Nama : Helsa Vania Siringoringo

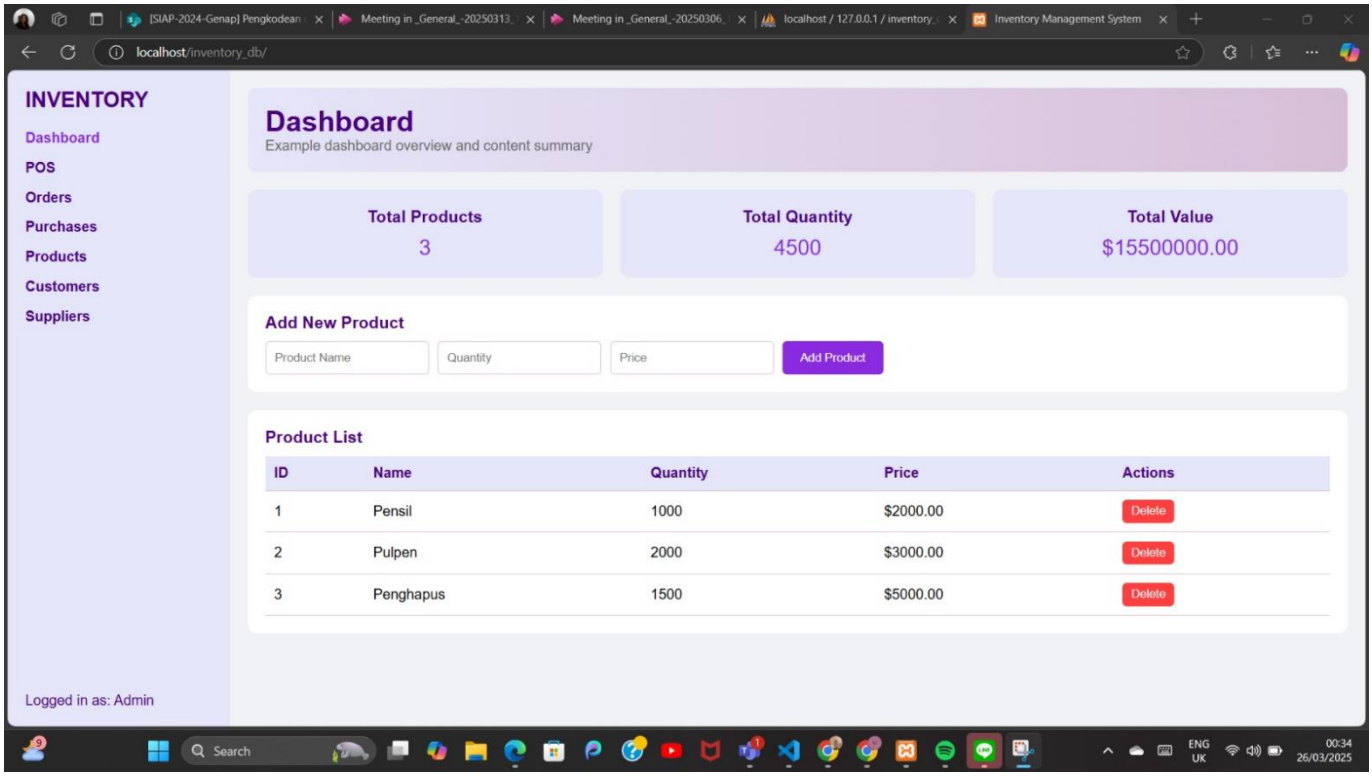
NIM : 12030123140297

Kelas : E

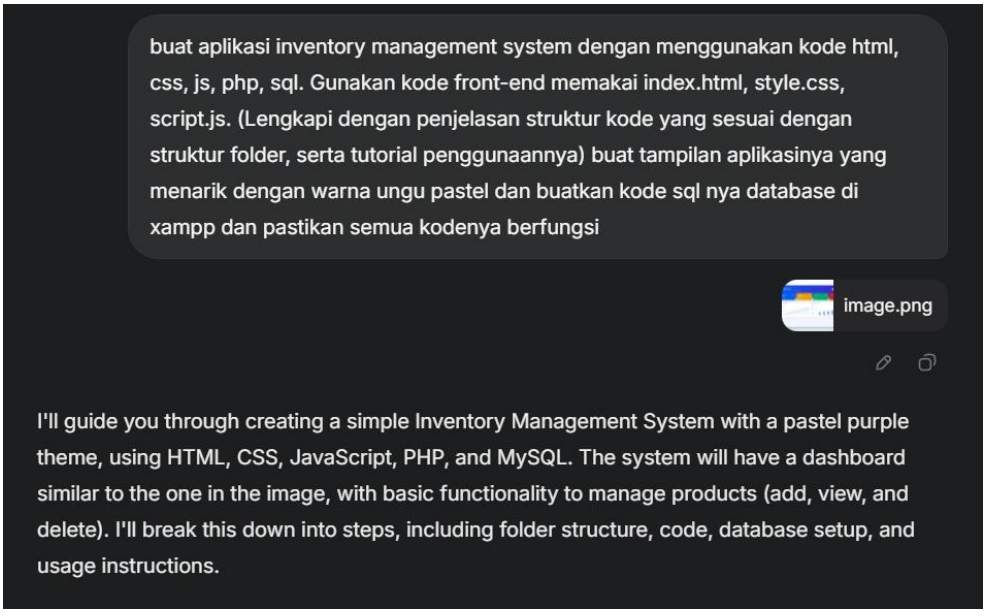
Mata Kuliah : Pengkodean dan Pemrograman

