QNAP MariaDB SQL Server Data Base Setup

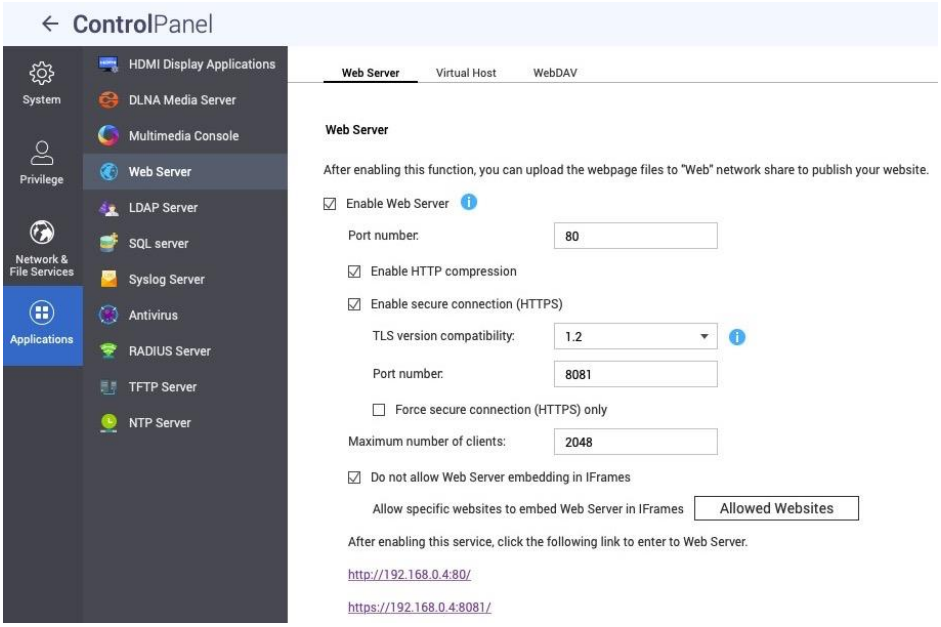
This document is explaining how to create a MariaDB SQL Server Data Base in a QNAP NAS. In this case I was using a QNAP TS-151+ NAS to receive data. The data base was created for a Arduino Weather Station.

Following needs to be done on the QNAP NAS to make the a SQL Server:

1. **Enable Web Server**
2. **Enable MariaDB SQL Server**
3. **Install phpMyAdmin**
4. **Setup a new Database for the Arduino Weather Station Data**
5. **Create and save the qnapdbconnect.php in the phpMyAdmin directory**
6. **Create and save the addweatherdatatoqnapdb.php in the phpMyAdmin directory**
7. **Create and save the reviewweatherdataqnapdb.php in the phpMyAdmin directory**

