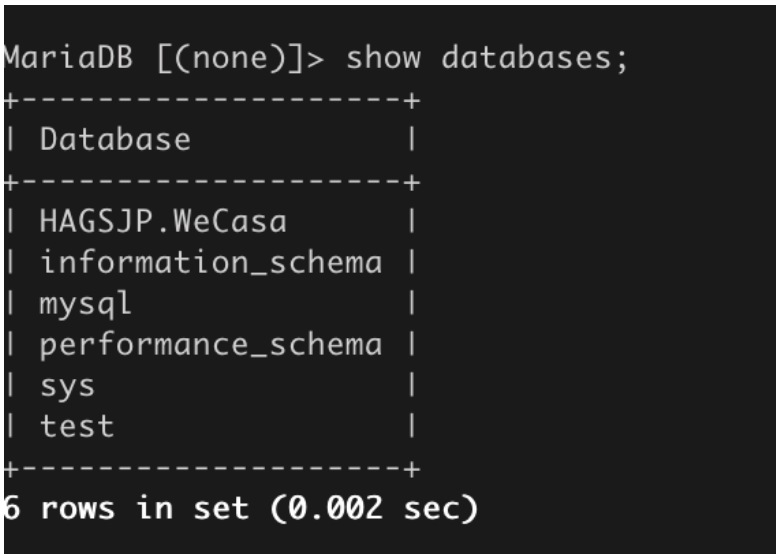
```
GRANT OPTION;
```

Database Setup

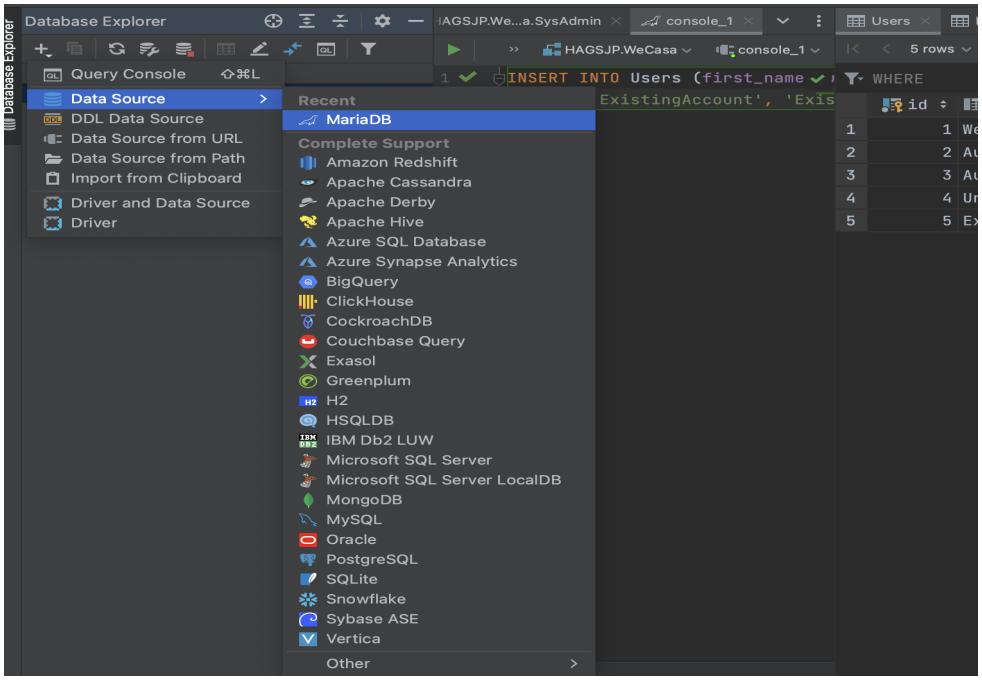
6. Create the WeCasa database in the same terminal session

```
CREATE DATABASE `HAGSJP.WeCasa` /*!40100 COLLATE 'latin1_swedish_ci' */;
```

