

Lewis University, ABQ Campus



**Title:** **Supply Chain Software Architecture: Dynamic Database and UI Integration.**

|  |  |
| --- | --- |
| **Course code** | FA23 CPSC 61200 |
| **Course Title** | Software Design and Architecture |
| **Term** | Spring 2024 |
| **Professor** | Prof. Fuad Abu Zahra |
| **Review** | Final Review |
| **Submission Date** | March 5th 2024 |
| **SQL Environment** | Windows |

**GitHub Link:**

<https://github.com/Romeo1003/Supply-Chain-Software-Architecture.git>

**Document by:**

|  |  |  |  |
| --- | --- | --- | --- |
| Name | ID | Roles | Percentage |
| Uday Kiran Reddy Dasireddygari | L30084359 | Overall System Designer | 34% |
| Saidalli Chukka | L30087247 | Project Workflow and System Administration | 33% |
| Jagadeesh Podila | L30086157 | Documentation and Analysis | 33% |

**Introduction**  
Welcome to our project, "Supply Chain Software Architecture: Dynamic Database and UI Integration." In today's rapidly evolving business landscape, effective supply chain management is paramount for success. This project introduces a comprehensive approach to streamline supply chain processes through innovative software architecture.

Our focus lies at the intersection of three key components: supply chain management, database design, and user interface. By leveraging dynamic database updates based on real-time supply chain values, we aim to enhance operational efficiency and decision-making capabilities.

Through this introduction, we invite you to explore how our project integrates cutting-edge software architecture principles to revolutionize supply chain management. Join us on this journey as we delve into the intricate complexities of modern supply chains and uncover solutions to optimize performance and drive sustainable growth.

Alpha Tech. Homes Innovations, Inc. is a company specializing in smart-home automation solutions. We design, manufacture, and distribute a wide range of smart devices and systems for home automation, including smart lighting, security, and HVAC control.

**Let’s consider Alpha Tech. Homes Innovations Inc., as a company**

**Company Size**: Alpha Tech. Homes Innovations, Inc. is a mid-sized company with approximately 300 employees across multiple locations (USA, India, China)

**Supply Chain**:

# Raw Materials:

1. Recycled Silicon based Microprocessors and Conductors
2. Sensors (e.g., motion sensors, temperature sensors)
3. Plastic enclosures and casings

# Material Sources:

1. Recycled Silicon based Micro controllers and processors are sourced from two primary suppliers: **Supplier A** in *China* and **Supplier B** in the *Singapore*.
2. Sensors, such as motion sensors and temperature sensors, are procured from

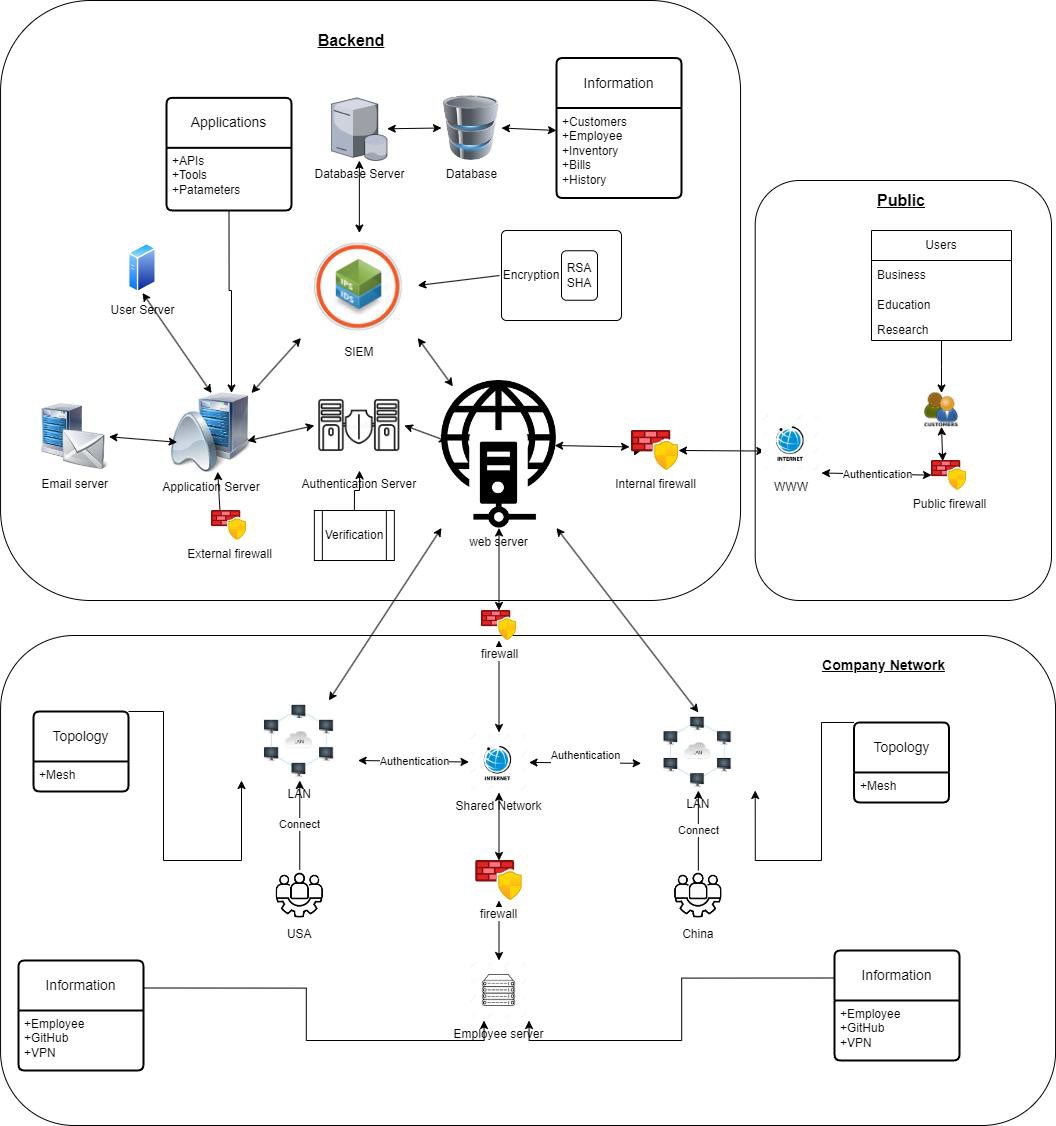
**Supplier X** in *Russia* and **Supplier Y** in *Germany*.

1. Plastic enclosures and casings are supplied by **Manufacturer P** in *India* and

**Manufacturer Q** in *Madagascar*.

**Primary Office**: Our primary office is located in Silicon Valley, *California*. Alpha Tech. Homes Innovations, Inc. operates in three cities: Silicon Valley, for research and development; *Shenzhen, China*, for manufacturing; and *New York City*, for sales and marketing.

# Model Diagram For Supply Chain Software Architecture: Dynamic Database and UI Integration



1. **Smart Devices:**

Description: These are various automation devices that can be controlled within a smart home, such as smart bulbs, smart thermostats, and smart locksAttributes:

|  |  |
| --- | --- |
| Attribute | Data Type |
| Device ID | Alphanumeric |
| Device Name | String |
| Device Type | String |
| Manufacturer | String |
| Connectivity Protocol | String |

1. **User Profiles**:

Description: These represent the individuals or users who interact with the smart home system through their user interfaces like smartphones or voice assistants.

Attributes:

|  |  |
| --- | --- |
| Attribute | Data Type |
| User ID | Alphanumeric |
| Username | String |
| Email Address | String |
| Device Preferences | JSON |
| Security Settings | JSON |

1. **Automation Rules:**

Description: These are predefined rules and conditions that govern how the smart devices should behave and interact with each other, based on user-defined criteria. Attributes:

|  |  |
| --- | --- |
| Attribute | Data Type |
| Rule ID | Alphanumeric |
| Rule Name | String |
| Trigger Device | String |
| Action Device | String |
| Conditions | JSON |

1. **Sensor Data:**

Description: This entity represents the data generated by sensors installed in the smart home, providing information about temperature, motion, and security status.