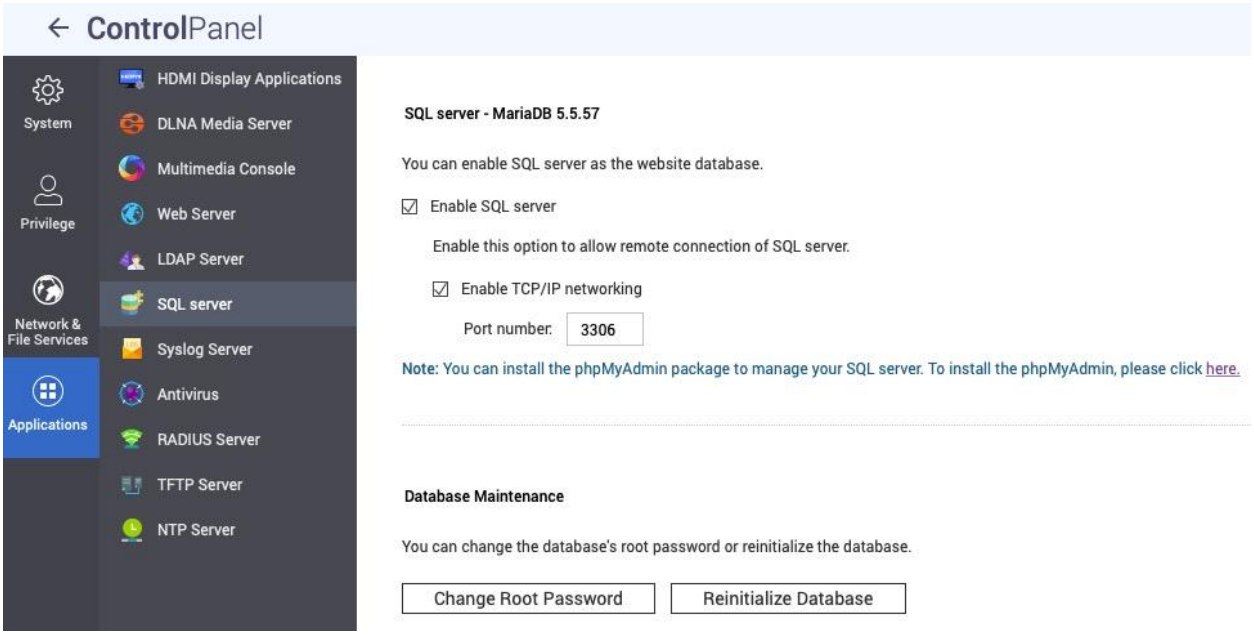
Web Server Setup

The first thing to do is to enable the web server.
You should have the following configuration:



SQL Server Setup

The second thing to do is to enable the SQL server.
You should have the following configuration:

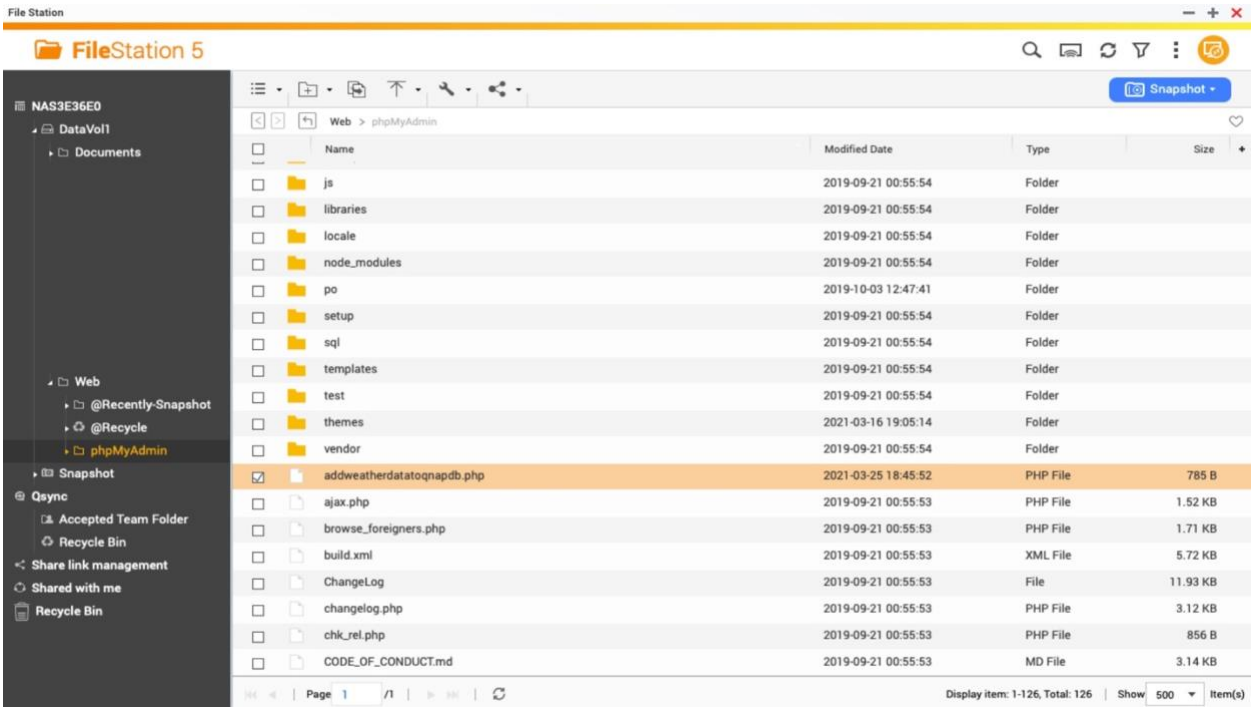


phpMyAdmin installation

In the SQL Server Setup look at the note and click on the link to install the phpMyAdmin app. Follow the instruction and make sure that the app has been installed correctly by login in to the app before continue the next configuration steps. Normally you will have to login as root.

Web Server and phpMyAdmin Directory

When you have added you enabled the SQL Server and the Web Server and installed the phpMyAdmin package you should be able to see following directory structure in your QNAP NAS.



