**1.A simple calculator web application that takes two numbers and an operator (+, -, /, \* and %) from an HTML page and returns the result page with the operation performed on the operands.**

**Scal.html**

<!DOCTYPE html>

<html>

<head>

    <title>Simple Calculator</title>

</head>

<body>

    <h1>Simple Calculator</h1>

    <form action="scal.php" method="post">

        <label for="num1">Enter first number:</label><br>

        <input type="number" id="num1" name="num1" required><br><br>

        <label for="num2">Enter second number:</label><br>

        <input type="number" id="num2" name="num2" required><br><br>

        <label for="operator">Select operator:</label><br>

        <select id="operator" name="operator">

            <option value="+">+</option>

            <option value="-">-</option>

            <option value="\*">\*</option>

            <option value="/">/</option>

            <option value="%">%</option>

        </select><br><br>

        <input type="submit" value="Calculate">

    </form>

</body>

</html>

**Scal.php**

<?php

if (isset($\_POST['num1']) && isset($\_POST['num2']) && isset($\_POST['operator'])) {

    $num1 = $\_POST['num1'];

    $num2 = $\_POST['num2'];

    $operator = $\_POST['operator'];

    switch ($operator) {

        case '+':

            $result = $num1 + $num2;

            break;

        case '-':

            $result = $num1 - $num2;

            break;

        case '\*':

            $result = $num1 \* $num2;

            break;

        case '/':

            if ($num2 != 0) {

                $result = $num1 / $num2;

            } else {

                $result = "Error: Division by zero";

            }

            break;

        case '%':

            if ($num2 != 0) {

                $result = $num1 % $num2;

            } else {

                $result = "Error: Modulo by zero";

            }

            break;

        default:

            $result = "Invalid operator";

    }

    echo "<h1>Result:</h1>";

    echo "<p>" . $num1 . " " . $operator . " " . $num2 . " = " . $result . "</p>";

