DataBase Assignment

EE 470 - Introduction to the Internet of Things- Fall 2024 / Version: V2

NAME Jair Pacheco DATE: 10/3/2025

This project must be done individually!

Link to this document:
https://docs.google.com/document/d/1zhLaeiWjtqfpCTeJzd_n543yl6_SxahBxWd4lHh4srY/edit?usp=sharing

Please make sure you submit your assignment individually!

DataBase Assignment

EE 470 - Introduction to the Internet of Things- Fall 2024 / Version: V2

1. Assignment Overview

The purpose of this exercise is to become familiar with MySQL database and its commands.

2. Learning Objectives

- Set Up an online web page
- Learn about SQL
- Practice SQL commands
- Create a Table for sensor data
- Using Github.

3. Required Materials & Reference Information

- Signup for https://hostinger.com
- database tutorial available on the course GitHub
- Tutorial on creating an account with hostinger.com
- You can find the longitude and latitude format here
- Good tutorial: https://www.tutorialspoint.com/mariadb/mariadb_introduction.htm

4. Experiment

This experiment has TWO parts.

Part 1:
In this part of the experiment you need to signup for an account at https://hostinger.com
This will allow you to have your personal web page and database.

Part 2:
In the second part of the experiment you will need to access the database using a PHP file and visualize the data using the Chartjs library.

DataBase Assignment

EE 470 - Introduction to the Internet of Things- Fall 2024 / Version: V2

PART 1

1- A Short Introduction to Databases

A database is a systematic or organized collection of related information that is stored in such a way that it can be easily accessed, retrieved, managed, and updated. It is where all data is stored, very much like a library that houses a wide range of books from different genres. Think of data as books.

In a database, you can organize the data in rows and columns in the form of a table. Indexing the data makes it easy to find and retrieve it again as and when required. Many websites on the World Wide Web are managed with the help of databases. To create a database so that the data is accessible to users through only one set of software programs, database handlers are used.

MySQL, SQL Server, MongoDB, Oracle Database, PostgreSQL, Informix, Sybase, etc. are all examples of different databases. These modern databases are managed by DBMS. Structured Query Language, or SQL as it is more widely known, is used to operate on the data in a database.

MariaDB is a popular fork of MySQL created by MySQL's original developers. It grew out of concerns related to MySQL's acquisition by Oracle. It offers support for both small data processing tasks and enterprise needs. It aims to be a drop-in replacement for MySQL requiring only a simple uninstall of MySQL and an install of MariaDB. MariaDB offers the same features of MySQL and much more.

For more information about Databases read this: https://intellipaat.com/blog/what-is-database/