The phpMyAdmin directory is where you are going to put the “qnapdbconnect.php”, “addweatherdatatoqnapdb.php” and the “reviewweatherdataqnapdb.php” that will be explained in the next steps and should be under <NAS>/<DataVolume>/Web/.

Data Base Setup with SQL

You should use following SQL commands to create the table to avoid any mistyping:

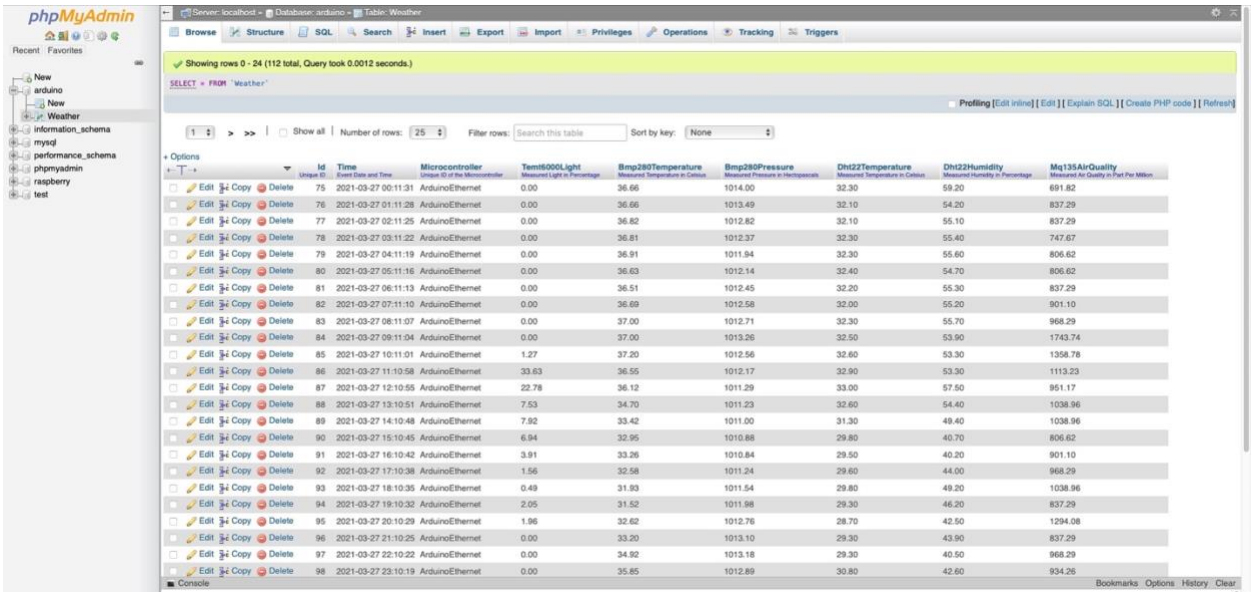
weatherarduino.sql

```
CREATE TABLE `arduino`.`Weather` (

`Id` INT NOT NULL AUTO_INCREMENT PRIMARY KEY COMMENT 'Unique ID',
`Time` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP COMMENT 'Event Date and Time',
`Microcontroller` VARCHAR( 30 ) NOT NULL COMMENT 'Unique ID of the Microcontroller',
`Temt6000Light` VARCHAR( 10 ) NOT NULL COMMENT 'Measured Light in Percentage',
```

`Bmp280Temperature` VARCHAR(10) NOT NULL COMMENT 'Measured Temperature in Celsius',
`Bmp280Pressure` VARCHAR(10) NOT NULL COMMENT 'Measured Pressure in Hectopascals',
`Dht22Temperature` VARCHAR(10) NOT NULL COMMENT 'Measured Temperature in Celsius',
`Dht22Humidity` VARCHAR(10) NOT NULL COMMENT 'Measured Humidity in Percentage',
`Mq135AirQuality` VARCHAR(10) NOT NULL COMMENT 'Measured Air Quality in Part Per Million',
INDEX (`Time` , `Microcontroller`)
) ENGINE = InnoDB;

The Database should look like this (Will be empty when created):



PHP: MySQL Connection

This file has only one purpose: setup a connection from PHP to the MariaDB QL which will be used for both submitting and retrieving data.
Save it as “qnapdbconnect.php” in your Web/phpMyAdmin directory.

