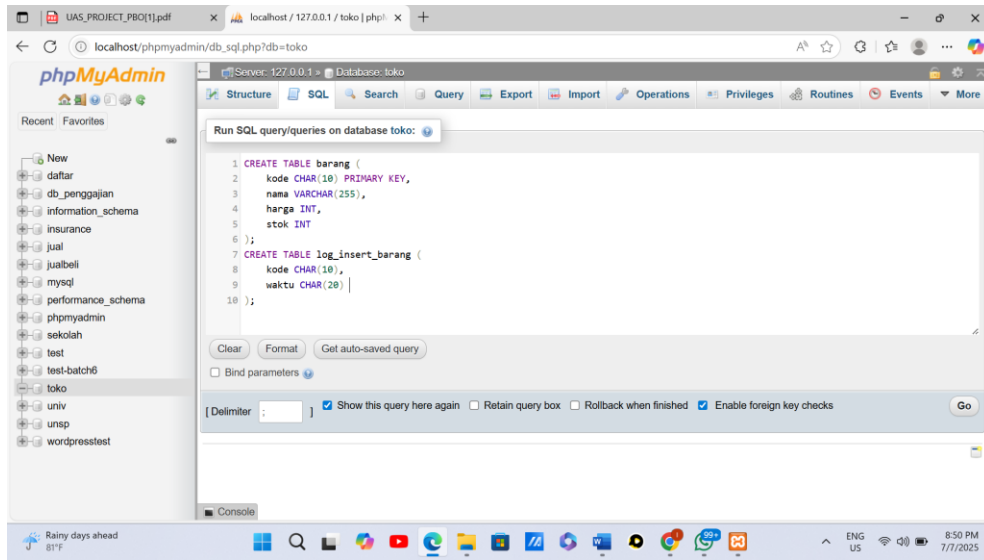
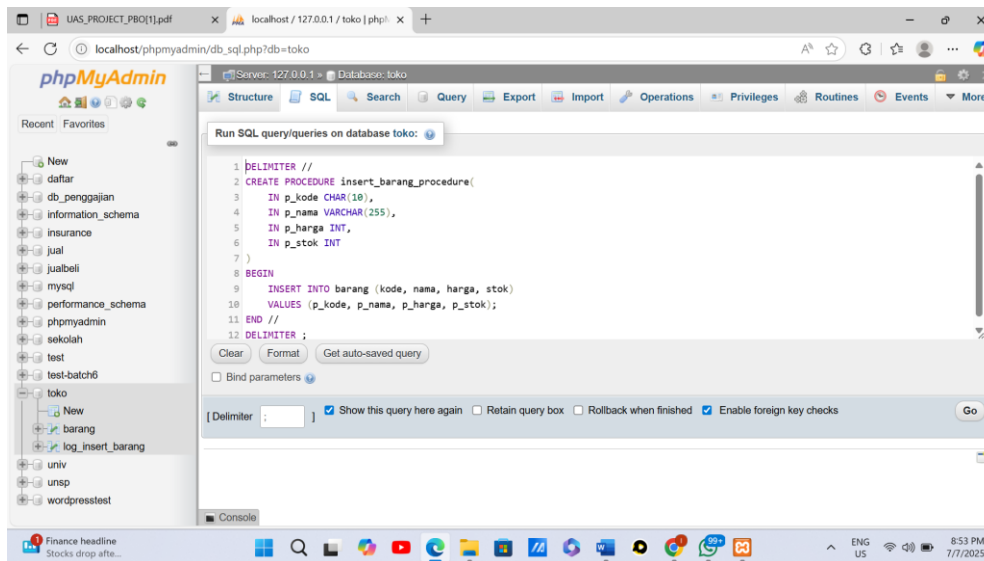