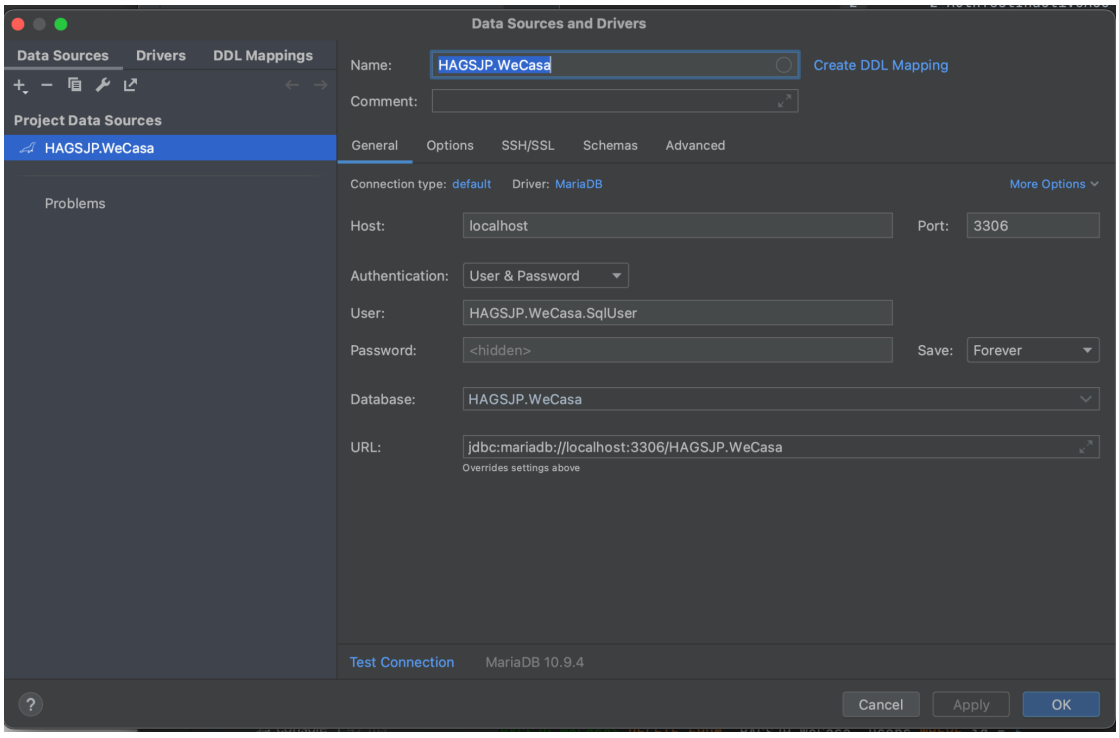
It should show up when you run the next command to verify it got created:



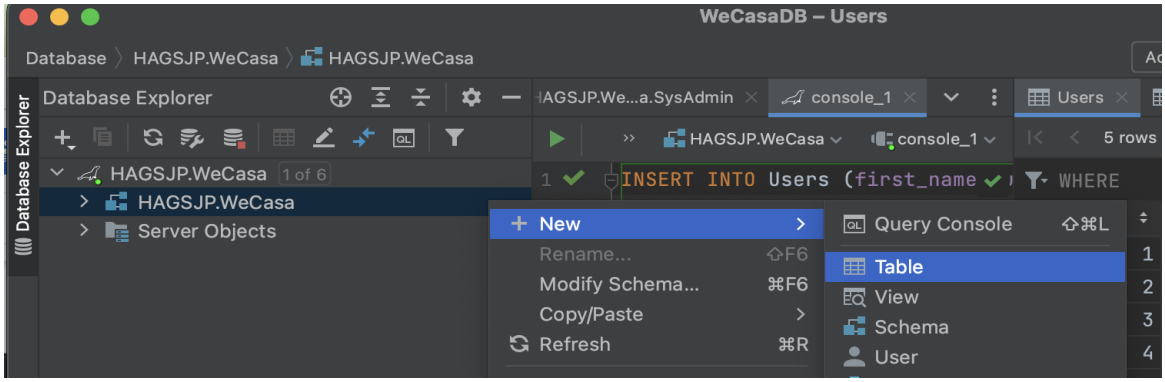
7. In DataGrip, create a new MariaDB data source



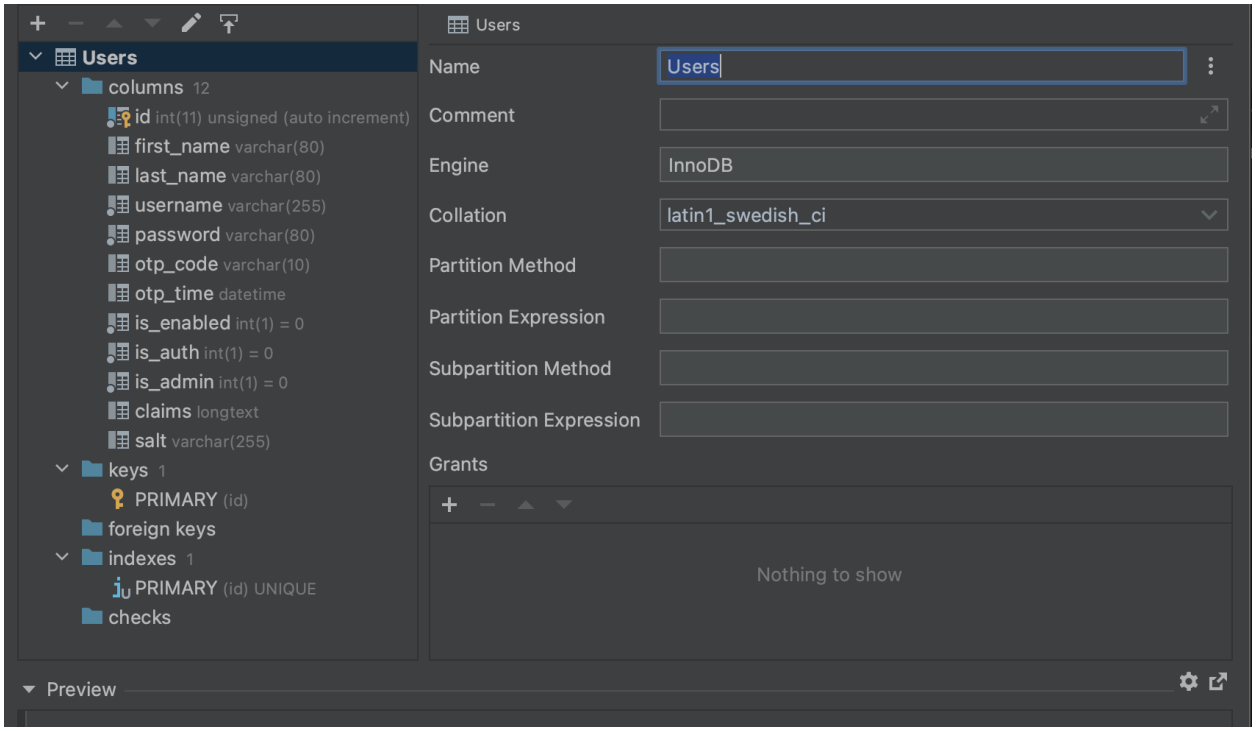
8. Configure the following properties for the new data source:
   Password for HAGSJP.WeCasa.SqlUser is `cecs491`
   Make sure the port number is 3306. If it is not, the code will not work, and the tests will fail.