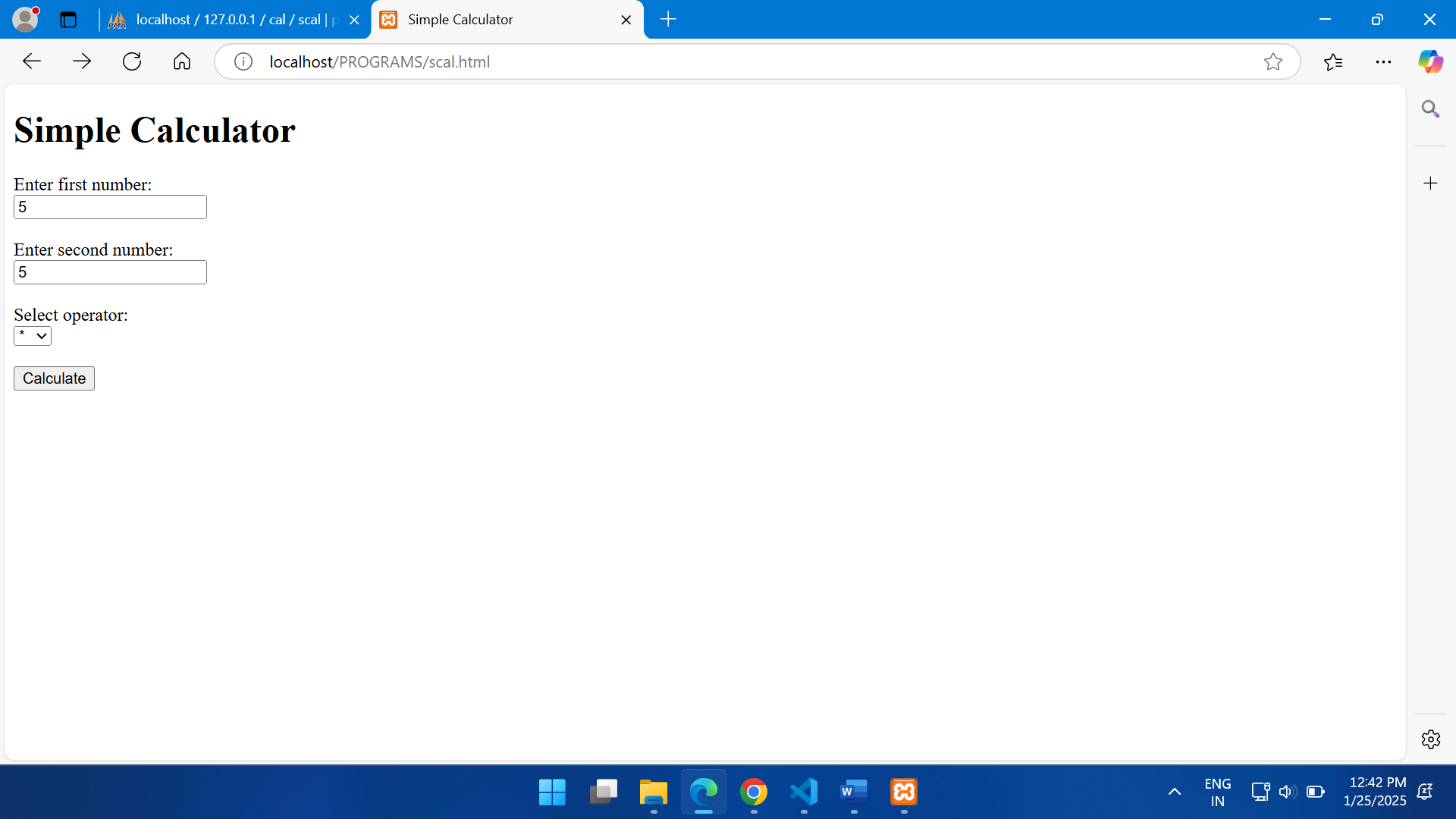
} else {

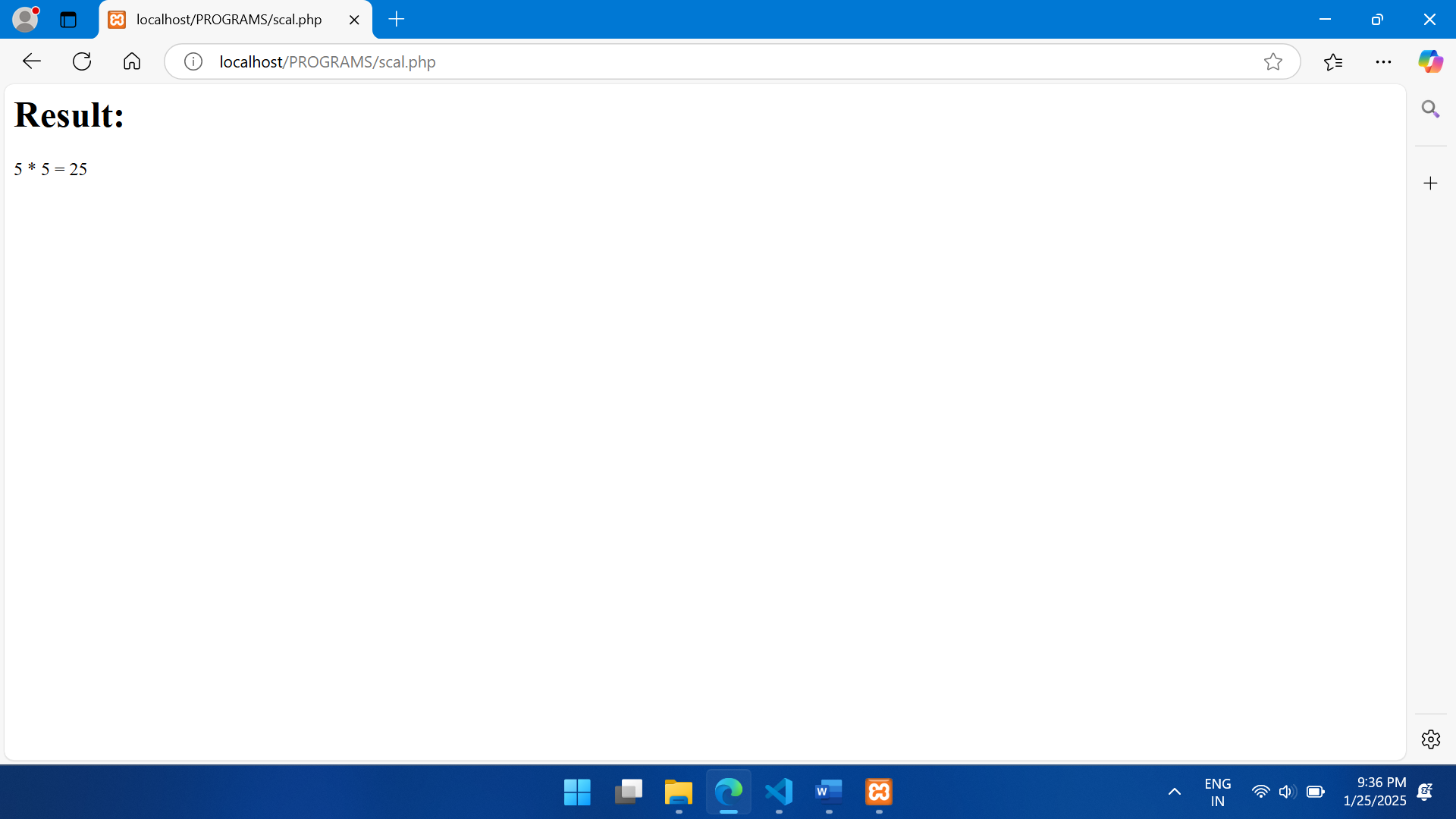
    echo "<h1>Error:</h1>";