Tugas 5 – Membuat Web Inventory Management System

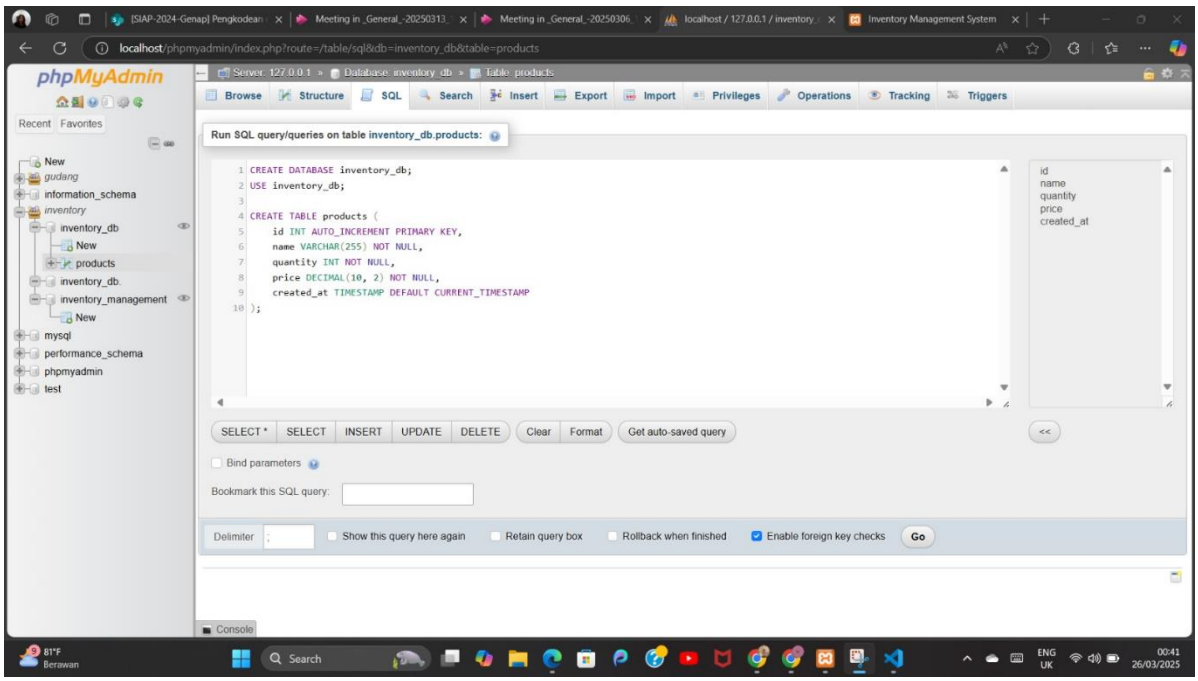
Tampilan Web Inventory Management System :