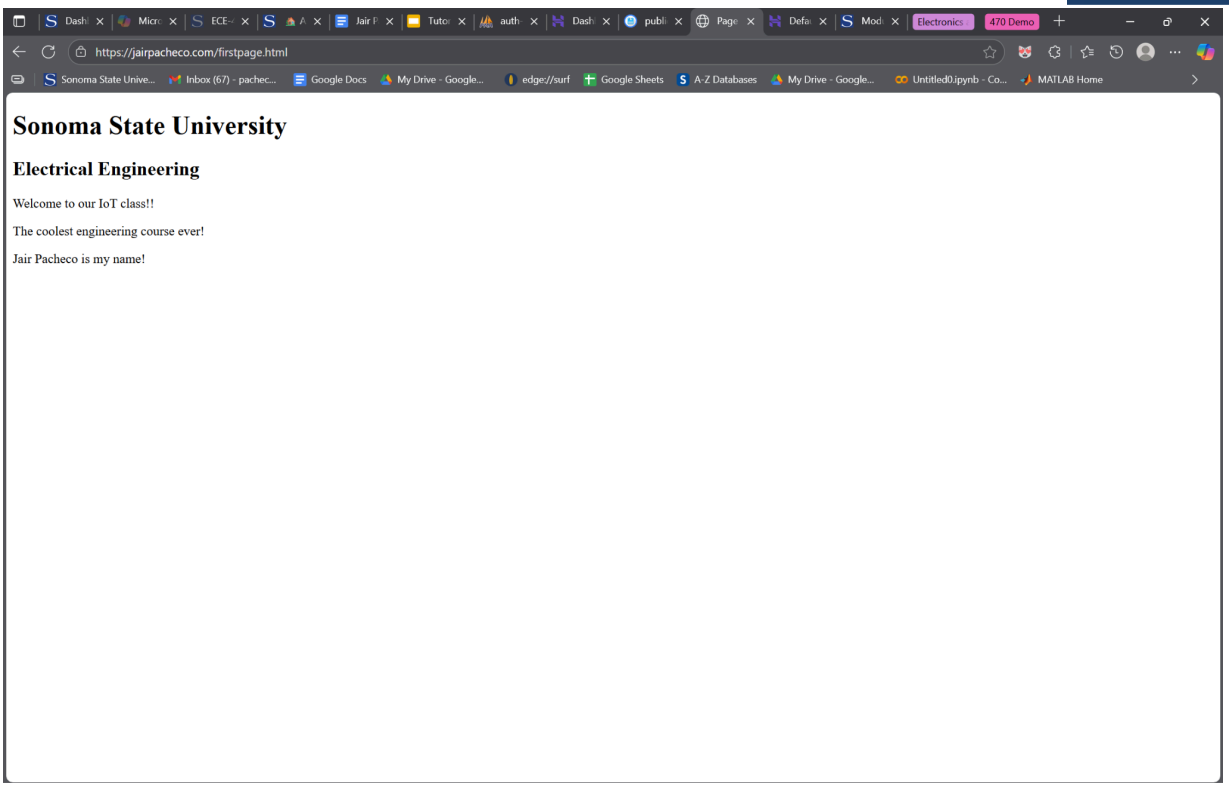
1.1. Create an Account

A- Carefully review the Tutorial on creating an account with hostinger.com. Create a very simple webpage, as described in the tutorial. Make sure your web page is working. Add your name to your webpage. We will call this web page Internet of Things (IoT). We will use this web page to place links to all your future assignments.

https://jairpacheco.com/firstpage.html
Link to your webpage

DataBase Assignment

EE 470 - Introduction to the Internet of Things- Fall 2024 / Version: V2



Snapshot of your webpage

B- Create a dBase. Make sure your database's name is your `YourFirstNameLastName`. The username for the database must be `db_YourFirstNameLastName`. For example database: `FaridFarahmand`
User: `db_FaridFarahmand`

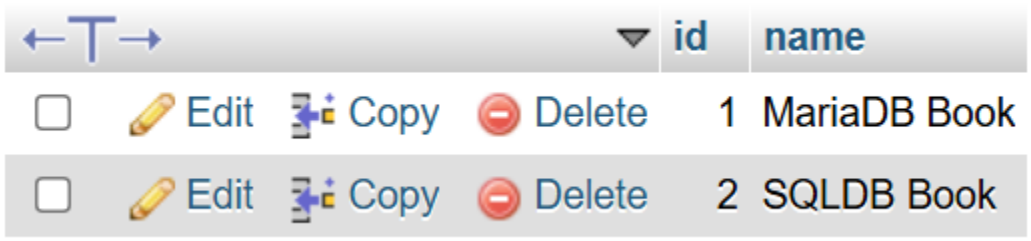
You should see something like this:

Database
information_schema
u301268454_AnthonyMeyer

Follow the Tutorial and create a simple `BOOK` table. Using `SHOW DATABASES` you can show your dBase. For more information Please refer to Github Tutorial. Your final `BOOK TABLE` should look something like this:

id	name
1	MariaDB Book
2	SQLDB Book

Take a snapshot of your results:



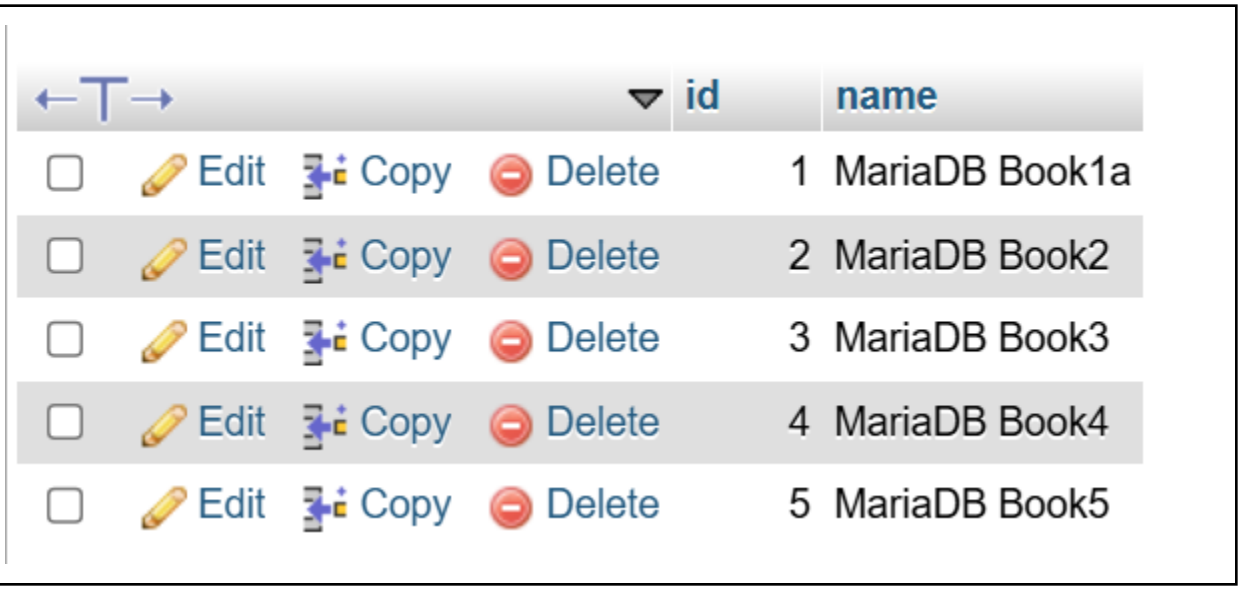
	id	name
Edit Copy Delete	1	MariaDB Book
Edit Copy Delete	2	SQLDB Book

Snapshot of your BOOK table.

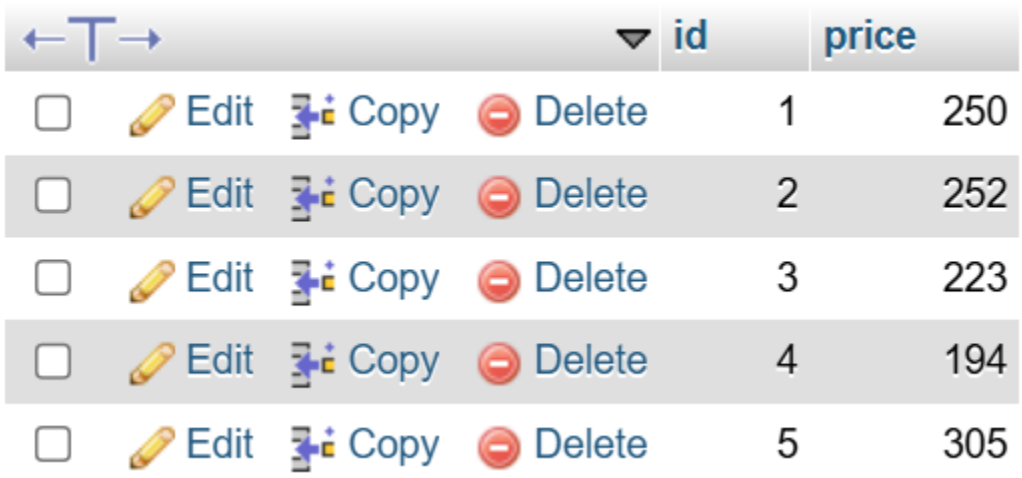
1.2. Practice with MySQL Commands

Using the phpMyAdmin in hostinger.com let's practice more with MySQL commands.

Review the commands in the database tutorial available on the course GitHub. Make sure you complete the tutorial. Once you complete the tutorial you should have a database table with your name and TWO tables having: `Price and Book`. Each table must have the appropriate elements.



	id	name
Edit Copy Delete	1	MariaDB Book1a
Edit Copy Delete	2	MariaDB Book2
Edit Copy Delete	3	MariaDB Book3
Edit Copy Delete	4	MariaDB Book4
Edit Copy Delete	5	MariaDB Book5



		id	price
☐	Edit Copy Delete	1	250
☐	Edit Copy Delete	2	252
☐	Edit Copy Delete	3	223
☐	Edit Copy Delete	4	194
☐	Edit Copy Delete	5	305

Snapshot of each table separately: `Price and Book`.