9. Create the necessary tables (Users & Logs Table):



10. Users Table



```
create table Users
(
    id         INT(11) UNSIGNED auto_increment,
    first_name VARCHAR(80)  null,
    last_name  VARCHAR(80)  null,
    username   VARCHAR(255)  not null,
    password   VARCHAR(80)  not null,
    otp_code   VARCHAR(10)  null,
    otp_time   DATETIME null,
    is_enabled INT(1)      not null,
    is_auth    INT(1)      null,
    is_admin   INT(1)      null,
    claims     LONGTEXT    default NULL null,
    salt varchar(255) default '' not null,
    constraint Users_pk
        primary key (id)
)
    charset = latin1;
```

Use the following CREATE statement in the **Preview** section of the **Table Creation window** to do so:

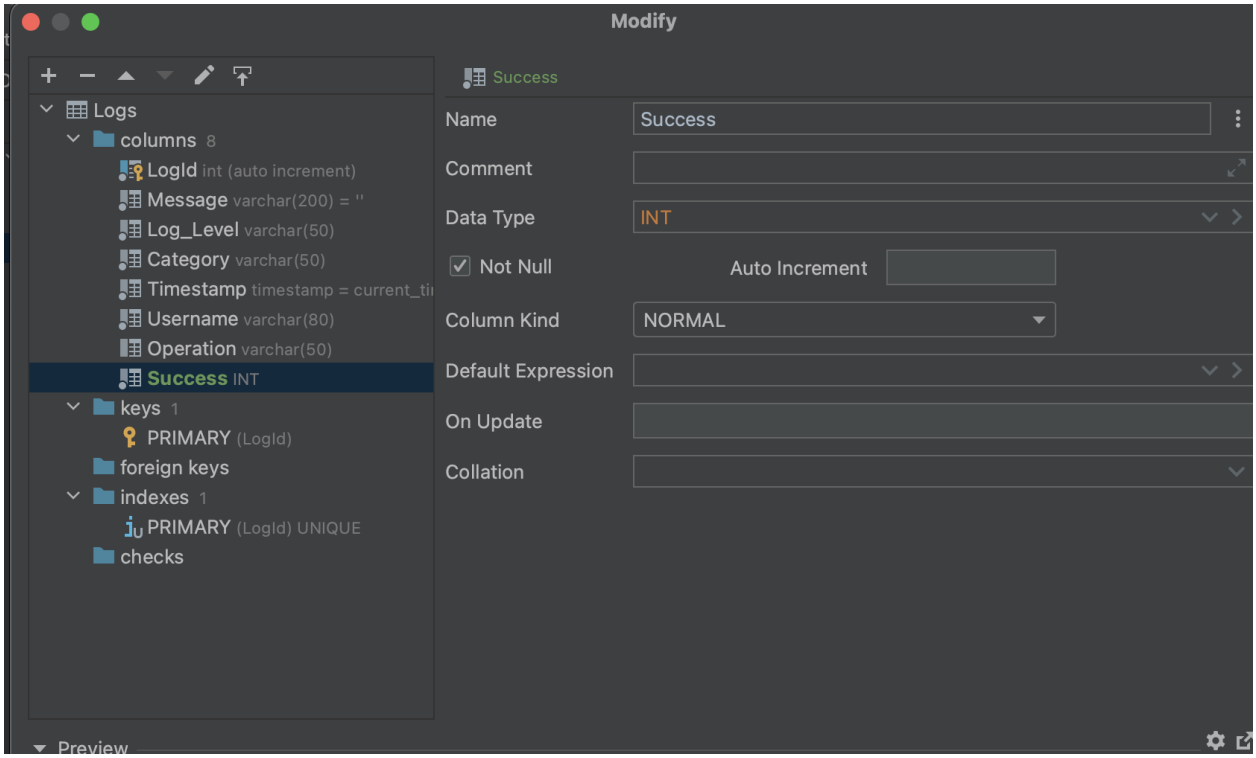
```
create table Users ( id INT(11) UNSIGNED auto_increment, first_name
VARCHAR(80) null, last_name VARCHAR(80) null, username VARCHAR(255) not null,
password VARCHAR(80) not null, otp_code VARCHAR(10) null, otp_time DATETIME
null, is_enabled INT(1) not null, is_auth INT(1) null, is_admin INT(1) null,
      claims LONGTEXT default NULL null,        salt varchar(255) default ''
not null, constraint Users_pk primary key (id) ) charset = latin1;
```

Make sure to specify the lengths of each field because DataGrip throws errors and will not save your table if the lengths are not defined.