    echo "<p>Please enter valid numbers and select an operator.</p>";

}

?>





**2.** **Modify the above program such that it stores each query in a database and checks the database first for the result. If the query is already available in the DB, it returns the value that was previously computed (from DB) or it computes the result and returns it after storing the new query and result in DB.**

**scaldb.php**

<?php

// Database credentials

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "calculator";

// Create connection

$conn = new mysqli($servername, $username, $password, $dbname);

// Check connection

if ($conn->connect\_error) {

    die("Connection failed: " . $conn->connect\_error);

}

// Get input from the form

$num1 = $\_POST['num1'];

$num2 = $\_POST['num2'];

$operator = $\_POST['operator'];

// Check if the query exists in the database

$sql = "SELECT result FROM numbers WHERE num1 = '$num1' AND num2 = '$num2' AND operator = '$operator'";

$result = $conn->query($sql);

if ($result->num\_rows > 0) {

    // Query exists, retrieve result from database

    $row = $result->fetch\_assoc();

    $result = $row['result'];

} else {

    // Query doesn't exist, calculate result

    switch ($operator) {

        case '+':

            $result = $num1 + $num2;

            break;

        case '-':

            $result = $num1 - $num2;

            break;

        case '\*':

            $result = $num1 \* $num2;

            break;

        case '/':

            if ($num2 == 0) {

                $result = "Division by zero is not allowed";

            } else {

                $result = $num1 / $num2;

            }

            break;

        case '%':

            if ($num2 == 0) {

                $result = "Modulo by zero is not allowed";

            } else {

                $result = $num1 % $num2;

            }

            break;

        default:

            $result = "Invalid operator";

    }

    // Store the query and result in the database

    $sql = "INSERT INTO numbers(num1, num2, operator, result) VALUES ('$num1', '$num2', '$operator', '$result')";

    if ($conn->query($sql) === TRUE) {

        echo "New record created successfully";

    } else {

        echo "Error: " . $sql . "<br>" . $conn->error;