Attributes:

|  |  |
| --- | --- |
| Attribute | Data Type |
| Sensor ID | Alphanumeric |
| Sensor Type | String |
| Timestamp | Date Time |
| Sensor Value | Numeric |
| Location | String |

1. **Maintenance Logs:**

Description: This entity keeps track of maintenance and service records for the smart devices, ensuring their proper functioning and longevity.

Attributes:

|  |  |
| --- | --- |
| Attribute | Data Type |
| Log ID | Alphanumeric |
| Device ID | Alphanumeric |
| Maintenance Date | DateTime |

|  |  |
| --- | --- |
| Service Technician | String |
| Description of Maintenance | Text |

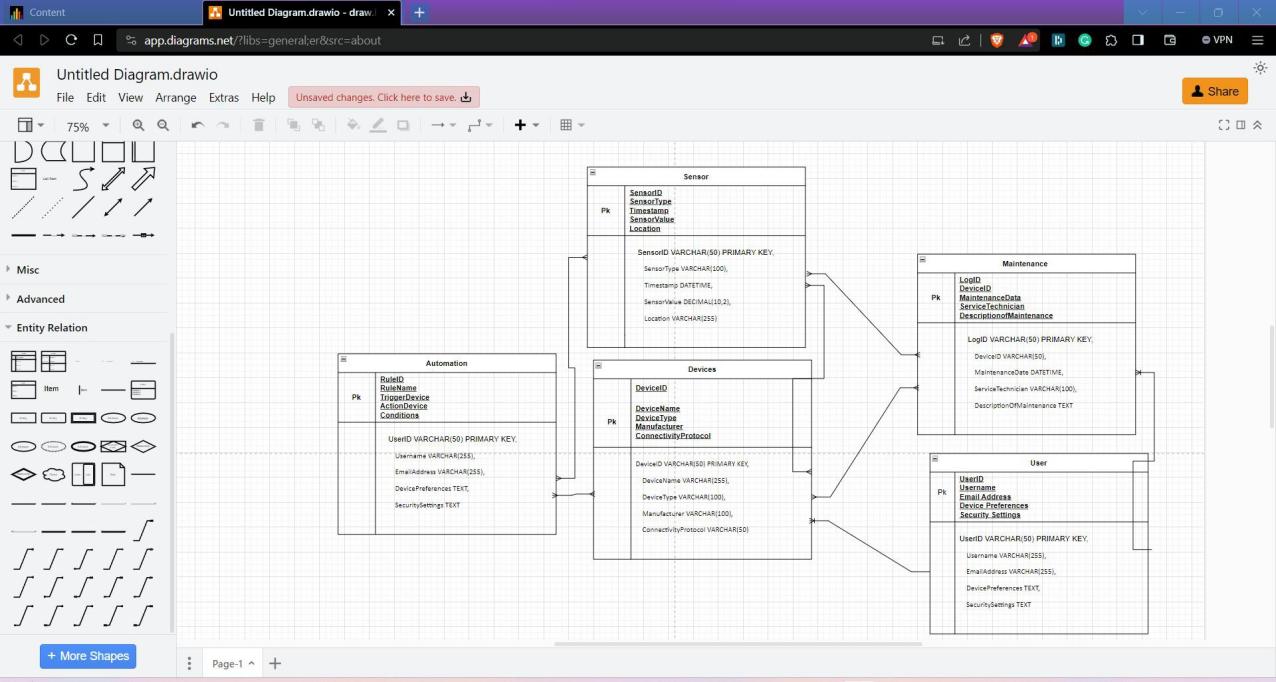
Here's a table summarizing the entities and their attributes with their respective data types:

|  |  |  |  |
| --- | --- | --- | --- |
| Entity | Description | Attributes | Data Types |
| Smart Devices | Various automation devices | Device ID, Device Name, Device Type, Manufacturer, Connectivity  Protocol | Alphanumeric, String, String, String, String |
| User Profiles | Individuals interacting with the system | User ID, Username, Email Address, Device Preferences,  Security Settings | Alphanumeric, String, String, JSON, JSON |
| Automation Rules | Predefined rules for device interactions | Rule ID, Rule Name, Trigger Device,  Action Device, Conditions | Alphanumeric, String, String, String, JSON |
| Sensor Data | Data generated by sensors | Sensor ID, Sensor Type, Timestamp,  Sensor Value, Location | Alphanumeric, String, DateTime, Numeric, String |
| Maintenance Logs | Records of device maintenance | Log ID, Device ID, Maintenance Date, Service Technician, Description of  Maintenance | Alphanumeric, Alphanumeric, DateTime, String, Text |

These entities and their attributes will be critical for managing and maintaining the smart home automation system effectively.

CSV and XMl files into a zip:

# ERD Model



**description of the relationships**

**Entities:**

**User**: Attributes might include UserID, Username, Email, Device Preferences,SecuritySettings.

**Devices**: Attributes can be DeviceID, DeviceType, DeviceName, ConnectivityProtocol, Manufacturer.

**Automation**: Attributes like RuleID, RuleName, TriggerDevice, ActionDevice, Condition could be included.

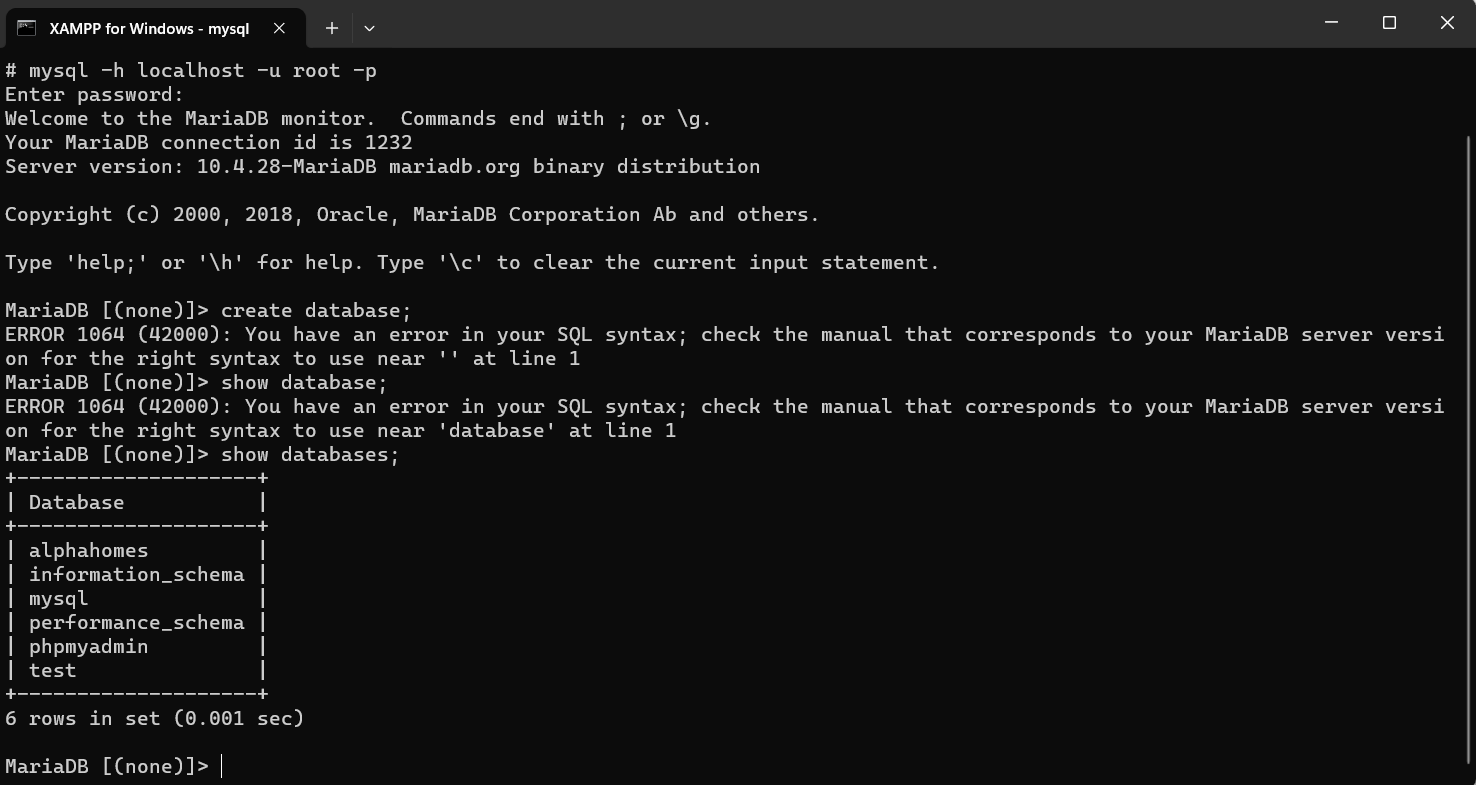
**Maintenance**: This could link SmartDevices to User, Sensor, Devices… LogID, DeviceID, MaintenanceData, ServiceTechnician, DescriptionofMaintenance could be included Sensor: Linking Users to SmartDevices to track usage. Attributes might include SensorID, SensorType, Timestamp, SensorValue, Location.