For example, instead of just leaving the data type as VARCHAR, I changed it to VARCHAR(length), etc. to avoid the error in this table.

The same type of length specification applies to the Logs table.

11. Logs Table




```
create table Logs
(
    LogId     INT(11) auto_increment,
    Message   VARCHAR(200)   NOT NULL,
    Log_Level VARCHAR(50)    not null,
    Category  VARCHAR(50)    not null,
    Timestamp TIMESTAMP not null,
    Username  VARCHAR(80)    not null,
    Operation VARCHAR(50) null,
    Success   INT(1) not null,
    constraint Logs_pk
        primary key (LogId)
)
    charset = latin1;
```

Use the following CREATE statement in the **Preview** section of the **Table Creation window** to do so:

```
create table Logs ( LogId INT(11) auto_increment, Message VARCHAR(200) NOT NULL, Log_Level VARCHAR(50) not null, Category VARCHAR(50) not null,
```

```
Timestamp TIMESTAMP not null, Username VARCHAR(80) not null, Operation
VARCHAR(50) null, Success INT(1) not null, constraint Logs_pk primary key
(LogId) ) charset = latin1;
```

Inserting Database Data for Running Projects

Users Table

Run the following commands in a **DataGrip query console** within the HAGSJP.WeCasa schema to insert the preliminary data for the Users Table in order for our project/tests to run:

```
INSERT INTO Users (first_name, username, password, is_enabled, is_auth)
VALUES ('WeCasaSysAdmin', 'wecasa@gmail.com', 'Have@GreatSumm3rJapan!', 1, 0);

INSERT INTO Users (username, password, is_enabled, is_auth, is_admin) VALUES
('AuthTestInactiveAcc@gmail.com', 'P@ssw0rd!', 0, 0, 0);

INSERT INTO Users (username, password, is_enabled, is_auth, is_admin)
VALUES ('UpdateClaims@gmail.com', 'P@ssw0rd!', 1, 1, 0);

INSERT INTO Users (username, password, is_enabled, is_auth, is_admin)
VALUES ('AuthTestSuccess@gmail.com', 'P@ssw0rd!', 1, 1, 0);

INSERT INTO Users (username, password, is_enabled, is_auth, is_admin) VALUES
('UnauthorizedTest@gmail.com', 'P@ssw0rd!', 1, 1, 0);

INSERT INTO Users (username, password, is_enabled, is_auth, is_admin) VALUES
('ExistingEmail@gmail.com', 'P@ssw0rd!', 0, 0, 0);
```

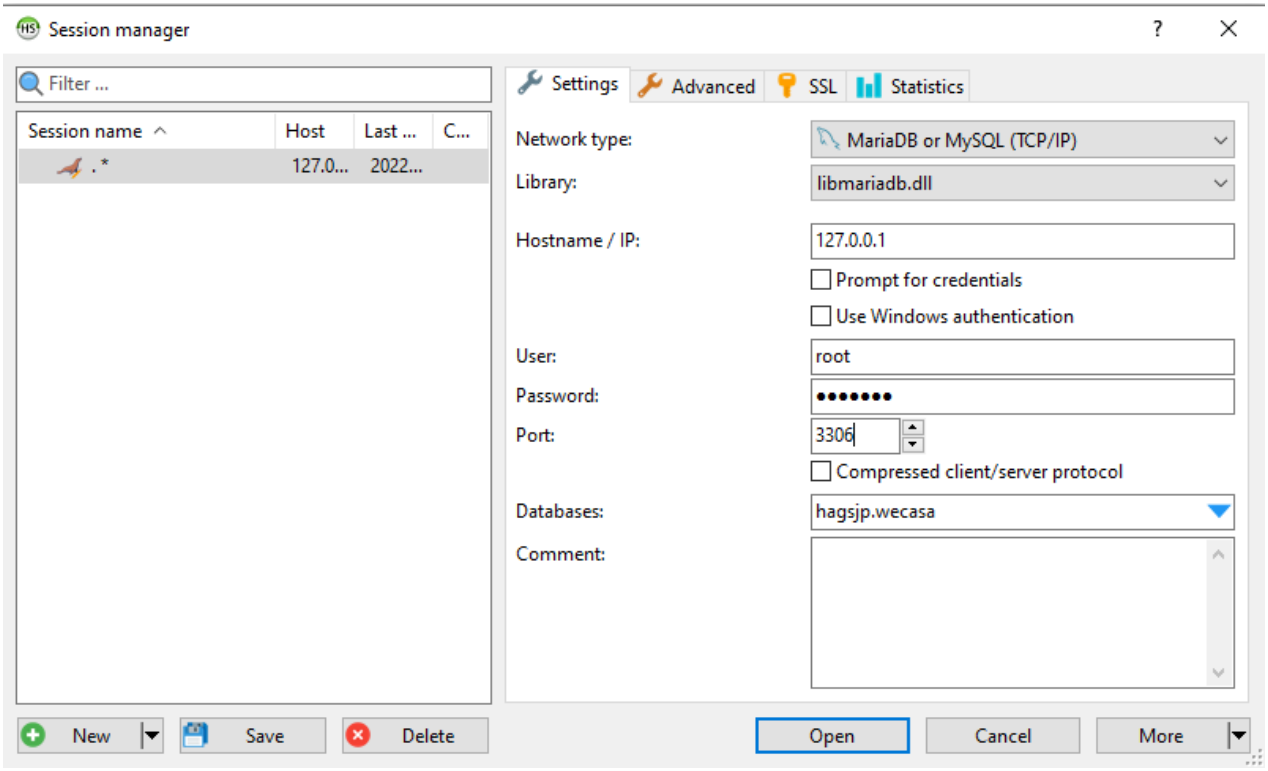
MariaDB (Windows Setup)