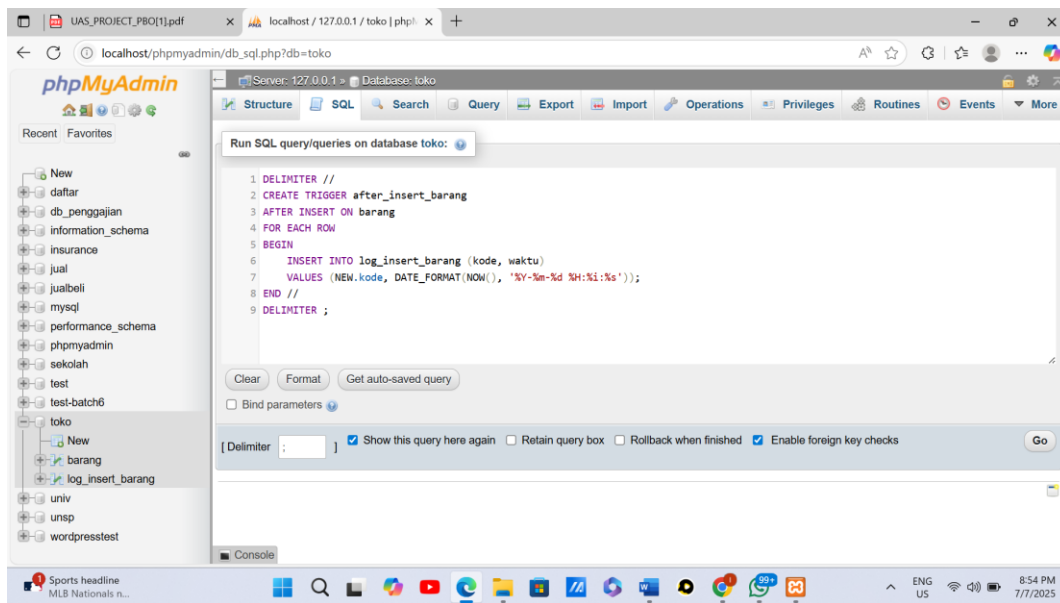
1. Membuat table “barang” dan “log_insert_barang”



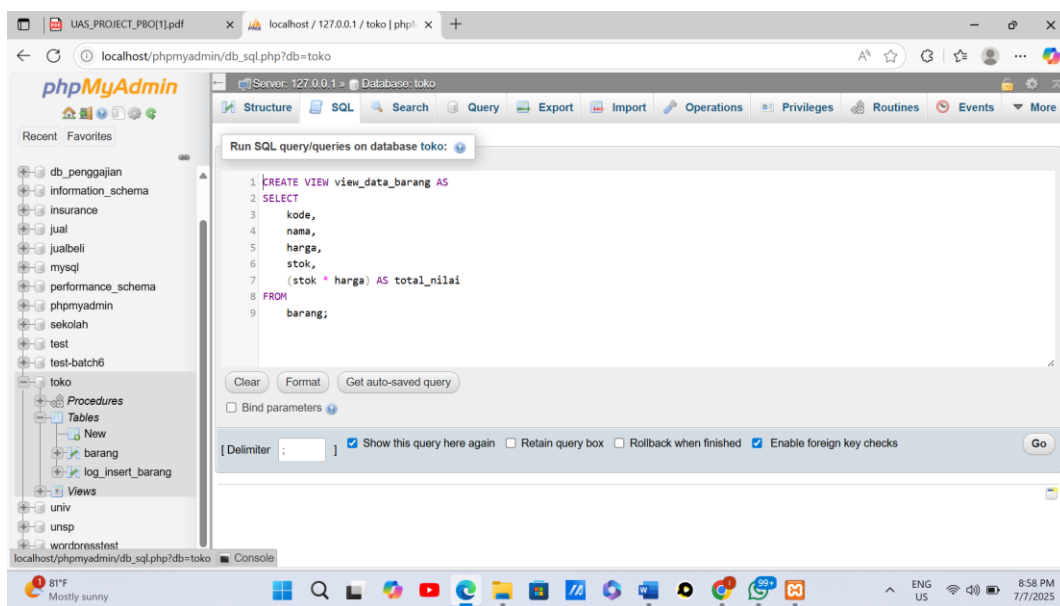
2. Membuat prosedur; Procedure ini akan digunakan untuk menyisipkan data ke tabel