Now let's combine different fields and create ONE single table showing both book name and price - similar to below:

name	price
MariaDB Book1a	250
MariaDB Book2	252
MariaDB Book3	223
MariaDB Book4	194
MariaDB Book5	305

Take a snapshot of your results:

name	price
MariaDB Book1a	250
MariaDB Book2	252
MariaDB Book3	223
MariaDB Book4	194
MariaDB Book5	305

Snapshot: create ONE single table showing both book name and price

1.3. Create Your Own Sensory-Table

Now you need to design your own database using the following **design requirements**:
Design Two tables: `sensor_data & sensor_resister.`

`sensor_data` must have the following fields:

- node_name <up to 10 characters>; this will be used as the KEY
- time_received <indicating time/date>
- temperature <a numeric value between -10 to +100 celsius with maximum of 6 positions>
- humidity <a number value between 0-100 with maximum of 6 positions>

`sensor_register` must have the following fields:

- node_name <up to 10 characters>; this will be used as the KEY
- manufacturer <up to 10 characters>
- longitude < <a real (positive/negative number with maximum of 15 positions>
- Latitude <a real (positive/negative number with maximum of 15 positions>

You can find the longitude and latitude format here.

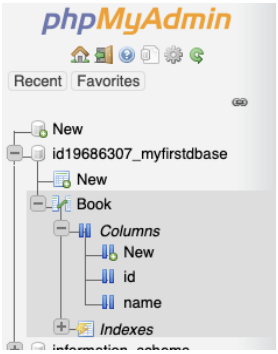
You can use a command similar to the following to create sensor data:

```
INSERT INTO sensor (node_name, time_received, temperature, humidity)
VALUES ('node-1','10-05-2022 10:05:34',84,25);
```

Your database must have the following requirements:

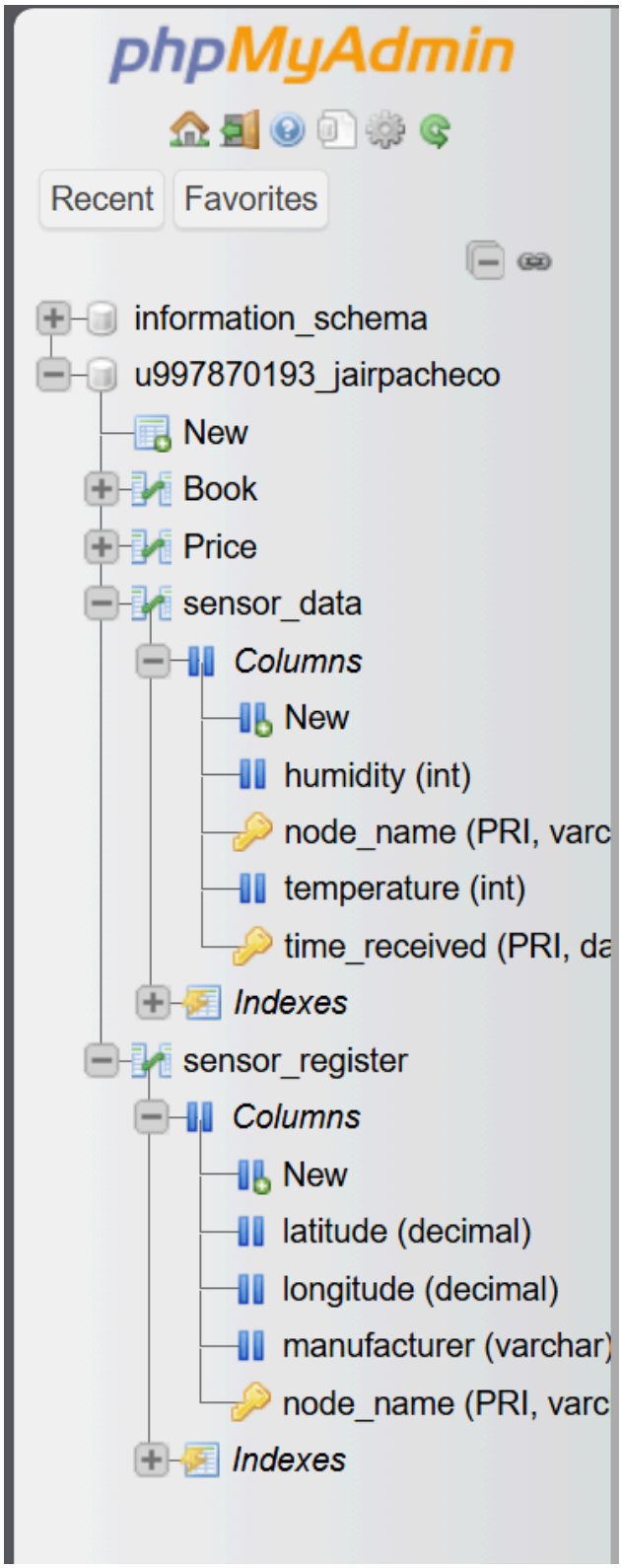
1- The starting data must be October 1, 2022, at 11:00 am.
2- The data must be sent every half an hour.
3- We need to register 5 different sensors with five different manufacturer names.
5- Each sensor must transmit at least 4 different data sets. This means there must be at least 4 different entries from each sensor node.
6- Node name format must be the following: node_1, node_2, node_3…. Etc.
7- No data from the sensor_data table is accepted if the transmitting node is not registered in the sensor_register table.
8- Create a VIEW of both tables and call it sensor_combined.