qnapdbconnect.php:

```
<?php
$servername = "localhost";
$username = "root";
$password = "<PASSWORD>”;                    Password for the MySQL Arduino Database!!”
$dbname = "arduino";

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
  die("CPanel Connection Failed: " . $conn->connect_error);
}
else {
        echo "cPanel Database Connected: ";
}
```

PHP: Upload Data

In this file we will use the mechanism of the HTML <FORM> with GET.
The GET values will be inserted into the database and the “reviewweatherdataqnapdb.php” will be loaded right after that, which is optional, and not needed for the Arduino, but practical when you try to submit data from your browser for testing purposes. During normal operation of your “end product” you *could* remove line 12.
Save this file as “addweatherdatatoqnapdb.php” in your Web/phpMyAdmin directory.

addweatherdatatoqnapdb.php:

```
<?php
   // Connect to MySQL
   include("qnapdbconnect.php");
   // Prepare the SQL statement
   $sql = "INSERT INTO arduino.Weather (Microcontroller ,Temt6000Light,
Bmp280Temperature, Bmp280Pressure, Dht22Temperature, Dht22Humidity, Mq135AirQuality)
   VALUES ('".$_GET["Microcontroller"]."', '".$_GET["Temt6000Light"]."',
'".$_GET["Bmp280Temperature"]."', '".$_GET["Bmp280Pressure"]."',
'".$_GET["Dht22Temperature"]."', '".$_GET["Dht22Humidity"]."',
'".$_GET["Mq135AirQuality"]."')";
          if ($conn->query($sql) === TRUE) {
            echo "New SQL record created";
          } else {
          echo "Error: " . $sql . "<br>" . $conn->error;
          }
   // Go to the review_data.php (optional)
   header("Location: https://192.168.0.4:8081/phpMyAdmin/reviewweatherdataqnapdb.php");
          $conn->close();
?>
```

PHP: Show current table content

I'll keep this file very simple, it's just to illustrate that it works, although I added some CSS to make a simple table still look nice.
Save this file as "reviewweatherdataqnapdb.php" in in your Web/phpMyAdmin directory, and if you'd like you can already test it by entering the address of your web-server followed by "eviewweatherdataqnapdb.php" (the table will be empty of course).

For example: https://192.168.0.4:8081/phpMyAdmin/reviewweatherdataqnapdb.php.
(replace "192.168.0.4:8081" with the name or IP address of our server, or with "localhost")

The log should look like this:

Arduino Temperature Log

ID	Date and Time	Microcontroller	TEMT6000 Light	BMP280 Temperature	BMP280 Air Pressure	DHT22 Temperature	DDHT22 Humidity	MQ135 Air Quality
75	2021-03-27 00:11:31	ArduinoEthernet	0.00	36.66	1014.00	32.30	59.20	691.82
76	2021-03-27 01:11:28	ArduinoEthernet	0.00	36.66	1013.49	32.10	54.20	837.29
77	2021-03-27 02:11:25	ArduinoEthernet	0.00	36.82	1012.82	32.10	55.10	837.29
78	2021-03-27 03:11:22	ArduinoEthernet	0.00	36.81	1012.37	32.30	55.40	747.67
79	2021-03-27 04:11:19	ArduinoEthernet	0.00	36.91	1011.94	32.30	55.60	806.62
80	2021-03-27 05:11:16	ArduinoEthernet	0.00	36.63	1012.14	32.40	54.70	806.62
81	2021-03-27 06:11:13	ArduinoEthernet	0.00	36.51	1012.45	32.20	55.30	837.29
82	2021-03-27 07:11:10	ArduinoEthernet	0.00	36.69	1012.58	32.00	55.20	901.10
83	2021-03-27 08:11:07	ArduinoEthernet	0.00	37.00	1012.71	32.30	55.70	968.29
84	2021-03-27 09:11:04	ArduinoEthernet	0.00	37.00	1013.26	32.50	53.90	1743.74
85	2021-03-27 10:11:01	ArduinoEthernet	1.27	37.20	1012.56	32.60	53.30	1358.78
86	2021-03-27 11:10:58	ArduinoEthernet	33.63	36.55	1012.17	32.90	53.30	1113.23
87	2021-03-27 12:10:55	ArduinoEthernet	22.78	36.12	1011.29	33.00	57.50	951.17
88	2021-03-27 13:10:51	ArduinoEthernet	7.53	34.70	1011.23	32.60	54.40	1038.96
89	2021-03-27 14:10:48	ArduinoEthernet	7.92	33.42	1011.00	31.30	49.40	1038.96
90	2021-03-27 15:10:45	ArduinoEthernet	6.94	32.95	1010.88	29.80	40.70	806.62
91	2021-03-27 16:10:42	ArduinoEthernet	3.91	33.26	1010.84	29.50	40.20	901.10
92	2021-03-27 17:10:38	ArduinoEthernet	1.56	32.58	1011.24	29.60	44.00	968.29
93	2021-03-27 18:10:35	ArduinoEthernet	0.49	31.93	1011.54	29.80	49.20	1038.96
94	2021-03-27 19:10:32	ArduinoEthernet	2.05	31.52	1011.98	29.30	46.20	837.29
95	2021-03-27 20:10:29	ArduinoEthernet	1.96	32.62	1012.76	28.70	42.50	1294.08
96	2021-03-27 21:10:25	ArduinoEthernet	0.00	33.20	1013.10	29.30	43.90	837.29
97	2021-03-27 22:10:22	ArduinoEthernet	0.00	34.92	1013.18	29.30	40.50	968.29
98	2021-03-27 23:10:19	ArduinoEthernet	0.00	35.85	1012.89	30.80	42.60	934.26
99	2021-03-28 00:10:16	ArduinoEthernet	0.00	35.72	1012.61	31.80	46.10	1380.85
100	2021-03-28 01:10:12	ArduinoEthernet	0.00	35.85	1012.24	31.60	48.10	1151.75
101	2021-03-28 02:10:09	ArduinoEthernet	0.00	35.88	1011.85	31.80	49.20	1231.63