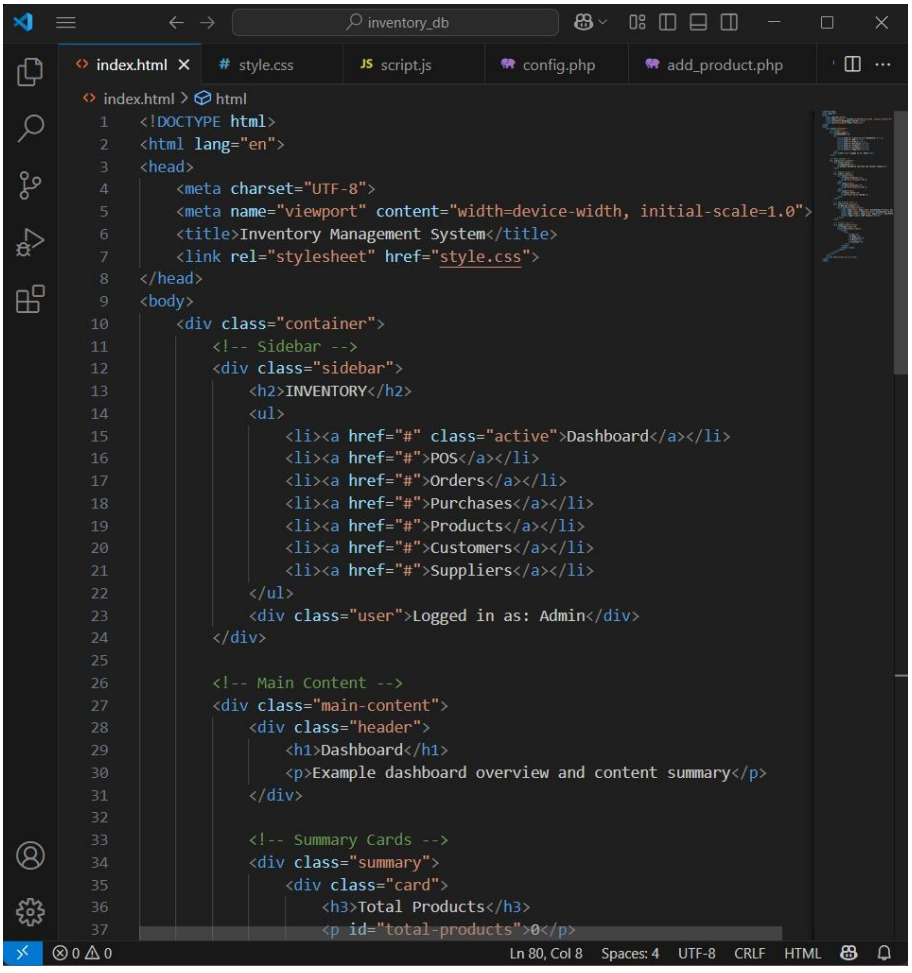
Step 1 : Buka AI lalu masukkan gambar yg sudah diambil dari Github, tuliskan perintah seperti yg saya buat dibawah (saya menggunakan Grok AI).