Once you have completed the database construction and the VIEW, take a snapshot of the content of the VIEW (sensor_combined) and all the commands you used to construct the databases. In addition, take a snapshot of the database structure, as it appears on PhpMyAdmin portal, as shown below. Your snapshots should be similar to the figures below.



Database as shown on the PhpMyAdmin page. something like the example above.

DataBase Assignment



```sql
-- Create a table
CREATE TABLE Book(
id INT NOT NULL AUTO_INCREMENT,
name VARCHAR(100) NOT NULL,
PRIMARY KEY (id));

CREATE TABLE Price(
id INT NOT NULL AUTO_INCREMENT,
price float NOT NULL,
PRIMARY KEY (id)); -- primary key is the id
```

Your complete sql script to generate the tables. Something like the example above.

```sql
/*
-- step 1:
CREATE TABLE sensor_data (
  node_name VARCHAR(10) NOT NULL,
  time_received DATETIME NOT NULL,
  temperature INT(6) NOT NULL CHECK (temperature BETWEEN -10 AND 100),
  humidity INT(6) NOT NULL CHECK (humidity BETWEEN 0 AND 100),
  PRIMARY KEY (node_name, time_received),
  FOREIGN KEY (node_name) REFERENCES sensor_register(node_name)
);
*/

/*
-- step 2
CREATE TABLE sensor_register (
  node_name VARCHAR(10) PRIMARY KEY NOT NULL,
  manufacturer VARCHAR(10) NOT NULL,
  longitude DECIMAL(15,6) NOT NULL,
  latitude DECIMAL(15,6) NOT NULL
);
*/
-- step 3
/*
INSERT INTO sensor_register (node_name, manufacturer, longitude, latitude) VALUES
('node-1','sysben',-100.000052, 25.256789);
INSERT INTO sensor_register (node_name, manufacturer, longitude, latitude) VALUES
('node-2','tomtech',-170.010927, 73.251709);
INSERT INTO sensor_register (node_name, manufacturer, longitude, latitude) VALUES
('node-3','angelsight',-150.921952, 85.457980);
```

```
INSERT INTO sensor_register (node_name, manufacturer, longitude, latitude) VALUES
('node-4','Hanklock',-160.743150, 100.916830);
INSERT INTO sensor_register (node_name, manufacturer, longitude, latitude) VALUES
('node-5','Datalink',-180.383657, 120.256789);
*/

-- step 4
/*

-- for sensor 1
INSERT INTO sensor_data (node_name, time_received, temperature, humidity) VALUES
('node-1','2022-10-01 11:00:00',84,25);
INSERT INTO sensor_data (node_name, time_received, temperature, humidity) VALUES
('node-1','2022-10-01 11:30:00',88,24);
INSERT INTO sensor_data (node_name, time_received, temperature, humidity) VALUES
('node-1','2022-10-01 12:00:00',94,23);
INSERT INTO sensor_data (node_name, time_received, temperature, humidity) VALUES