reviewweatherdataqnapdb.php:

```
<?php
   // Start MySQL Connection
   include('qnapdbconnect.php');
?>

<html>
<head>
   <title>Arduino Temperature Log</title>
   <style type="text/css">
```

```
      .table_titles, .table_cells_odd, .table_cells_even {
            padding-right: 20px;
            padding-left: 20px;
            color: #000;
      }
      .table_titles {
         color: #FFF;
         background-color: #666;
      }
      .table_cells_odd {
         background-color: #CCC;
      }
      .table_cells_even {
         background-color: #FAFAFA;
      }
      table {
         border: 2px solid #333;
      }
      body { font-family: "Trebuchet MS", Arial; }
   </style>
</head>

   <body>
      <h1>Arduino Temperature Log</h1>
   <table border="0" cellspacing="0" cellpadding="4">
      <tr>
         <td class="table_titles">ID</td>
         <td class="table_titles">Date and Time</td>
         <td class="table_titles">Microcontroller</td>
         <td class="table_titles">TEMT6000 Light</td>
         <td class="table_titles">BMP280 Temperature</td>
         <td class="table_titles">BMP280 Air Pressure</td>
         <td class="table_titles">DHT22 Temperature</td>
         <td class="table_titles">DDHT22 Humidity</td>
         <td class="table_titles">MQ135 Air Quality</td>
      </tr>
<?php
   // Retrieve all records and display them

   $sql = "SELECT * FROM arduino.Weather";
   $result = $conn->query($sql);

   $oddrow = true;
```

```
  if ($result->num_rows > 0) {
    // output data of each row
    while($row = $result->fetch_assoc()) {

      if ($oddrow)
      {
        $css_class=' class="table_cells_odd"';
      }
      else
      {
        $css_class=' class="table_cells_even"';
      }

      $oddrow = !$oddrow;
      echo '<tr>';
      echo '   <td'.$css_class.'>'.$row["Id"].'</td>';
      echo '   <td'.$css_class.'>'.$row["Time"].'</td>';
      echo '   <td'.$css_class.'>'.$row["Microcontroller"].'</td>';
      echo '   <td'.$css_class.'>'.$row["Temt6000Light"].'</td>';
      echo '   <td'.$css_class.'>'.$row["Bmp280Temperature"].'</td>';
      echo '   <td'.$css_class.'>'.$row["Bmp280Pressure"].'</td>';
      echo '   <td'.$css_class.'>'.$row["Dht22Temperature"].'</td>';
      echo '   <td'.$css_class.'>'.$row["Dht22Humidity"].'</td>';
      echo '   <td'.$css_class.'>'.$row["Mq135AirQuality"].'</td>';
      echo '</tr>';

    }
  } else {
    echo "0 results";
  }
  $conn->close();
?>
  </table>
  </body>
</html>
```

Final Words

If you have followed all the steps, you should now have the MariaDB SQL Server Data Base prepared to receive data from the Arduino Weather Station.