    }

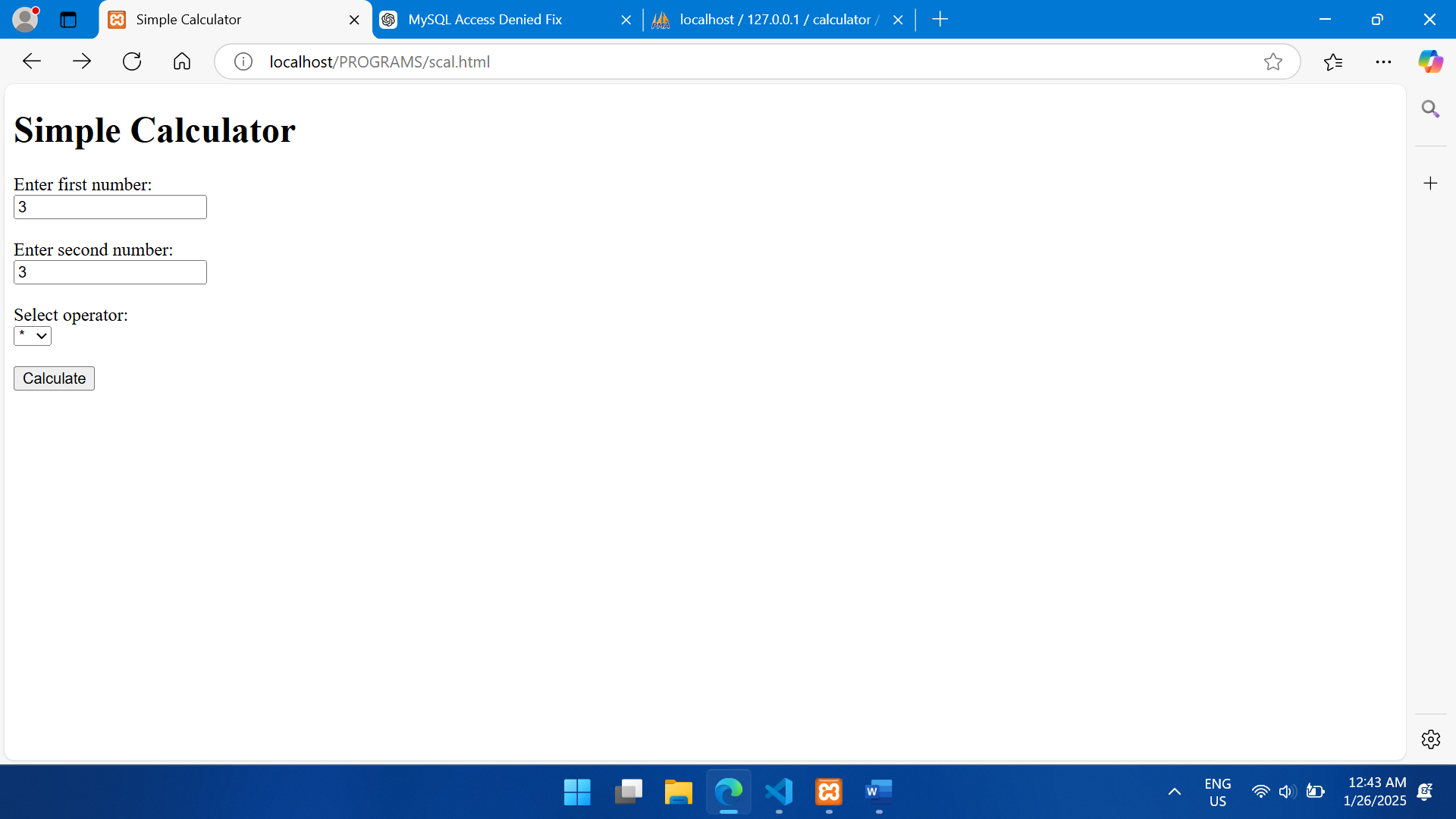
}

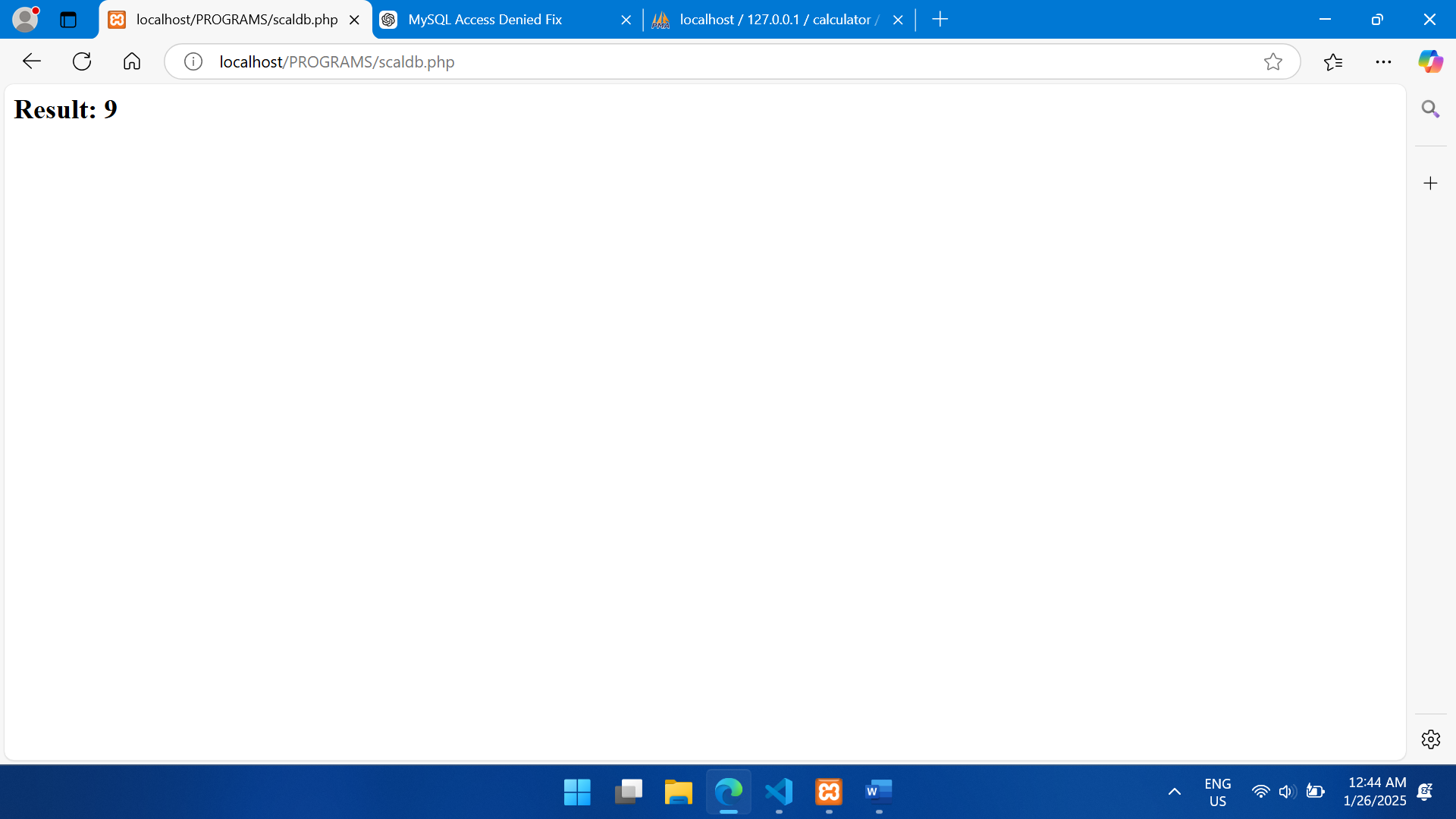
// Display the result

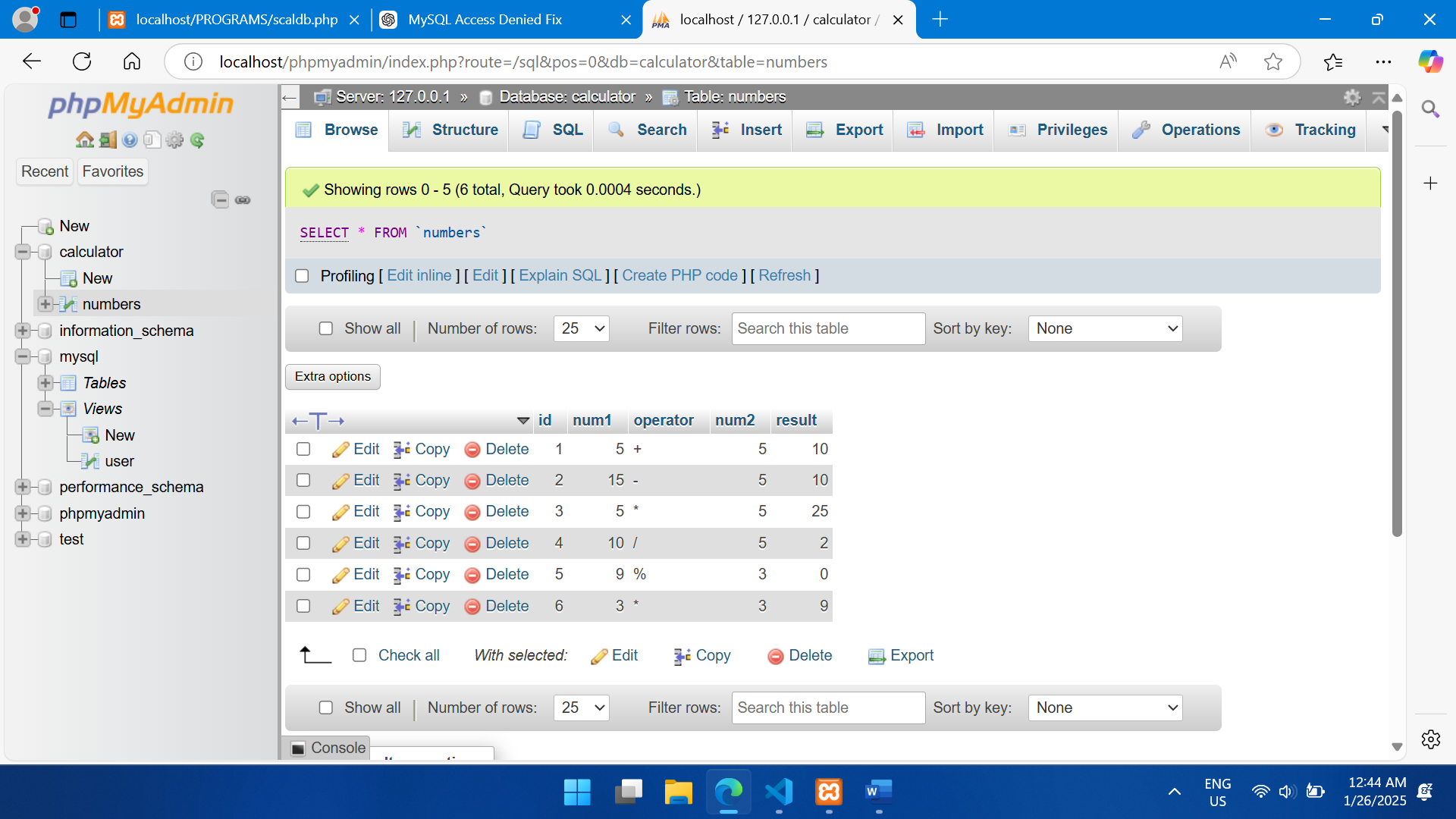
echo "<h2>Result: " . $result . "</h2>";

$conn->close();//

?>







3. A web application takes a name as input and on submit it shows a hello page where is taken from the request. It shows the start time at the right top corner of the page and provides a logout button. On clicking this button, it should show a logout page with Thank You message with the duration of usage (hint: Use session to store name and time).