('node-1','2022-10-01 12:30:00',90,20);

-- for sensor 2
INSERT INTO sensor_data (node_name, time_received, temperature, humidity) VALUES
('node-2','2022-10-01 11:00:00',30,25);
INSERT INTO sensor_data (node_name, time_received, temperature, humidity) VALUES
('node-2','2022-10-01 11:30:00',38,30);
INSERT INTO sensor_data (node_name, time_received, temperature, humidity) VALUES
('node-2','2022-10-01 12:00:00',44,28);
INSERT INTO sensor_data (node_name, time_received, temperature, humidity) VALUES
('node-2','2022-10-01 12:30:00',49,30);

-- for sensor 3
INSERT INTO sensor_data (node_name, time_received, temperature, humidity) VALUES
('node-3','2022-10-01 11:00:00',69,20);
INSERT INTO sensor_data (node_name, time_received, temperature, humidity) VALUES
('node-3','2022-10-01 11:30:00',72,23);
INSERT INTO sensor_data (node_name, time_received, temperature, humidity) VALUES
('node-3','2022-10-01 12:00:00',76,22);
INSERT INTO sensor_data (node_name, time_received, temperature, humidity) VALUES
('node-3','2022-10-01 12:30:00',74,21);

-- for sensor 4
INSERT INTO sensor_data (node_name, time_received, temperature, humidity) VALUES
('node-4','2022-10-01 11:00:00',0,10);
INSERT INTO sensor_data (node_name, time_received, temperature, humidity) VALUES
('node-4','2022-10-01 11:30:00',-2,7);
```

```
INSERT INTO sensor_data (node_name, time_received, temperature, humidity) VALUES
('node-4','2022-10-01 12:00:00',3,9);
INSERT INTO sensor_data (node_name, time_received, temperature, humidity) VALUES
('node-4','2022-10-01 12:30:00',8,11);

-- for sensor 5
INSERT INTO sensor_data (node_name, time_received, temperature, humidity) VALUES
('node-5','2022-10-01 11:00:00',75,30);
INSERT INTO sensor_data (node_name, time_received, temperature, humidity) VALUES
('node-5','2022-10-01 11:30:00',80,28);
INSERT INTO sensor_data (node_name, time_received, temperature, humidity) VALUES
('node-5','2022-10-01 12:00:00',84,24);
INSERT INTO sensor_data (node_name, time_received, temperature, humidity) VALUES
('node-5','2022-10-01 12:30:00',89,27);


 SELECT * FROM sensor_data;
*/

-- step 5
CREATE VIEW combined_sensor AS SELECT
    sensor_data.node_name,
    sensor_register.manufacturer,
    sensor_register.longitude,
    sensor_register.latitude,
    sensor_data.time_received,
    sensor_data.temperature,
    sensor_data.humidity

FROM
    sensor_data,
    sensor_register
WHERE
    sensor_data.node_name = sensor_register.node_name;

SELECT * FROM combined_sensor;
```