Technology used:

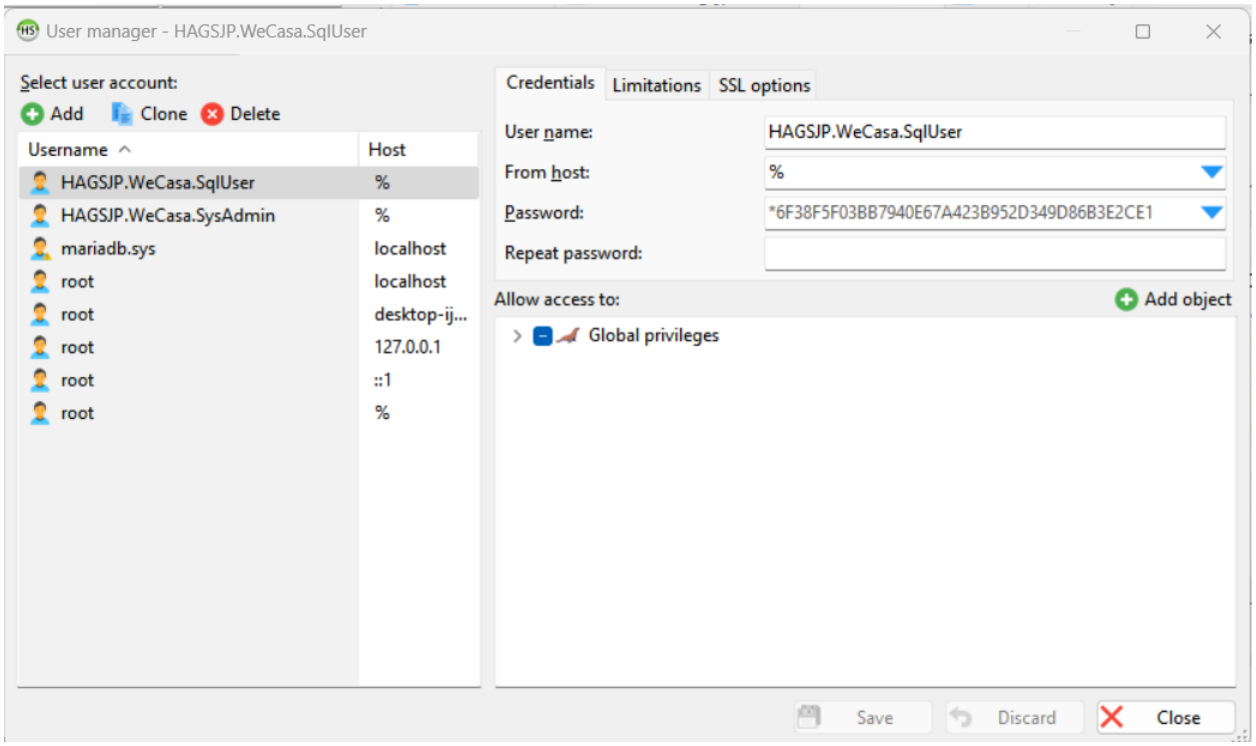
- MariaDB Server free download(comes with HeidiSQL if you're on Windows) (link: https://mariadb.org/download/?t=mariadb&p=mariadb&r=10.11.0&os=windows&cpu=x86_64&pkg=msi&m=gigenet)
    - Use Version 10.6.11

Server Setup

After setting the root user password, use the following configurations for your session. **Make sure the port number is 3306**. If it is not, the code will not work, and the tests will fail.



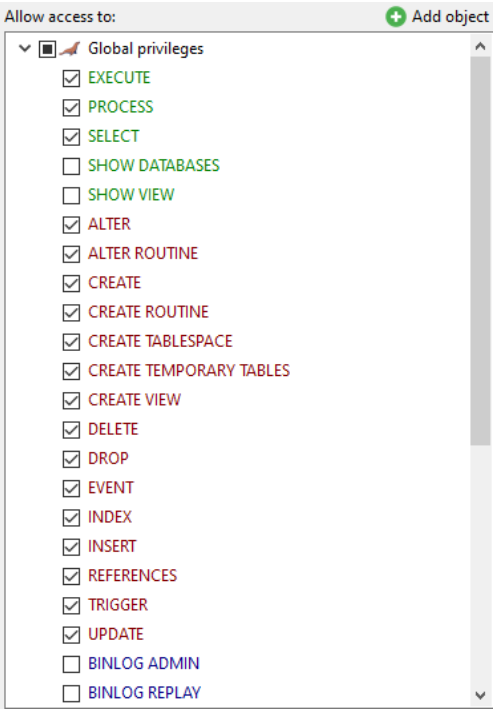
Once you start a session, click on User manager
Create two new users (shown below)
Select dropdown next to "From host: %" and make sure "Access from anywhere is selected"