**Relationships:**

Users own one or more SmartDevices and Maintenance (one-to-many).

Devices have sensors and automationtules and need maintainance (many-to-many). Users use one or more SmartDevices (one-to-many).

Automation can be done to all devices and sensors (many-to-many).net shell



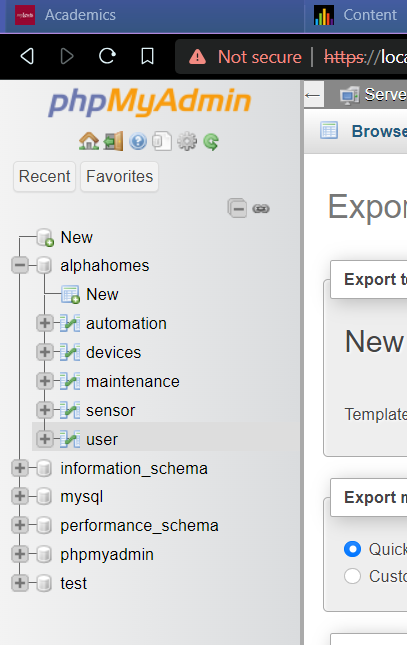
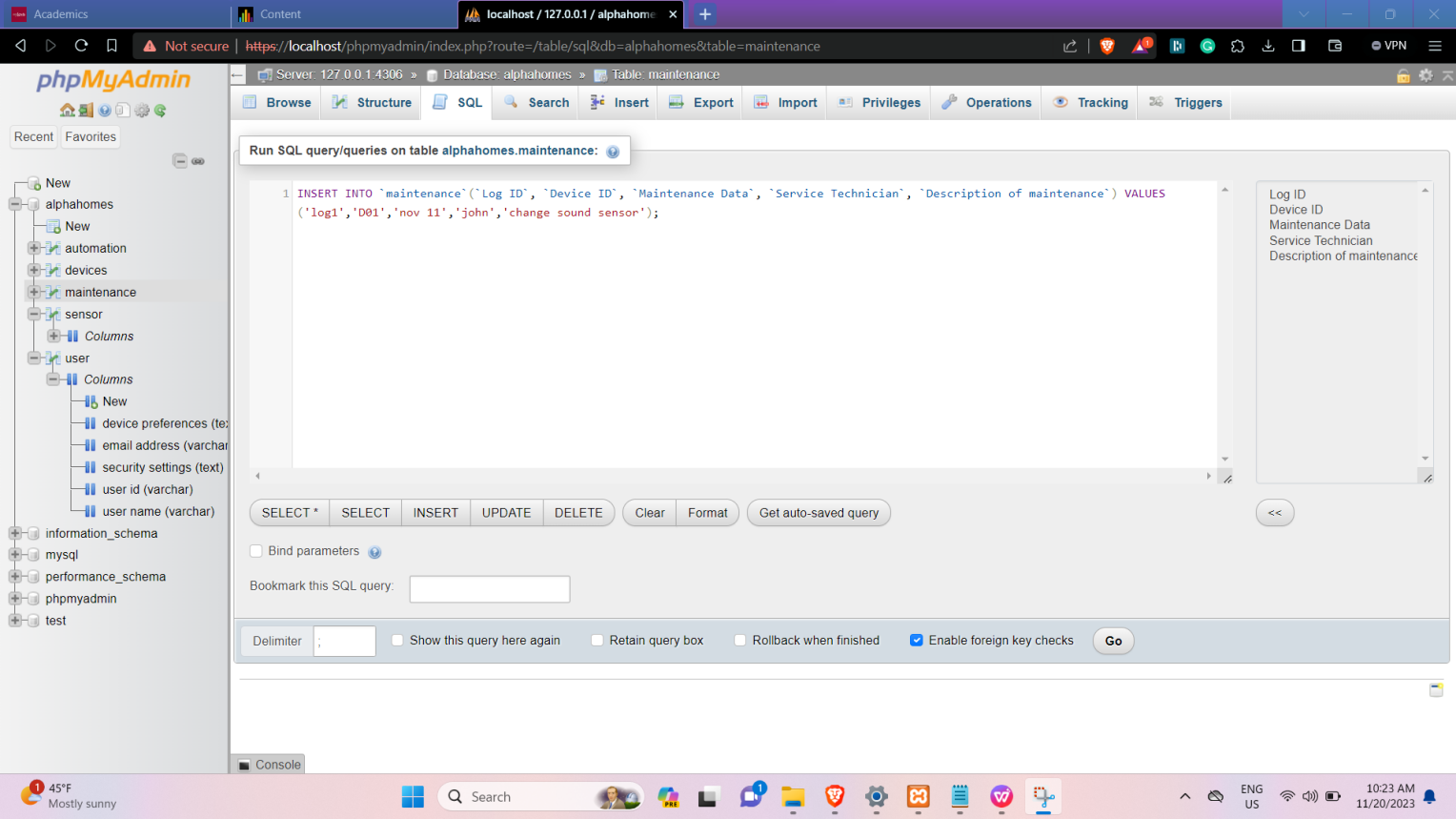
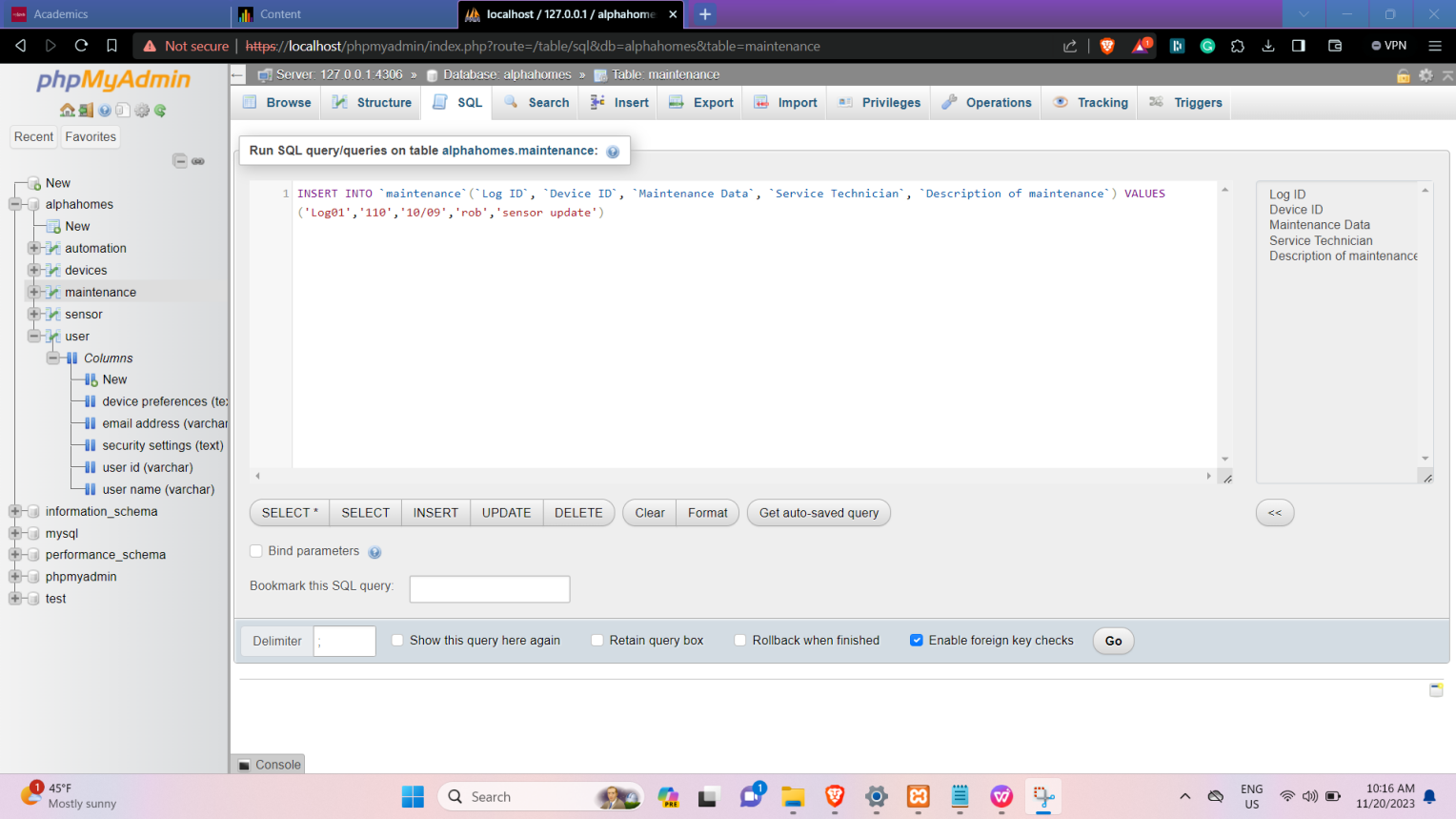
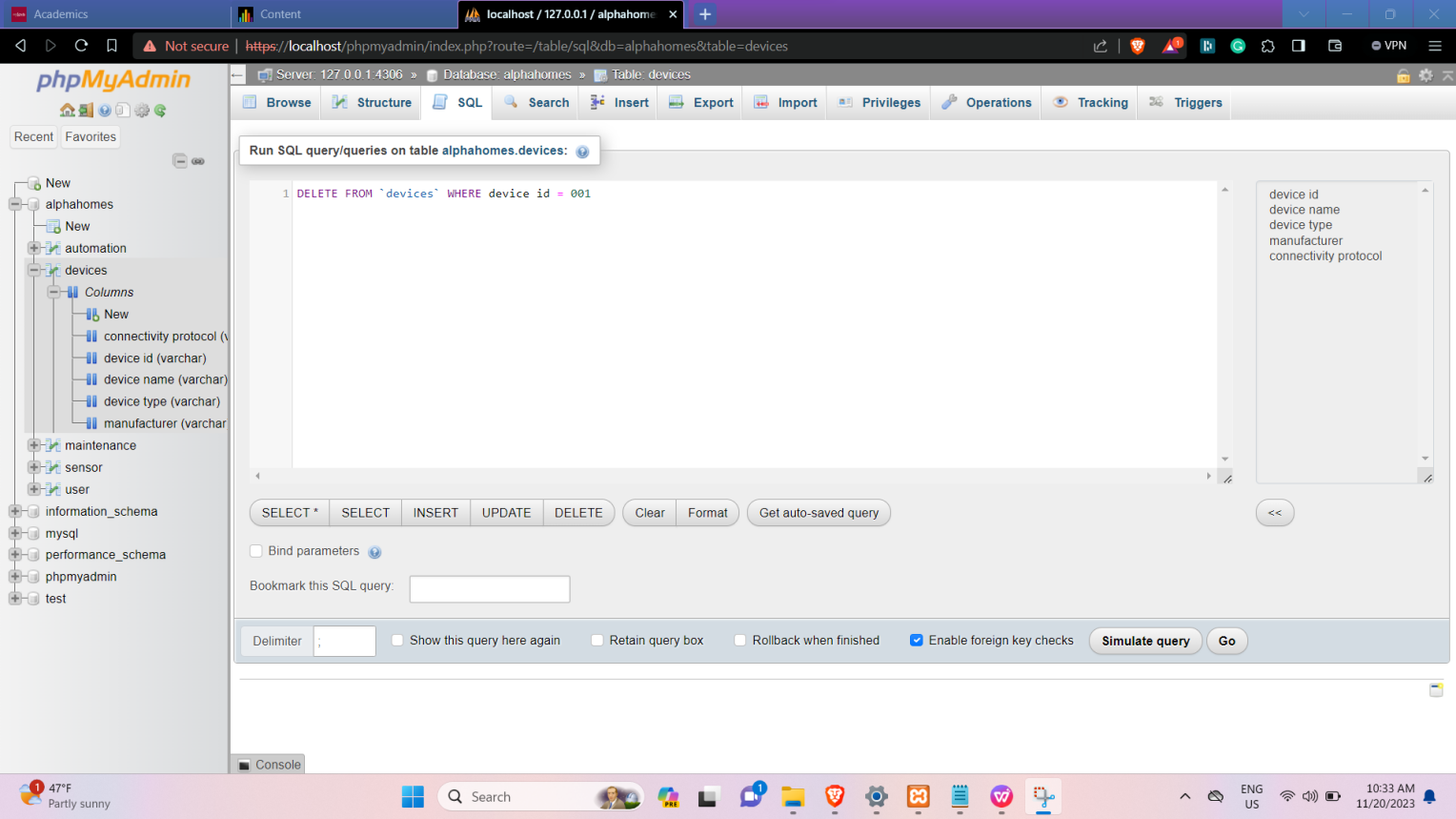
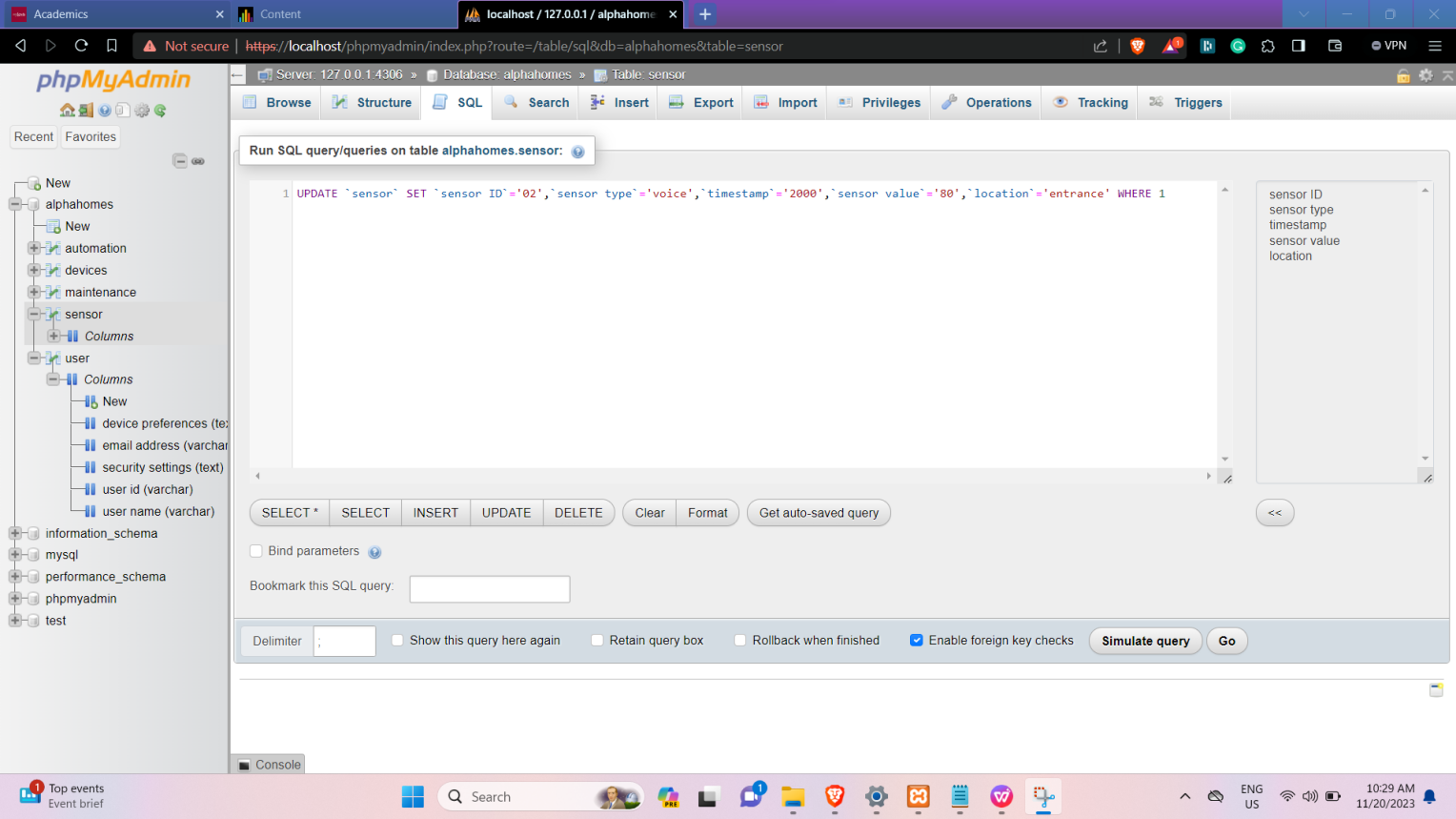