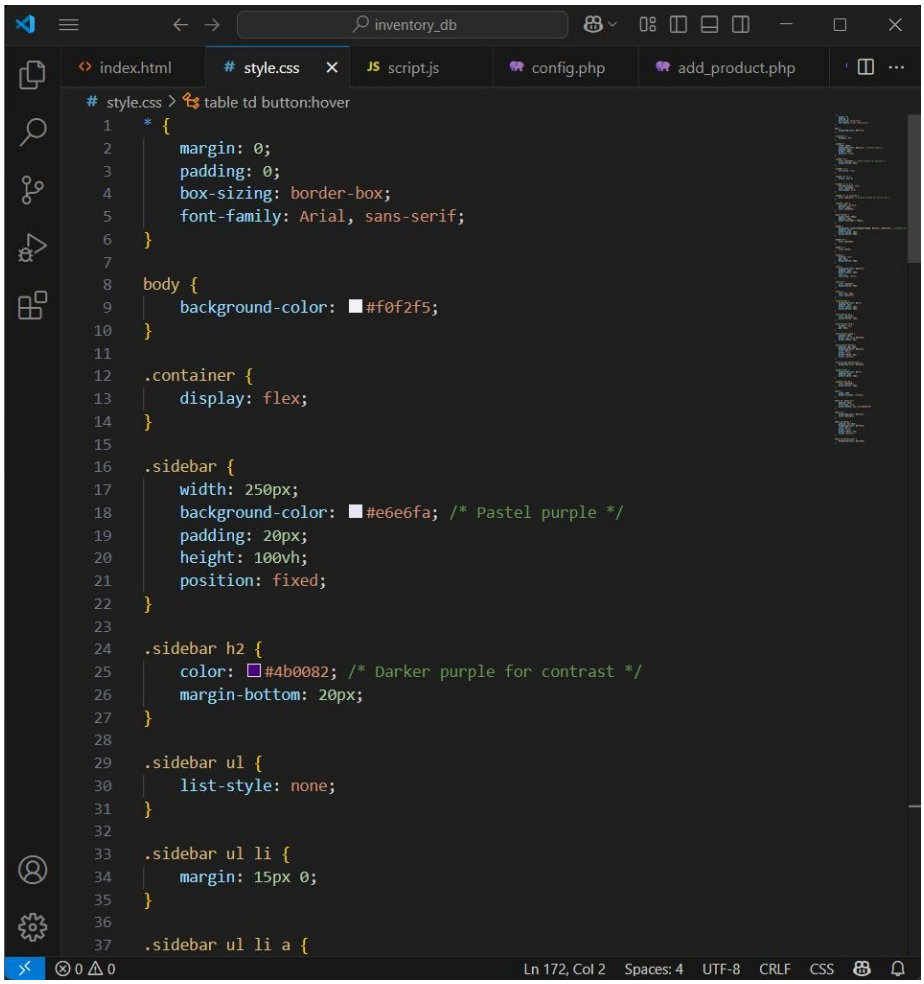


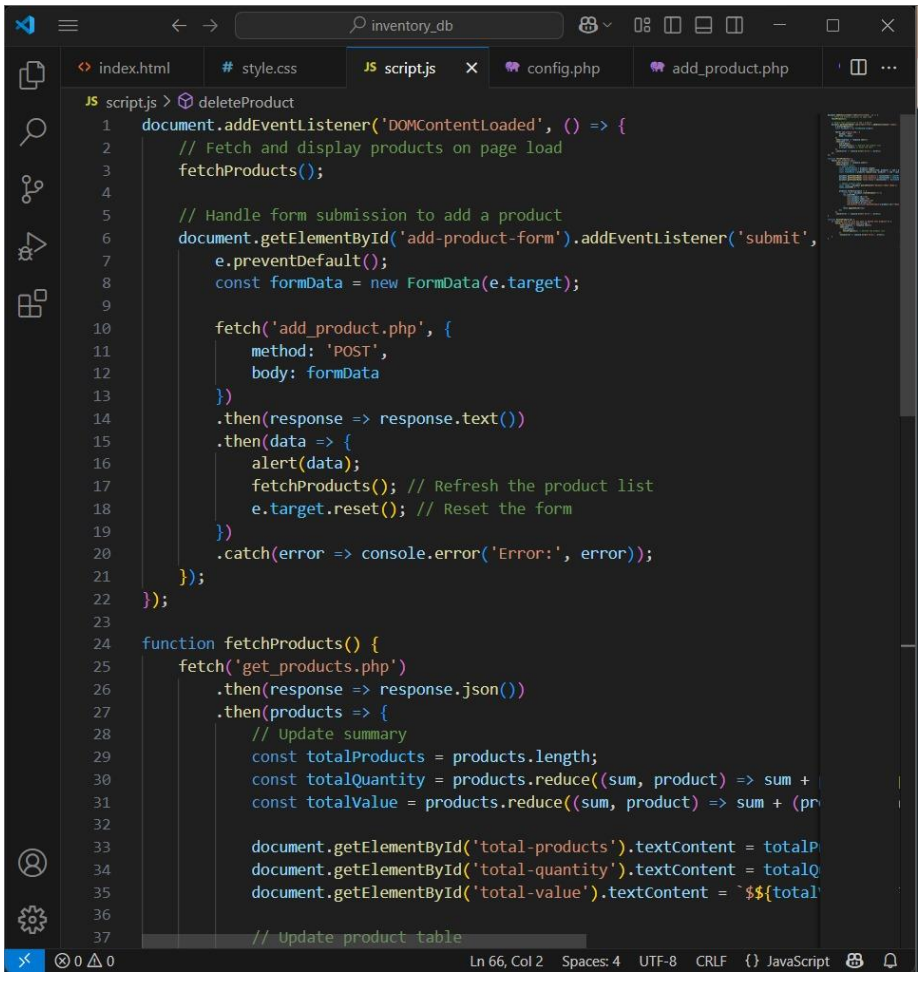
Step 2 : Setelah AI merespon, akan diberikan structure folder beserta kode-kode yg dibutuhkan untuk membuat Inventory Management System. Lalu buat database di phpMyAdmin dengan kode sql yg sudah diberikan.