3. Membuat trigger



4. Membuat view



5. Direktori Program

MyJavaProject/

└─ src/

| └─ com/

| └─ example/

| └─ app/

| └─ InsertData.java

| └─ DisplayData.java

└─ lib/

└─ mysql-connector-java-x.x.x.jar

6. Database sebelum program jalan

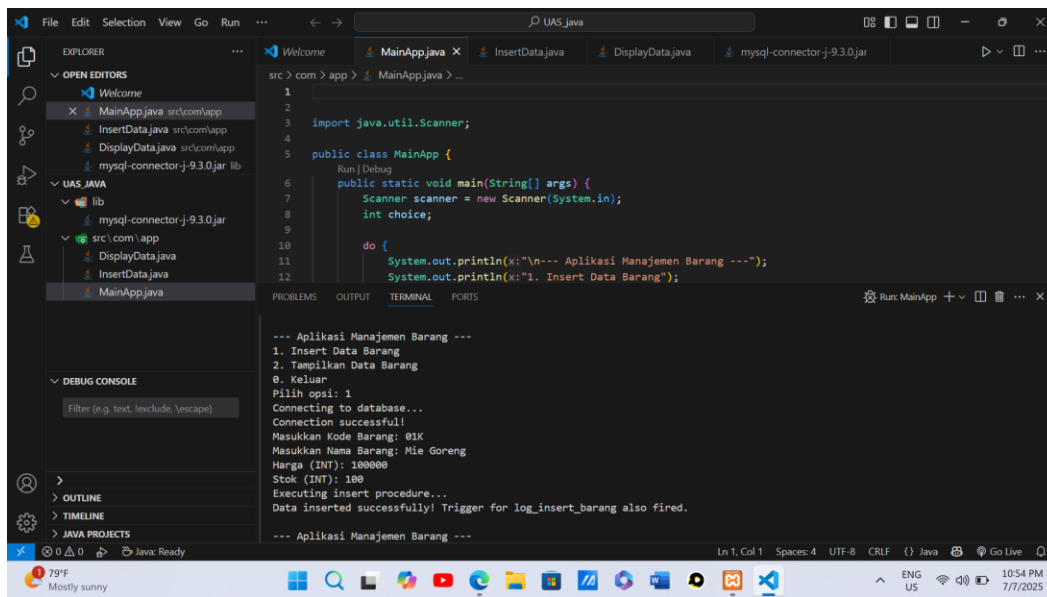
The image consists of two screenshots of the phpMyAdmin web interface, showing the state of a database before a program is executed.

Top Screenshot: The interface shows the 'Database: toko' and 'Table: barang'. The SQL query is `SELECT * FROM `barang``. The result is an empty set. The table structure is visible below the query, showing columns: `kode`, `nama`, `harga`, and `stok`.

Bottom Screenshot: The interface shows the 'Database: toko' and 'Table: log_insert_barang'. The SQL query is `SELECT * FROM `log_insert_barang``. The result is an empty set. The table structure is visible below the query, showing columns: `kode` and `waktu`.

Both screenshots show the phpMyAdmin interface with the 'Server: 127.0.0.1' and 'Database: toko' information. The left sidebar shows the database structure, including 'Procedures', 'Tables', and 'Views'. The bottom status bar indicates the system is 'Mostly sunny' with a temperature of '78°F' and the time is '9:30 PM 7/7/2025'.