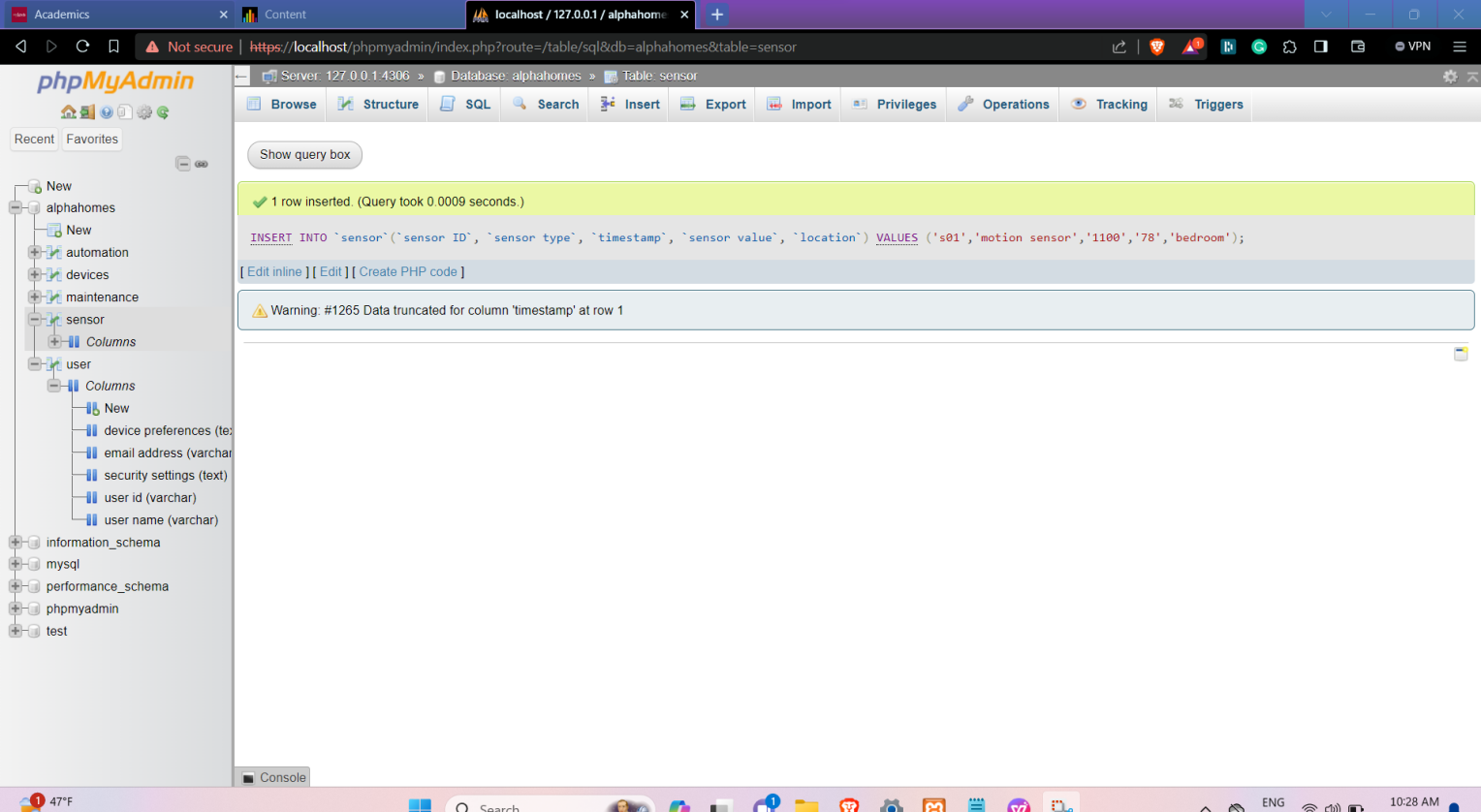
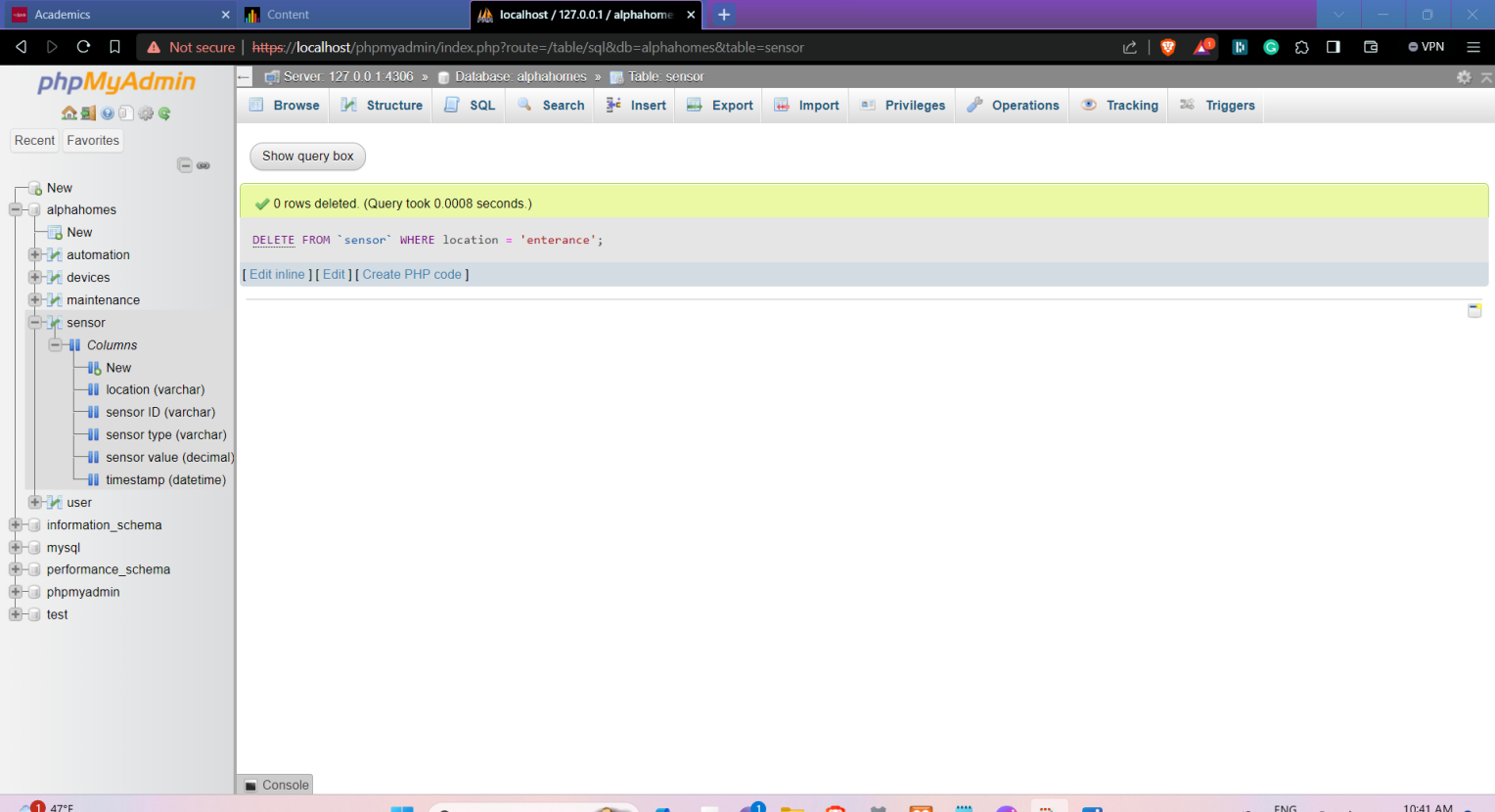
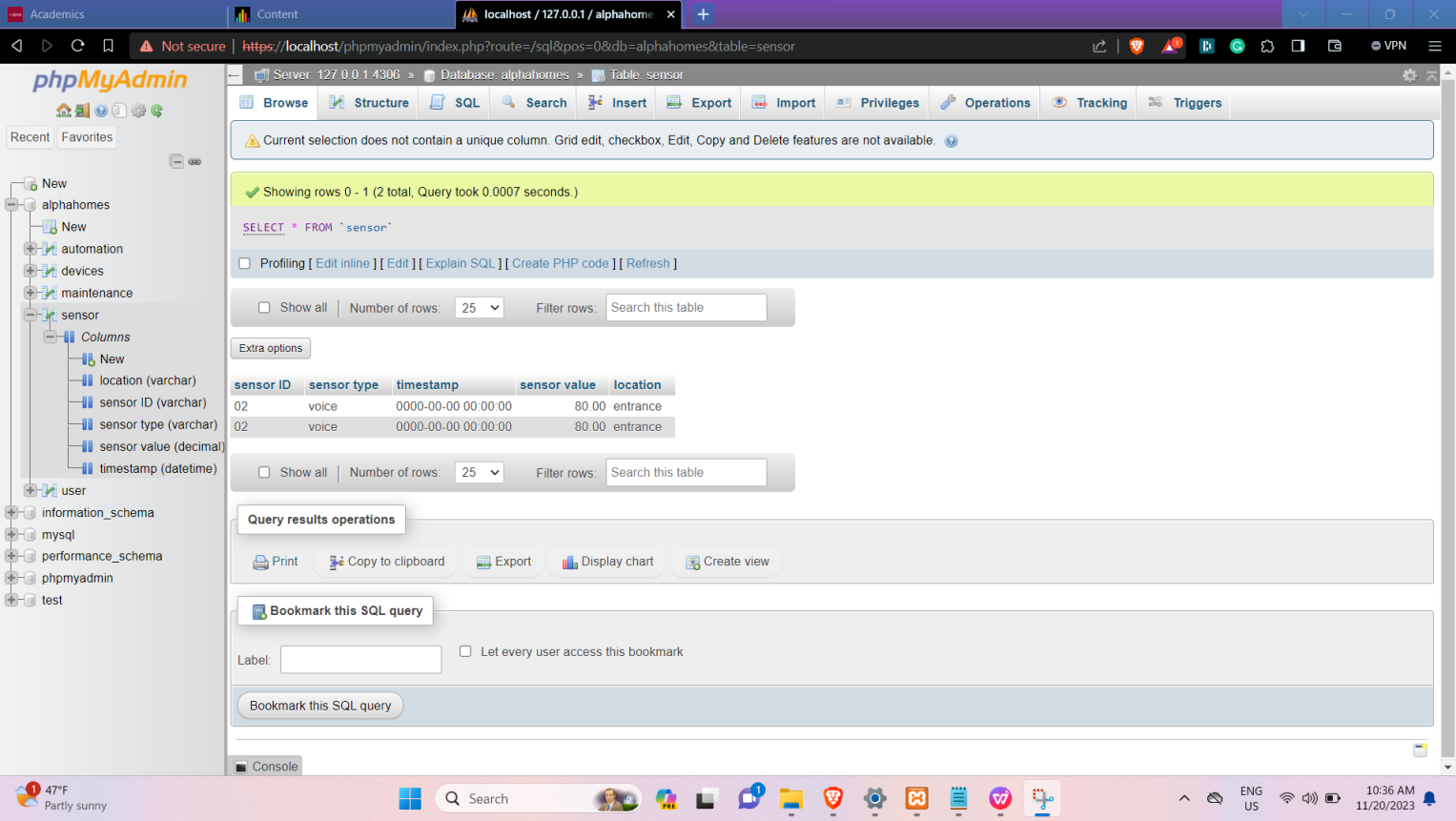
Tables created  