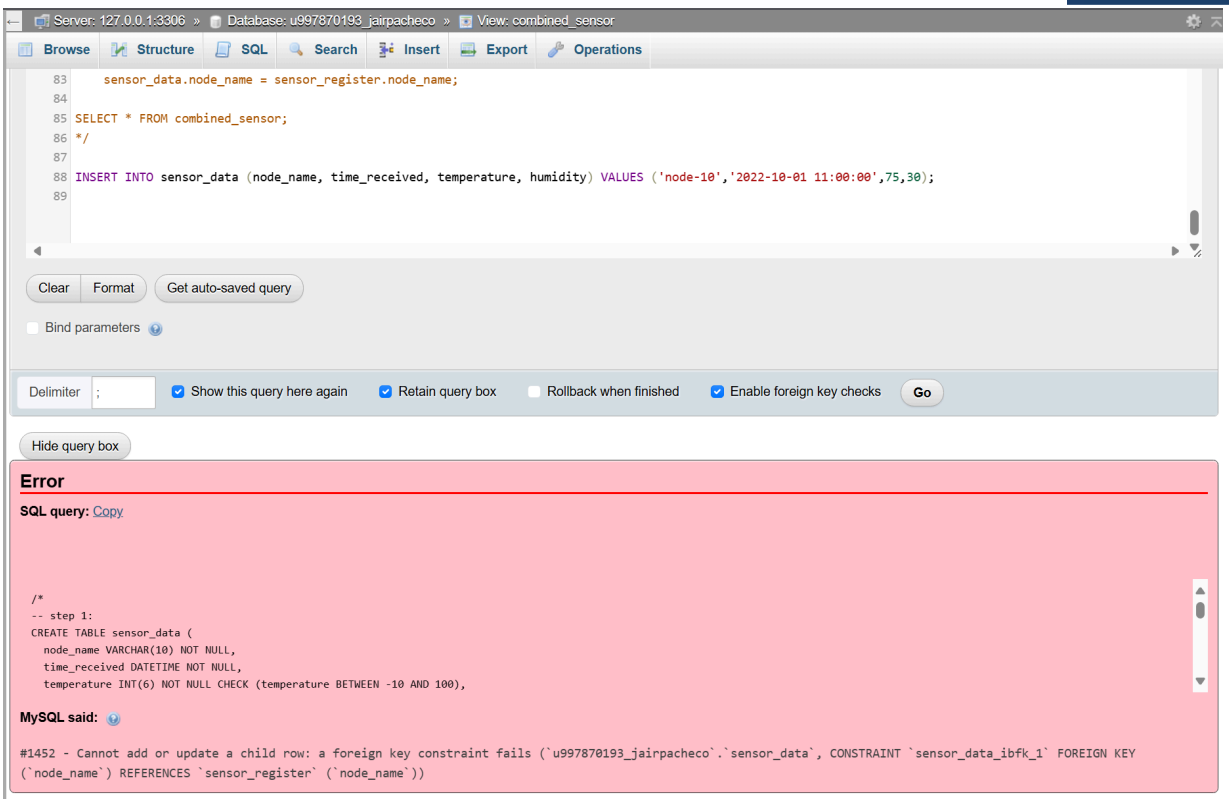
DataBase Assignment

	node_name	manufacturer	longitude	latitude	time_received	temperature	humidity
Edit Copy Delete	node-1	sysben	-100.000052	25.256789	2022-10-01 11:00:00	84	25
Edit Copy Delete	node-1	sysben	-100.000052	25.256789	2022-10-01 11:30:00	88	24
Edit Copy Delete	node-1	sysben	-100.000052	25.256789	2022-10-01 12:00:00	94	23
Edit Copy Delete	node-1	sysben	-100.000052	25.256789	2022-10-01 12:30:00	90	20
Edit Copy Delete	node-2	tomtech	-170.010927	73.251709	2022-10-01 11:00:00	30	25
Edit Copy Delete	node-2	tomtech	-170.010927	73.251709	2022-10-01 11:30:00	38	30
Edit Copy Delete	node-2	tomtech	-170.010927	73.251709	2022-10-01 12:00:00	44	28
Edit Copy Delete	node-2	tomtech	-170.010927	73.251709	2022-10-01 12:30:00	49	30
Edit Copy Delete	node-3	angelsight	-150.921952	85.457980	2022-10-01 11:00:00	69	20
Edit Copy Delete	node-3	angelsight	-150.921952	85.457980	2022-10-01 11:30:00	72	23
Edit Copy Delete	node-3	angelsight	-150.921952	85.457980	2022-10-01 12:00:00	76	22
Edit Copy Delete	node-3	angelsight	-150.921952	85.457980	2022-10-01 12:30:00	74	21
Edit Copy Delete	node-4	Hanklock	-160.743150	100.916830	2022-10-01 11:00:00	0	10
Edit Copy Delete	node-4	Hanklock	-160.743150	100.916830	2022-10-01 11:30:00	-2	7
Edit Copy Delete	node-4	Hanklock	-160.743150	100.916830	2022-10-01 12:00:00	3	9
Edit Copy Delete	node-4	Hanklock	-160.743150	100.916830	2022-10-01 12:30:00	8	11
Edit Copy Delete	node-5	Datalink	-180.383657	120.256789	2022-10-01 11:00:00	75	30
Edit Copy Delete	node-5	Datalink	-180.383657	120.256789	2022-10-01 11:30:00	80	28
Edit Copy Delete	node-5	Datalink	-180.383657	120.256789	2022-10-01 12:00:00	84	24
Edit Copy Delete	node-5	Datalink	-180.383657	120.256789	2022-10-01 12:30:00	89	27

Snapshot of the VIEW (sensor_combined with all the values in there)

Prove that if you insert data for a sensor that is NOT registered the value will not be accepted.