7. Runing insert data

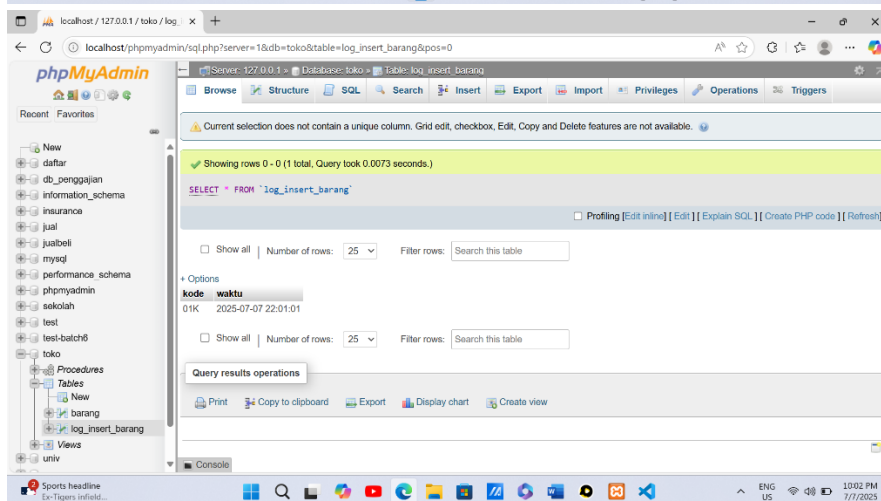
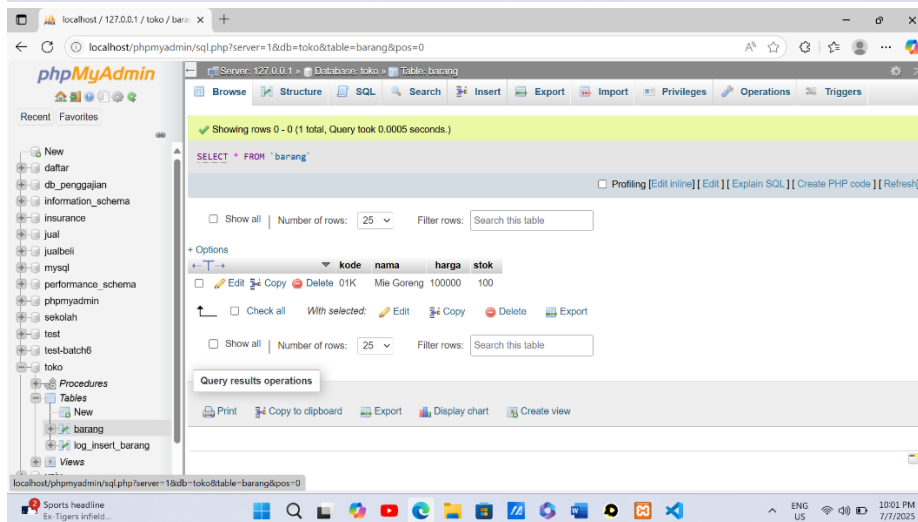


The screenshot shows an IDE with a project named 'UAS_Java'. The 'MainApp.java' file is open, displaying the following code:

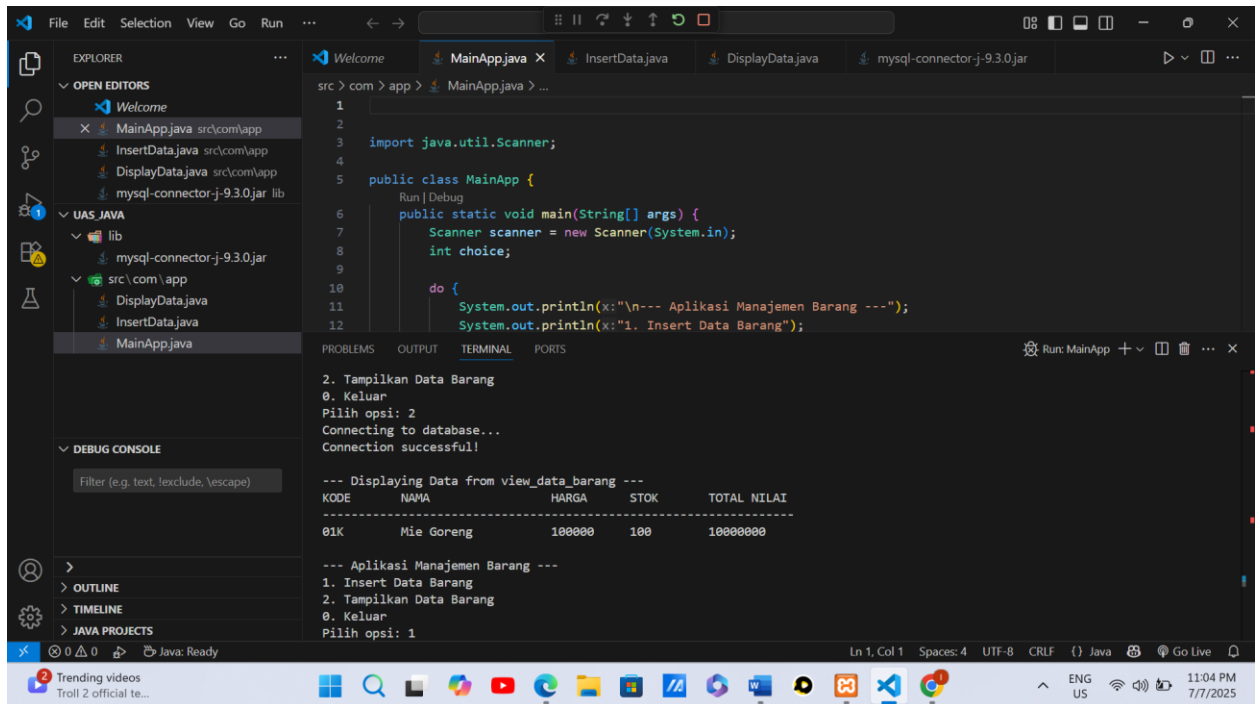
```
1  
2  
3 import java.util.Scanner;  
4  
5 public class MainApp {  
6     public static void main(String[] args) {  
7         Scanner scanner = new Scanner(System.in);  
8         int choice;  
9  
10        do {  
11            System.out.println(x:"\n--- Aplikasi Manajemen Barang ---");  
12            System.out.println(x:"1. Insert Data Barang");  
13        } while (true);  
14    }  
15 }
```

The 'TERMINAL' tab shows the execution output:

```
--- Aplikasi Manajemen Barang ---  
1. Insert Data Barang  
2. Tampilkan Data Barang  
0. Keluar  
Pilih opsi: 1  
Connecting to database...  
Connection successful!  
Masukkan Kode Barang: 01K  
Masukkan Nama Barang: Mie Goreng  
Harga (INT): 100000  
Stok (INT): 100  
Executing insert procedure...  
Data inserted successfully! Trigger for log_insert_barang also fired.  
--- Aplikasi Manajemen Barang ---
```



8. Implementasi View untuk menampilkan display data insert dan tambahan kolom Total nilai



```
src > com > app > MainApp.java > ...
1
2
3 import java.util.Scanner;
4
5 public class MainApp {
6     public static void main(String[] args) {
7         Scanner scanner = new Scanner(System.in);
8         int choice;
9
10        do {
11            System.out.println(x:"\n--- Aplikasi Manajemen Barang ---");
12            System.out.println(x:"1. Insert Data Barang");
```

2. Tampilkan Data Barang
0. Keluar
Pilih opsi: 2
Connecting to database...
Connection successful!

--- Displaying Data from view_data_barang ---

KODE	NAMA	HARGA	STOK	TOTAL NILAI
01K	Mie Goreng	100000	100	10000000

--- Aplikasi Manajemen Barang ---
1. Insert Data Barang
2. Tampilkan Data Barang
0. Keluar
Pilih opsi: 1