Table content:  
data insertion, upgradation and deletion:  


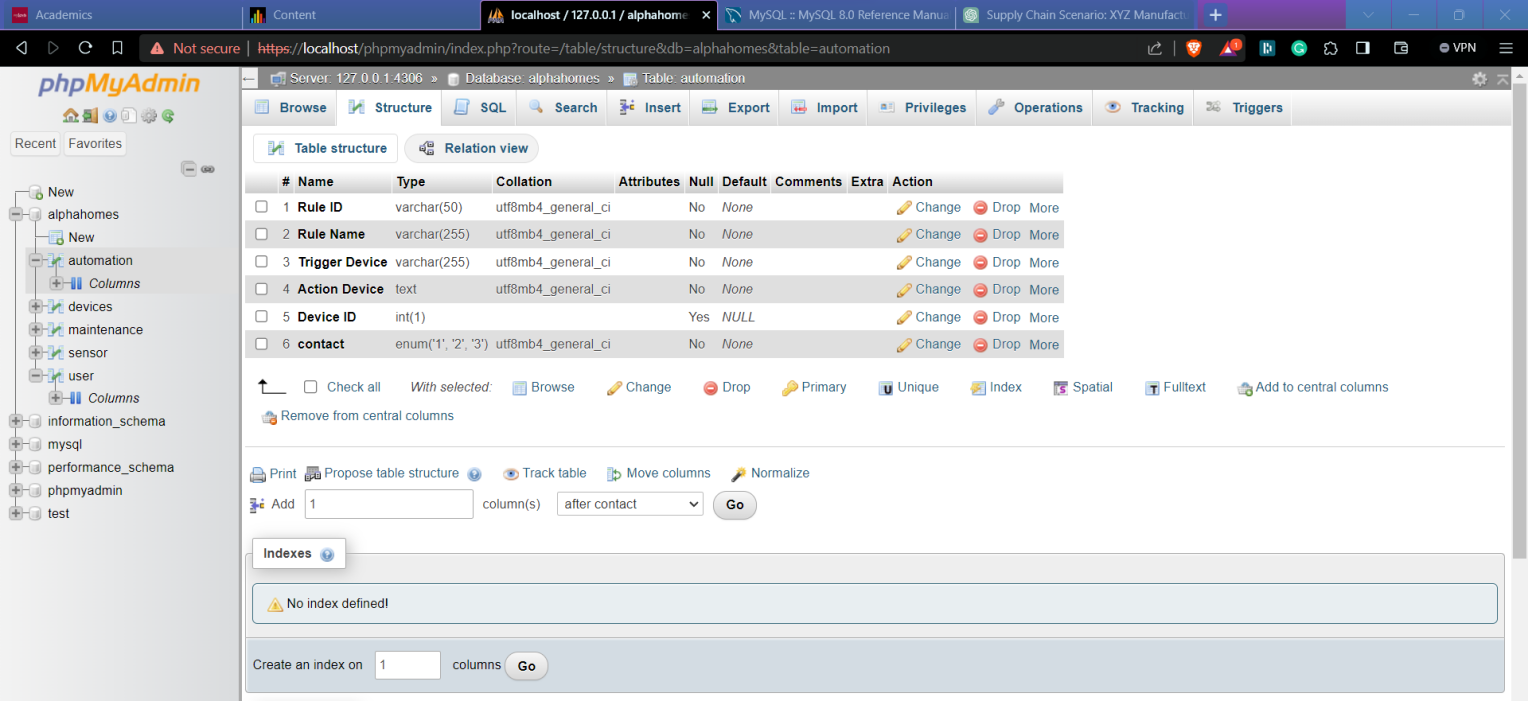


Show data: simple select statement.



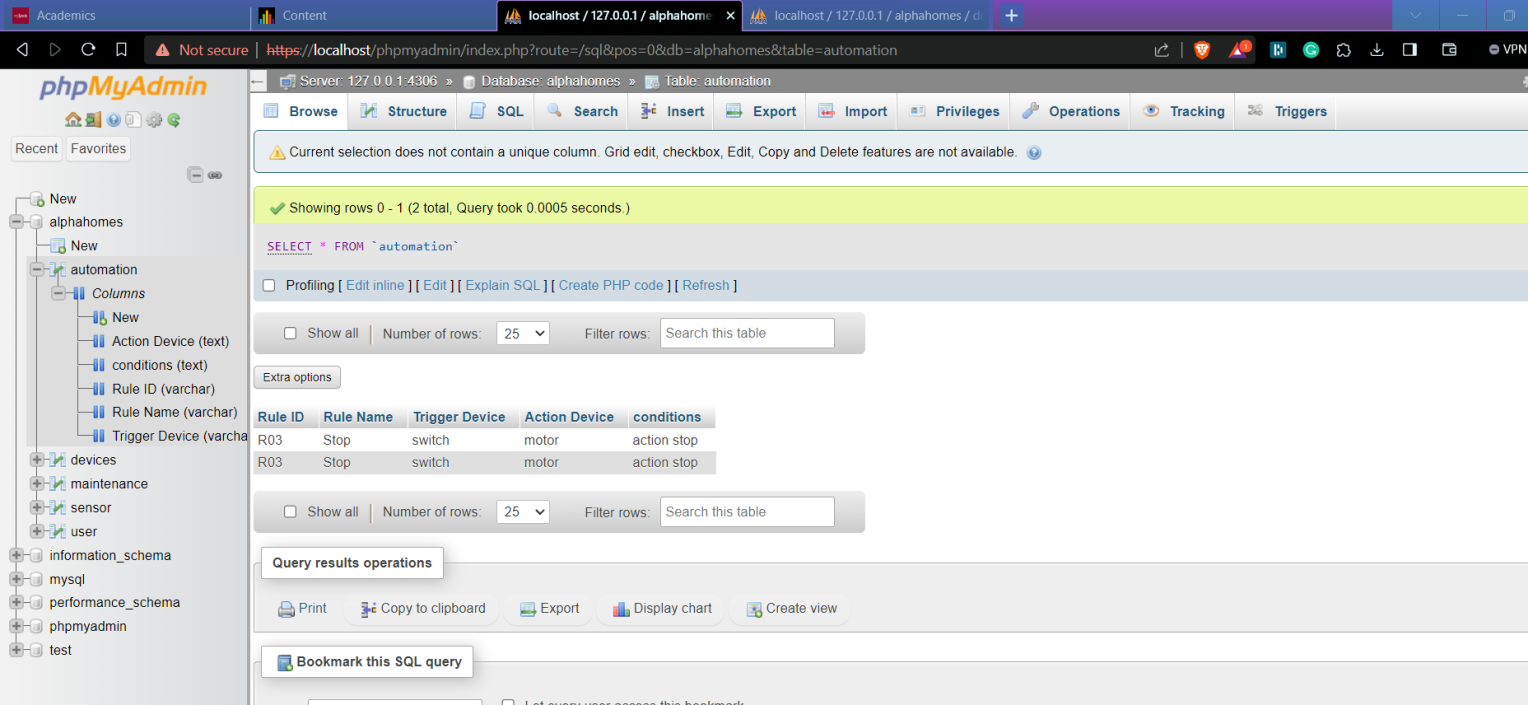
Inserting empty values in ENNUM:  


Joined table using enum:Device ID common key in device table and automation table