DataBase Assignment



Your snapshot

Please note: During your demonstration, you must show that all assigned features are fully functional. Each missing or non-functioning feature will result in a **7-point deduction** from your grade.

Please place your final SQL code for the sensor table in your Github. Provide the link to your Github:

https://github.com/J-PachEE/SSU_EE/blob/main/EE470/hwAssignments/SQL%20database%20tables

Your Github link

Then, in the provided space below, provide the link to that web page. This page will serve as your **main web page for the remainder of the semester**, and all assignment links must be posted there. This is a simple template I have created: https://faridfarahmand.net/SampleStudentAssignmentIndex.html

DataBase Assignment

You can copy the template page and easily update it.

My Favorite Class — EE/ECE XXX
Your web page link.

2- Answer the following questions

Answer the following questions.

1. What type of database does hostinger.com use?
   MySQL
2. What type of web server does hostinger.com use?
   LiteSpeed Web Server
3. Referring to the data type tutorial (https://www.tutorialspoint.com/mariadb/mariadb_data_types.htm) assuming we have the following:

```
CREATE TABLE ints (a INT,b INT UNSIGNED,c INT ZEROFILL);
```

   Would this insertion command work? `INSERT INTO ints VALUES (-10,-10,-10);`

This would not work because the second value is a negative value and the integer for that spot must be unsigned in order to insert that value.

4. What is the purpose of having a primary key in a dBase?
   It allows you to link different data to each other from separate tables, this is important because just like this assignment we tend to separate data from the sensor and using primary keys we can tell which sensor is responsible for what reading.

5. What is the purpose of having a foreign key in a dBase?
   In this assignment it is used to ensure we cant add data to a sensor that doesn't exist, ensuring that we have a relationship for all data inserted into the tables.

6. Can a database have multiple tables?
   Yes it can, we have 2 in this assignment and we can have way more.

7. What is a relational database?
   It is a database that has some or multiple of its data linked or correlated to each other, everything is formatted in terms of tables.

8. Is MariadBase a relational or non-relational dBase?
   MariaDB is a relational database.

9. What is a database schema?

It is the structure of the database that holds the information about relationships such as what tables are related and what data belings to what table.

10. What is a Table in a relational Database?
    It is the form in which data is stored in this particualr type of database, just a combination of rows and columns that are user specified and might be linked to another table.

11. What are the main components of a Table in a relational database?
    The elements, the keys, the data types and sometimes a constraint.

12. In terms of data type, what is the difference between char(size) and varchar(size)?

    Char(size) will always save that amount of characters regardless of how much the input actually takes up, putting in place holder values if the input is smaller than the character size. Varchar as you might expect will not do that, and only use the certain amount of characters it receives.

13. What would be an example of datetime datatype format?

'2022-10-01 11:30:00'

END OF PART 1

DataBase Assignment

Appendix A

Welcome to SSU IoT Lab

Registered Sensor Nodes

node_name	manufacturer	longitude	altitude
node_1	google	50.726024000000	-6.619468000000
node_2	amazon	51.607783000000	59.869286000000

Data Received

node_name	time_received	temperature	humid
node_1	2022-10-03 11:11:20	40	50
node_2	2022-10-02 11:11:11	30	80
node_1	2022-10-04 11:11:11	44	70
node_1	2022-10-05 10:05:33	34	40
node_2	2022-10-17 10:45:33	38	70
node_1	2022-10-22 02:05:03	34	66
node_2	2022-10-22 02:05:03	34	66
node_1	2022-10-23 02:05:03	45	66
node_1	2022-10-22 11:11:11	44	60
node_1	2022-10-23 11:11:23	50	50
node_2	2022-10-23 11:11:23	50	50
node_2	2022-10-23 11:11:23	20	70
node_2	2022-10-23 09:00:00	26	76

The Average Temperature for node_1 has been: 41.57142857142857F
The Average Humidity for node_1 has been: 57.42857142857143%