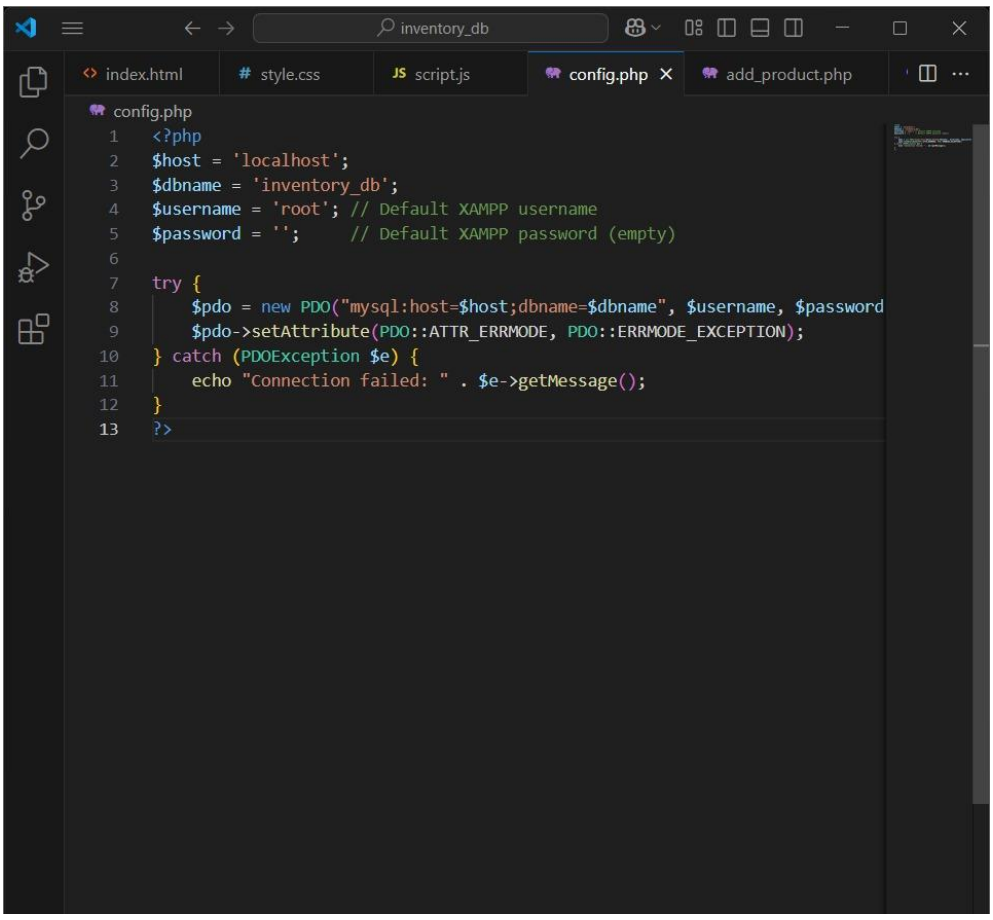


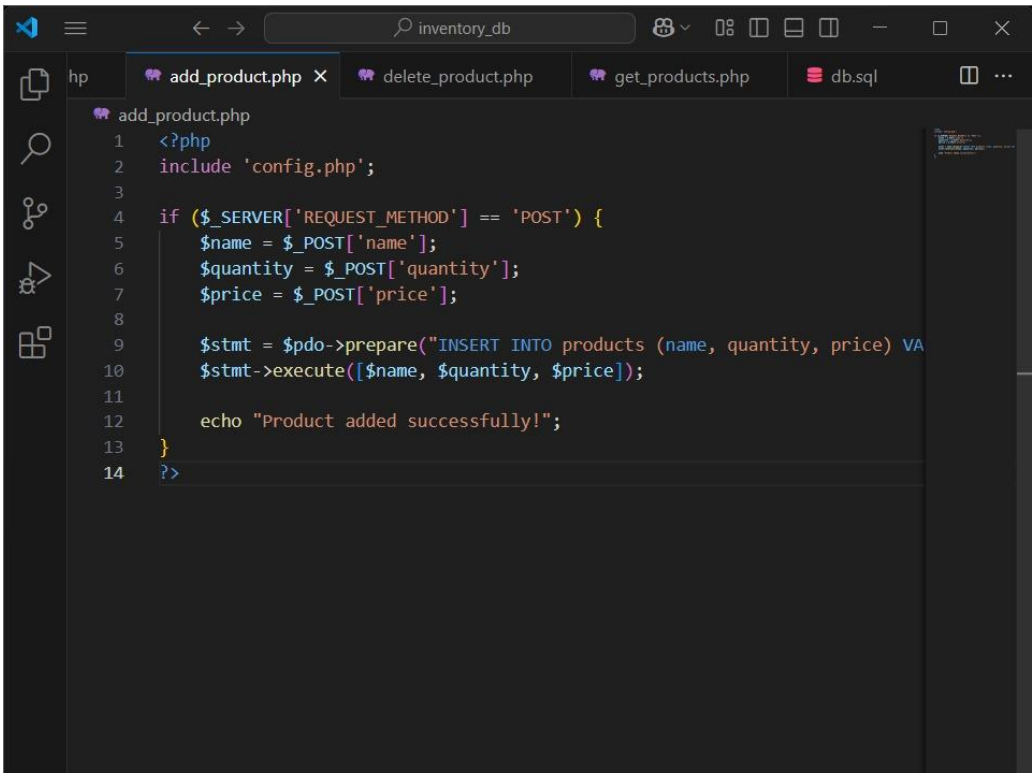
Step 3 : Setelah itu, masukkan kode-kode lainnya (html, style.css, javascript, dll) ke dalam Visual Code Studio, jangan lupa untuk save folder ke dalam Data C lalu klik xampp, klik htdocs, dan buat folder baru bernama “inventory_db”.