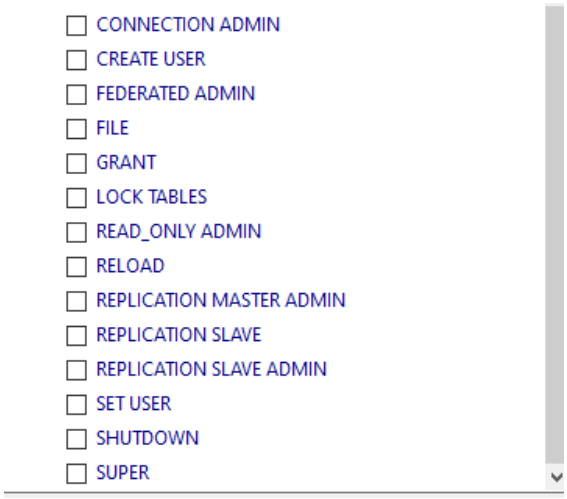


The password for HAGSJP.WeCasa.SqlUser is cecs491

Security privileges

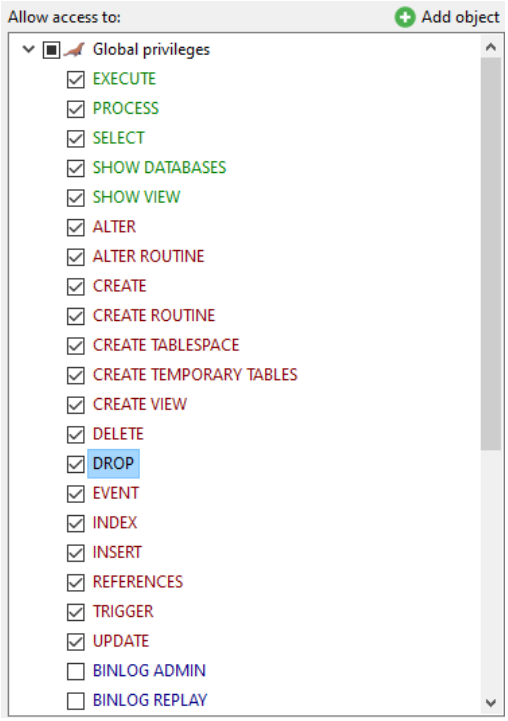
HAGSJP.WeCasa.SqlUser -

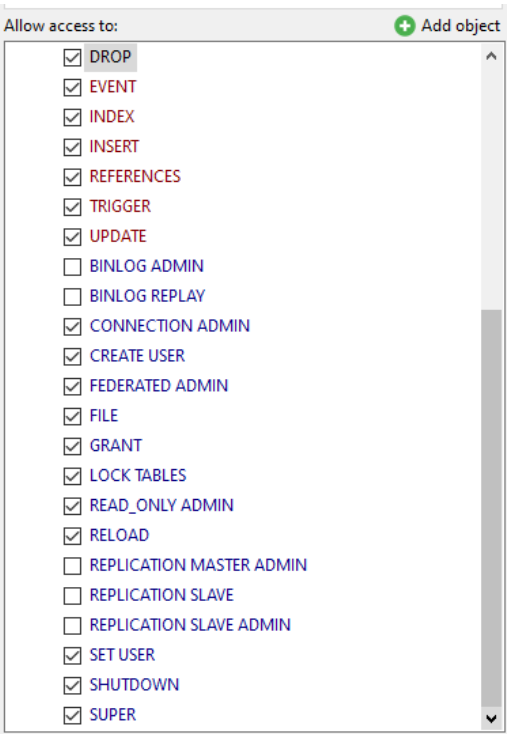




Click Save

HAGSJP.WeCasa.SysAdmin -



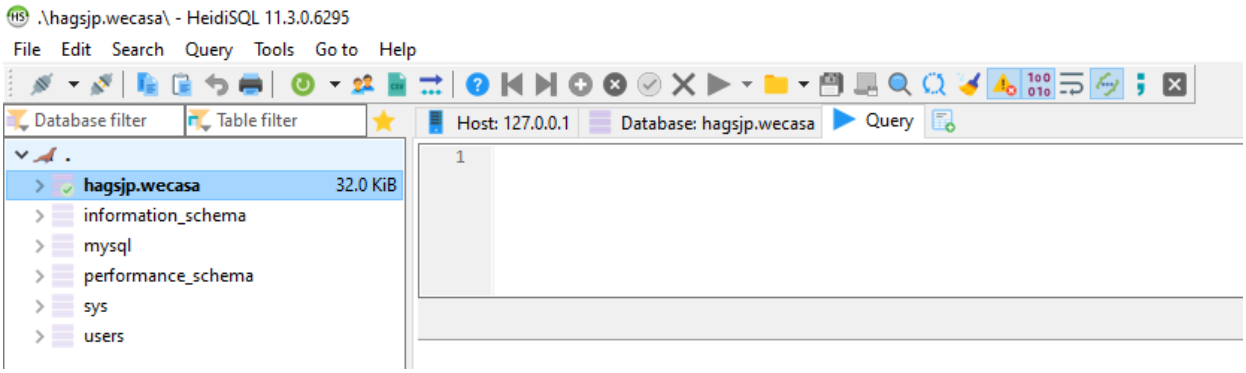


Click Save.