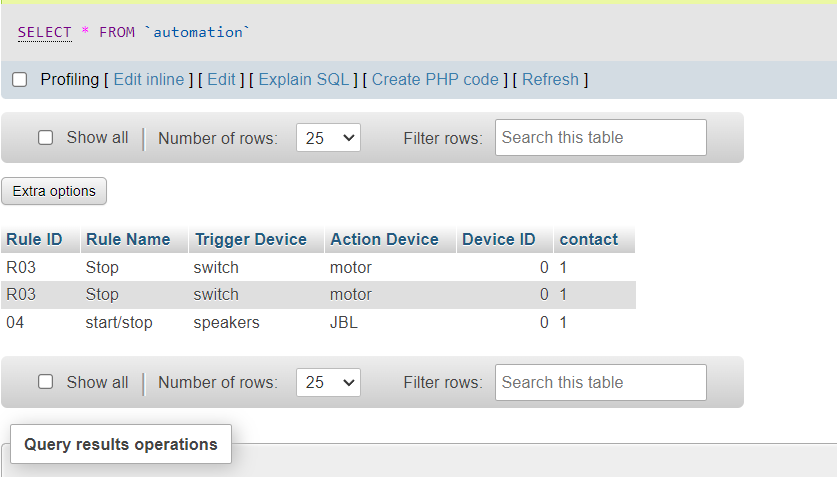


Updated values

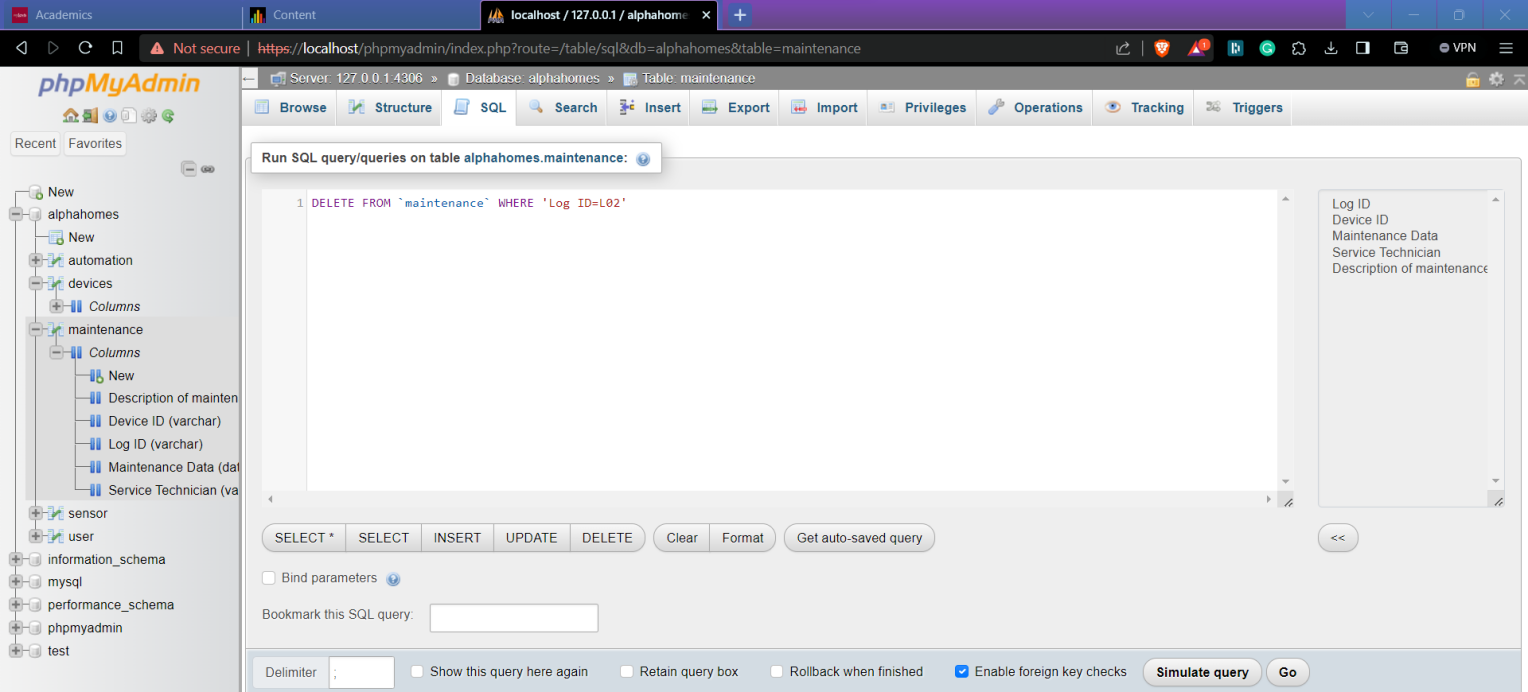
before

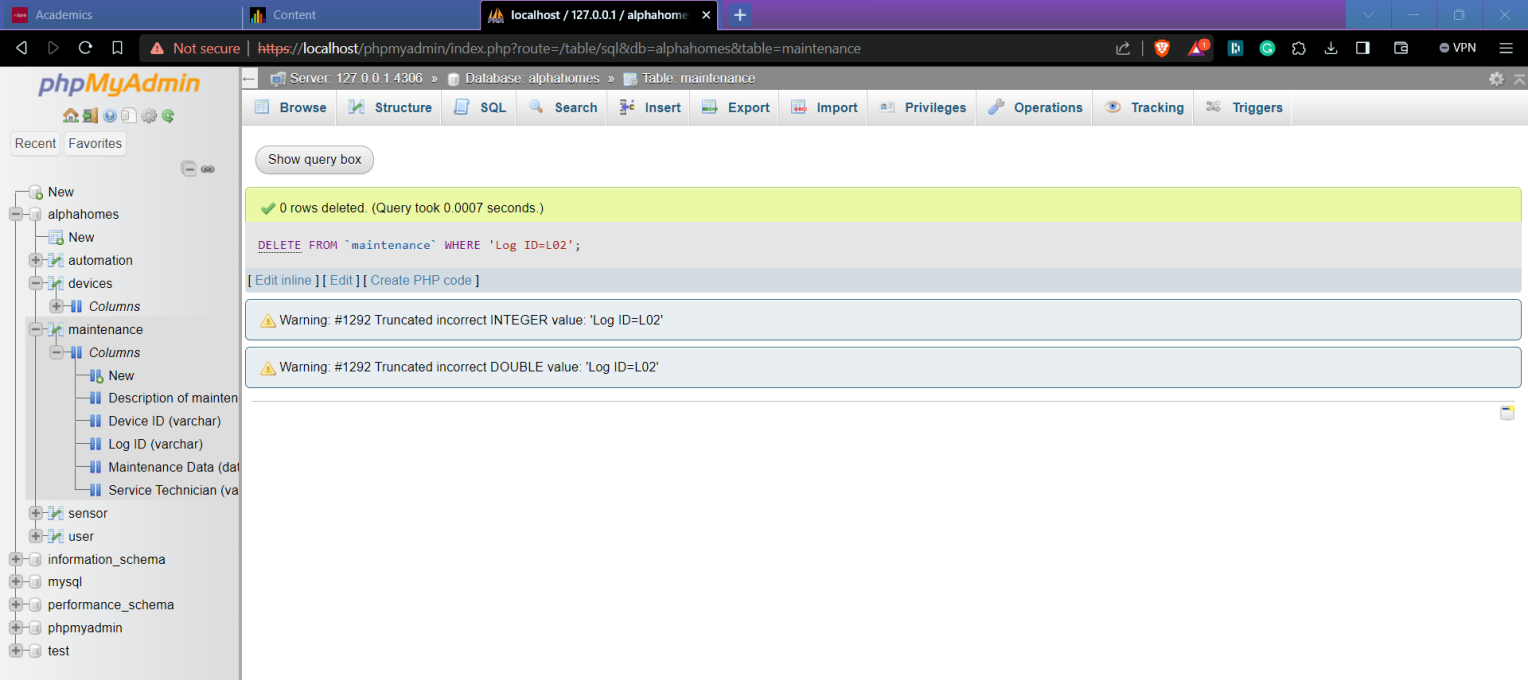
0

After:



“Where” command





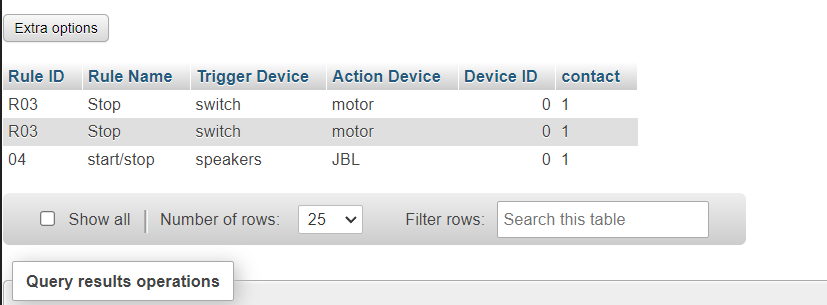
Auto increment:



If Default constraint is not mentioned it creates a new record in table:  
when default constaint is mentioned:



If not then.



Final Project :

#final code:

import mysql.connector

# Constants

HOST = '127.0.0.1'

HOST2 = '10.0.0.87'

# ... other constants if needed

# Function to establish a connection to the database

def connect\_to\_db():

try:

# Establish connection

cnx = mysql.connector.connect(

host=HOST,

user='Alpha',

password='8133',

database='alphahomes'

)

return cnx

except mysql.connector.Error as err:

print(f"Error: {err}")

return None

# Function to display the menu options

def display\_menu():

print("MENU:")

print("1. View Product Table")

print("2. View Customer Info Table")

print("3. View Sales Records")

print("4. Place an Order")

print("Q. Quit")

# Main function

def main():

connection = connect\_to\_db()

if connection:

# Loop to display menu until user chooses to quit

while True:

display\_menu()

choice = input("Enter your choice: ")

if choice.upper() == '1':

# Option 1: View Product Table

# ... fetch and display product table

elif choice.upper() == '2':

# Option 2: View Customer Info Table

# ... fetch and display customer info table

elif choice.upper() == '3':

# Option 3: View Sales Records

# ... fetch and display sales records

elif choice.upper() == '4':

# Option 4: Place an Order

# ... logic for placing an order

elif choice.upper() == 'Q':

break

else:

print("Invalid choice. Please choose again.")

connection.close()

# Execute the main function

if \_\_name\_\_ == "\_\_main\_\_":

main()