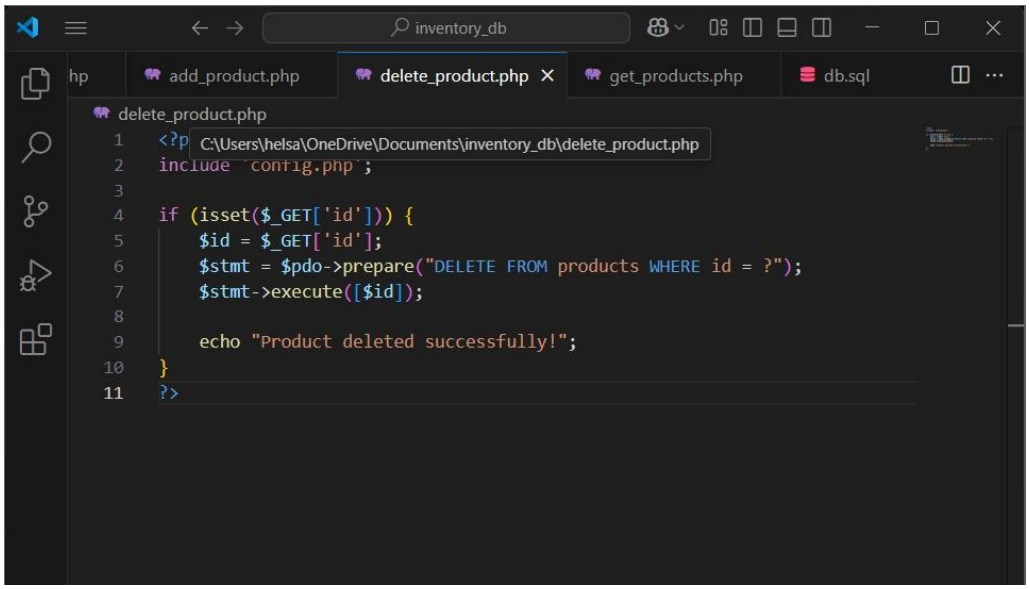


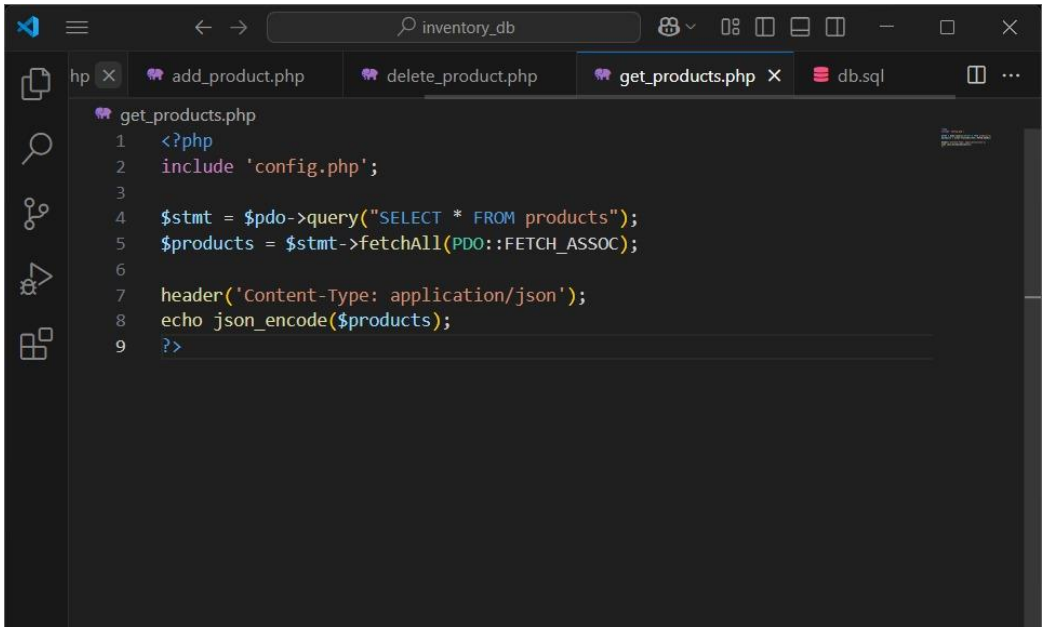

inventory_db
index.html
style.css
script.js
config.php
add_product.php
style.css > table td button:hover
* {
margin: 0;
padding: 0;
box-sizing: border-box;
font-family: Arial, sans-serif;
}
body {
background-color: #f0f2f5;
}
.container {
display: flex;
}
.sidebar {
width: 250px;
background-color: #e6e6fa; /* Pastel purple */
padding: 20px;
height: 100vh;
position: fixed;
}
.sidebar h2 {
color: #4b0082; /* Darker purple for contrast */
margin-bottom: 20px;
}
.sidebar ul {
list-style: none;
}
.sidebar ul li {
margin: 15px 0;
}
.sidebar ul li a {
Ln 172, Col 2
Spaces: 4
UTF-8
CRLF
CSS

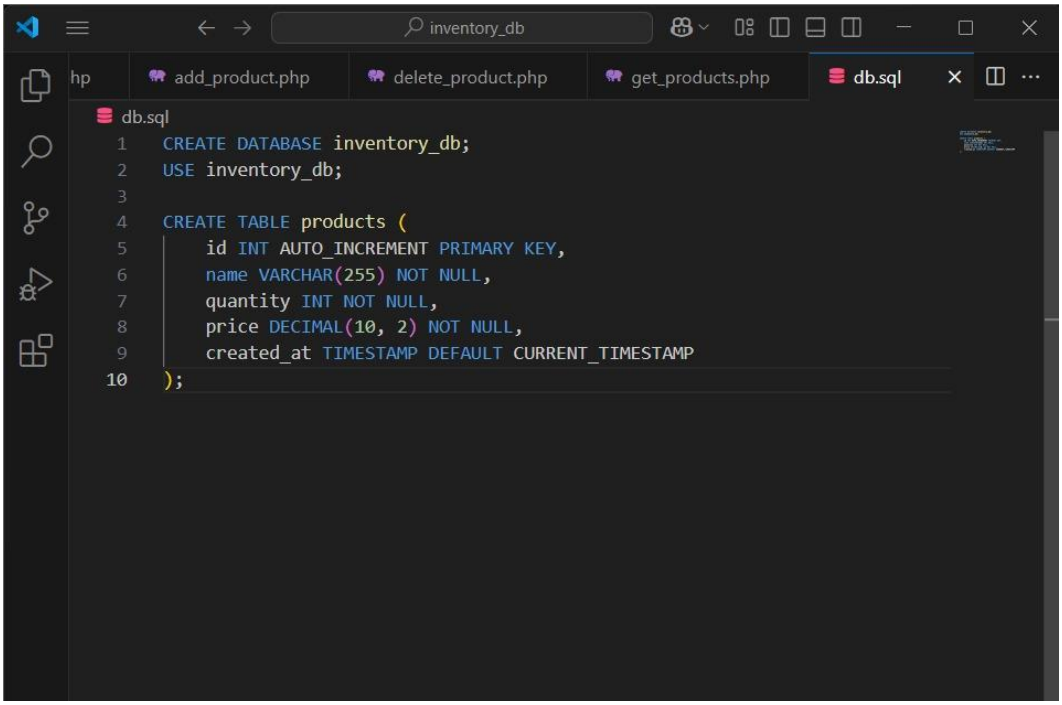

inventory_db
index.html
style.css
script.js
config.php
add_product.php
script.js > deleteProduct
document.addEventListener('DOMContentLoaded', () => {
// Fetch and display products on page load
fetchProducts();
// Handle form submission to add a product
document.getElementById('add-product-form').addEventListener('submit',
e.preventDefault();
const formData = new FormData(e.target);
fetch('add_product.php', {
method: 'POST',
body: formData
})
.then(response => response.text())
.then(data => {
alert(data);
fetchProducts(); // Refresh the product list
e.target.reset(); // Reset the form
})
.catch(error => console.error('Error:', error));
});
});
function fetchProducts() {
fetch('get_products.php')
.then(response => response.json())
.then(products => {
// Update summary
const totalProducts = products.length;
const totalQuantity = products.reduce((sum, product) => sum +
const totalValue = products.reduce((sum, product) => sum + (pr
document.getElementById('total-products').textContent = totalP
document.getElementById('total-quantity').textContent = totalQ
document.getElementById('total-value').textContent = `${total
// Update product table
Ln 66, Col 2
Spaces: 4
UTF-8
CRLF
JavaScript