Database

1. Create a new database with the name **HAGSJP.WeCasa**
2. Leave default character set as **latin1_swedish_ci**

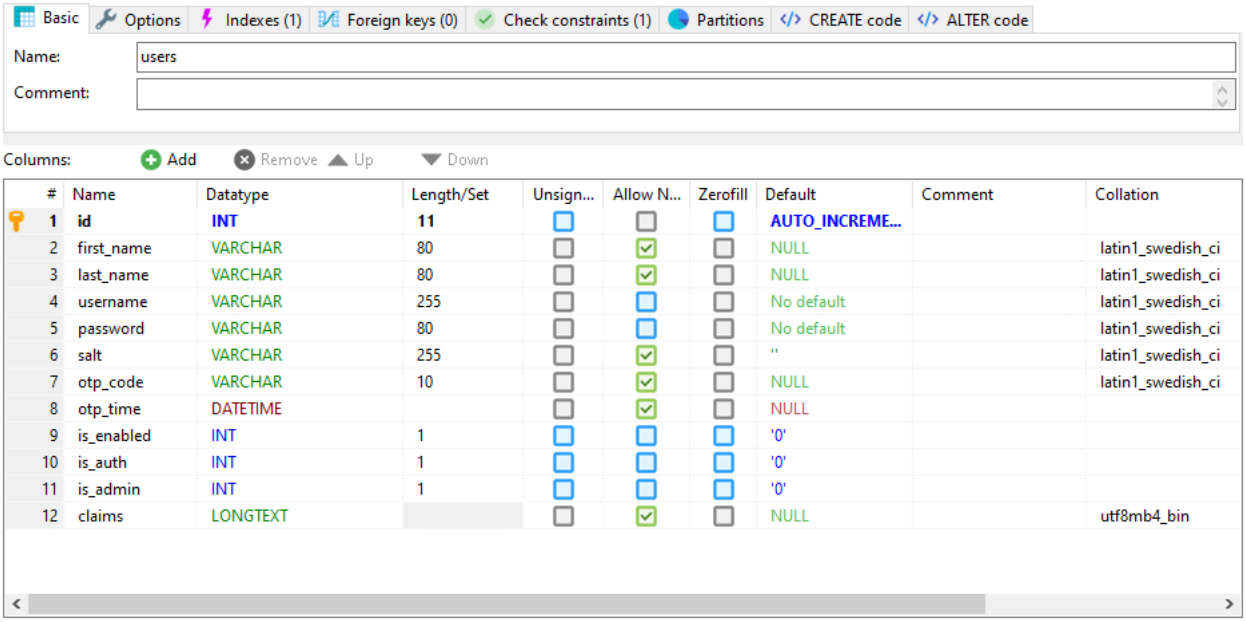
```
CREATE DATABASE `HAGSJP.WeCasa` /*!40100 COLLATE 'latin1_swedish_ci' */
```



3. Add two tables to the new database: **Users** and **Logs**
    - These tables can be created by running the CREATE TABLE code seen below in the Query Tab.
    - Alternatively, these tables can also be created manually by right clicking on the database and selecting New -> Table.

4. Add the following columns to **Users**

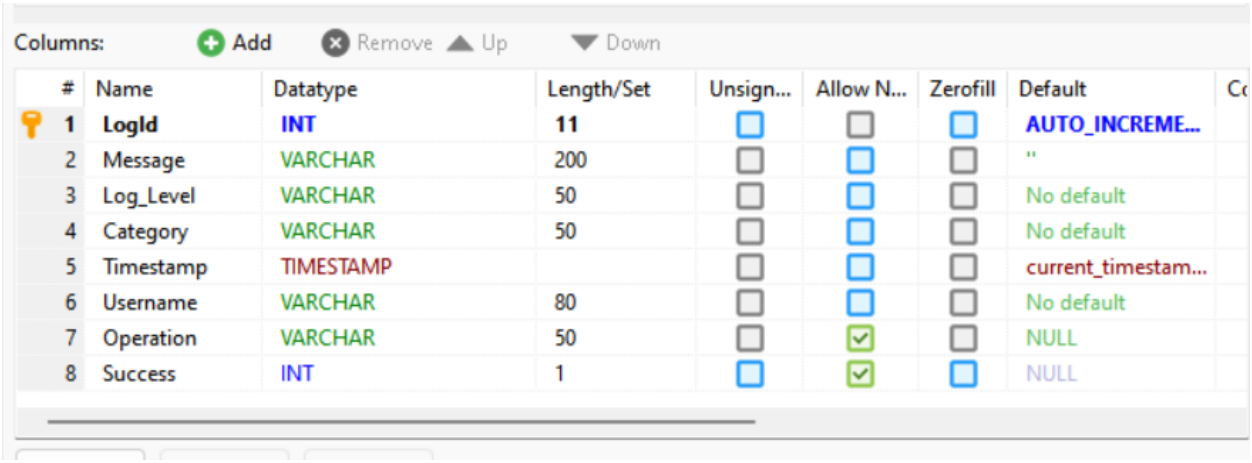
```
CREATE TABLE `users` (
      `id` INT(11) NOT NULL AUTO_INCREMENT,
      `first_name` VARCHAR(80) NULL DEFAULT NULL COLLATE ‘latin1_swedish_ci’,
      `last_name` VARCHAR(80) NULL DEFAULT NULL COLLATE ‘latin1_swedish_ci’,
      `username` VARCHAR(255) NOT NULL COLLATE 'latin1_swedish_ci',
      `password` VARCHAR(80) NOT NULL COLLATE 'latin1_swedish_ci',
      `salt`   VARCHAR(255) NOT NULL DEFAULT “”,
            `otp_code` VARCHAR(10) NULL DEFAULT NULL COLLATE
      'latin1_swedish_ci',
            `otp_time` DATETIME NULL DEFAULT NULL,
            `is_enabled` INT(1) NOT NULL DEFAULT '0',
            `is_auth` INT(1) NOT NULL DEFAULT '0',
            `is_admin` INT(1) NOT NULL DEFAULT '0',
            `claims` LONGTEXT NULL DEFAULT NULL COLLATE 'latin1_swedish_ci',
            PRIMARY KEY (`id`) USING BTREE
      )
      COLLATE='latin1_swedish_ci'
      ENGINE=InnoDB;
```