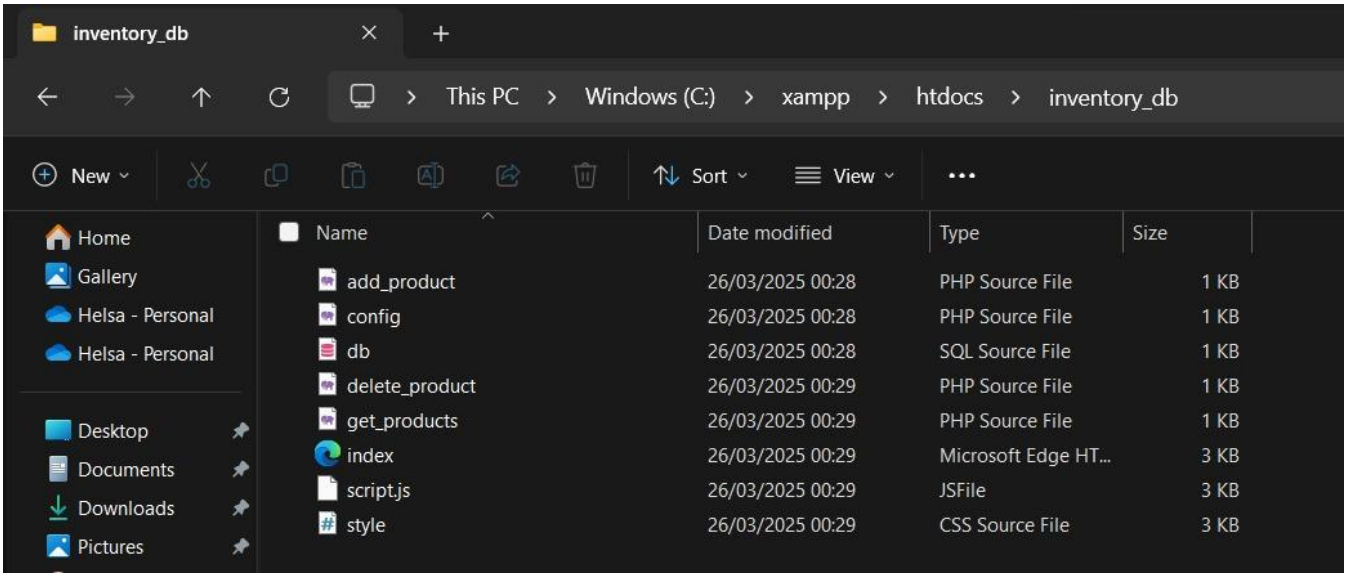

inventory_db
index.html
style.css
script.js
config.php
add_product.php
config.php
<?php
$host = 'localhost';
$dbname = 'inventory_db';
$username = 'root'; // Default XAMPP username
$password = '';   // Default XAMPP password (empty)
try {
$pdo = new PDO("mysql:host=$host;dbname=$dbname", $username, $password
$pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
} catch (PDOException $e) {
echo "Connection failed: " . $e->getMessage();
}
?>


inventory_db
hp
add_product.php
delete_product.php
get_products.php
db.sql
add_product.php
<?php
include 'config.php';
if ($_SERVER['REQUEST_METHOD'] == 'POST') {
$name = $_POST['name'];
$quantity = $_POST['quantity'];
$price = $_POST['price'];
$stmt = $pdo->prepare("INSERT INTO products (name, quantity, price) VA
$stmt->execute([$name, $quantity, $price]);
echo "Product added successfully!";
}
?>


inventory_db
add_product.php
delete_product.php
get_products.php
db.sql
delete_product.php
C:\Users\helsa\OneDrive\Documents\inventory_db\delete_product.php
include 'config.php';
if (isset($_GET['id'])) {
$id = $_GET['id'];
$stmt = $pdo->prepare("DELETE FROM products WHERE id = ?");
$stmt->execute([$id]);
echo "Product deleted successfully!";
}
?>


inventory_db
add_product.php
delete_product.php
get_products.php
db.sql
get_products.php
<?php
include 'config.php';
$stmt = $pdo->query("SELECT * FROM products");
$products = $stmt->fetchAll(PDO::FETCH_ASSOC);
header('Content-Type: application/json');
echo json_encode($products);
?>


inventory_db
add_product.php
delete_product.php
get_products.php
db.sql
db.sql
CREATE DATABASE inventory_db;
USE inventory_db;
CREATE TABLE products (
id INT AUTO_INCREMENT PRIMARY KEY,
name VARCHAR(255) NOT NULL,
quantity INT NOT NULL,
price DECIMAL(10, 2) NOT NULL,
created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);



Step 4 : Buka web lalu cari *localhost/inventory_db/* setelah itu akan langsung muncul tampilan web Inventory Management System yg sudah dibuat, silakan dicoba dan pastikan datanya bertambah juga di phpMyAdmin.