#	Name	Datatype	Length/Set	Unsign...	Allow N...	Zerofill	Default	Comment	Collation
1	id	INT	11	☐	☐	☐	AUTO_INCREME...		
2	first_name	VARCHAR	80	☐	☑	☐	NULL		latin1_swedish_ci
3	last_name	VARCHAR	80	☐	☑	☐	NULL		latin1_swedish_ci
4	username	VARCHAR	255	☐	☐	☐	No default		latin1_swedish_ci
5	password	VARCHAR	80	☐	☐	☐	No default		latin1_swedish_ci
6	salt	VARCHAR	255	☐	☑	☐	''		latin1_swedish_ci
7	otp_code	VARCHAR	10	☐	☑	☐	NULL		latin1_swedish_ci
8	otp_time	DATETIME		☐	☑	☐	NULL		
9	is_enabled	INT	1	☐	☐	☐	'0'		
10	is_auth	INT	1	☐	☐	☐	'0'		
11	is_admin	INT	1	☐	☐	☐	'0'		
12	claims	LONGTEXT		☐	☑	☐	NULL		utf8mb4_bin

5. Add the following columns to **Logs**

```
CREATE TABLE `logs` (
	`LogId` INT(11) NOT NULL AUTO_INCREMENT,
	`Message` VARCHAR(200) NOT NULL DEFAULT '' COLLATE
'latin1_swedish_ci',
	`Log_Level` VARCHAR(50) NOT NULL COLLATE 'latin1_swedish_ci',
	`Category` VARCHAR(50) NOT NULL COLLATE 'latin1_swedish_ci',
	`Timestamp` TIMESTAMP NOT NULL DEFAULT current_timestamp(),
	`Username` VARCHAR(80) NOT NULL COLLATE 'latin1_swedish_ci',
	`Operation` VARCHAR(50) NULL DEFAULT NULL COLLATE
'latin1_swedish_ci',
	`Success` INT(1) NULL DEFAULT NULL,
	PRIMARY KEY (`LogId`) USING BTREE
)
COLLATE='latin1_swedish_ci'
ENGINE=InnoDB;
```



#	Name	Datatype	Length/Set	Unsign...	Allow N...	Zerofill	Default	Co
1	LogId	INT	11	☐	☐	☐	AUTO_INCREME...	
2	Message	VARCHAR	200	☐	☐	☐	''	
3	Log_Level	VARCHAR	50	☐	☐	☐	No default	
4	Category	VARCHAR	50	☐	☐	☐	No default	
5	Timestamp	TIMESTAMP		☐	☐	☐	current_timestam...	
6	Username	VARCHAR	80	☐	☐	☐	No default	
7	Operation	VARCHAR	50	☐	☑	☐	NULL	
8	Success	INT	1	☐	☑	☐	NULL	

6. Run the following SQL command to create the System Administrator user:

```
INSERT INTO Users (first_name, username, password, is_enabled, is_auth)
VALUES ('WeCasaSysAdmin', 'wecasa@gmail.com', 'Have@GreatSumm3rJapan!', 1, 0);

INSERT INTO Users (username, password, is_enabled, is_auth, is_admin) VALUES
('AuthTestInactiveAcc@gmail.com', 'P@ssw0rd!', 0, 0, 0);

INSERT INTO Users (username, password, is_enabled, is_auth, is_admin)
VALUES ('UpdateClaims@gmail.com', 'P@ssw0rd!', 1, 1, 0);

INSERT INTO Users (username, password, is_enabled, is_auth, is_admin)
```

```
VALUES ('AuthTestSuccess@gmail.com', 'P@ssw0rd!', 1, 1, 0);

INSERT INTO Users (username, password, is_enabled, is_auth, is_admin) VALUES
('UnauthorizedTest@gmail.com', 'P@ssw0rd!', 1, 1, 0);

INSERT INTO Users (username, password, is_enabled, is_auth, is_admin) VALUES
('ExistingEmail@gmail.com', 'P@ssw0rd!', 0, 0, 0);
```