**Index.html:**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Login</title>

</head>

<body>

    <form method="post" action="hello.php">

        <label for="name">Enter your name:</label>

        <input type="text" id="name" name="name" required>

        <button type="submit">Submit</button>

    </form>

</body>

</html>

**Index.php**

<?php

session\_start();

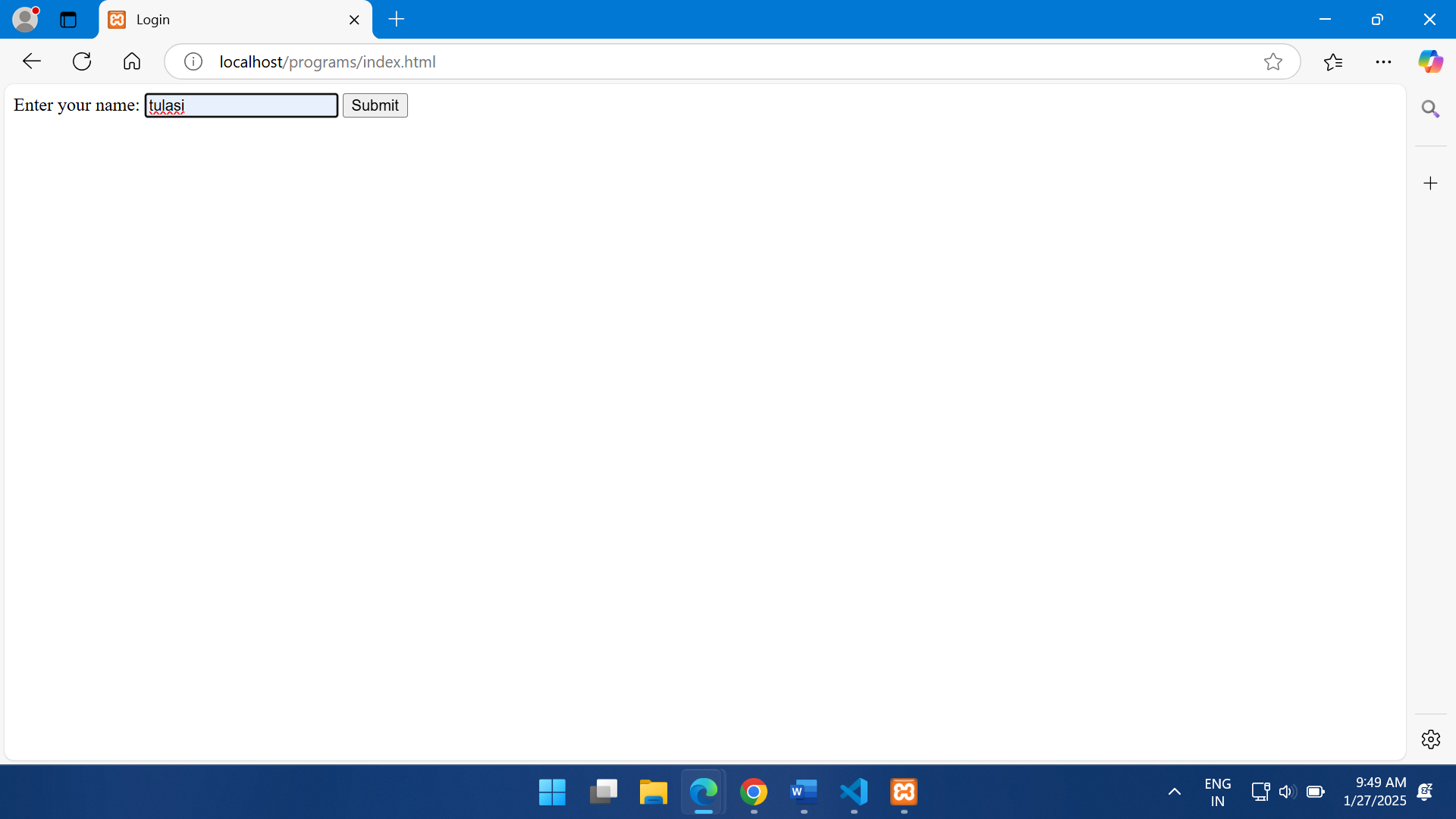
if (isset($\_SESSION['name'])) {

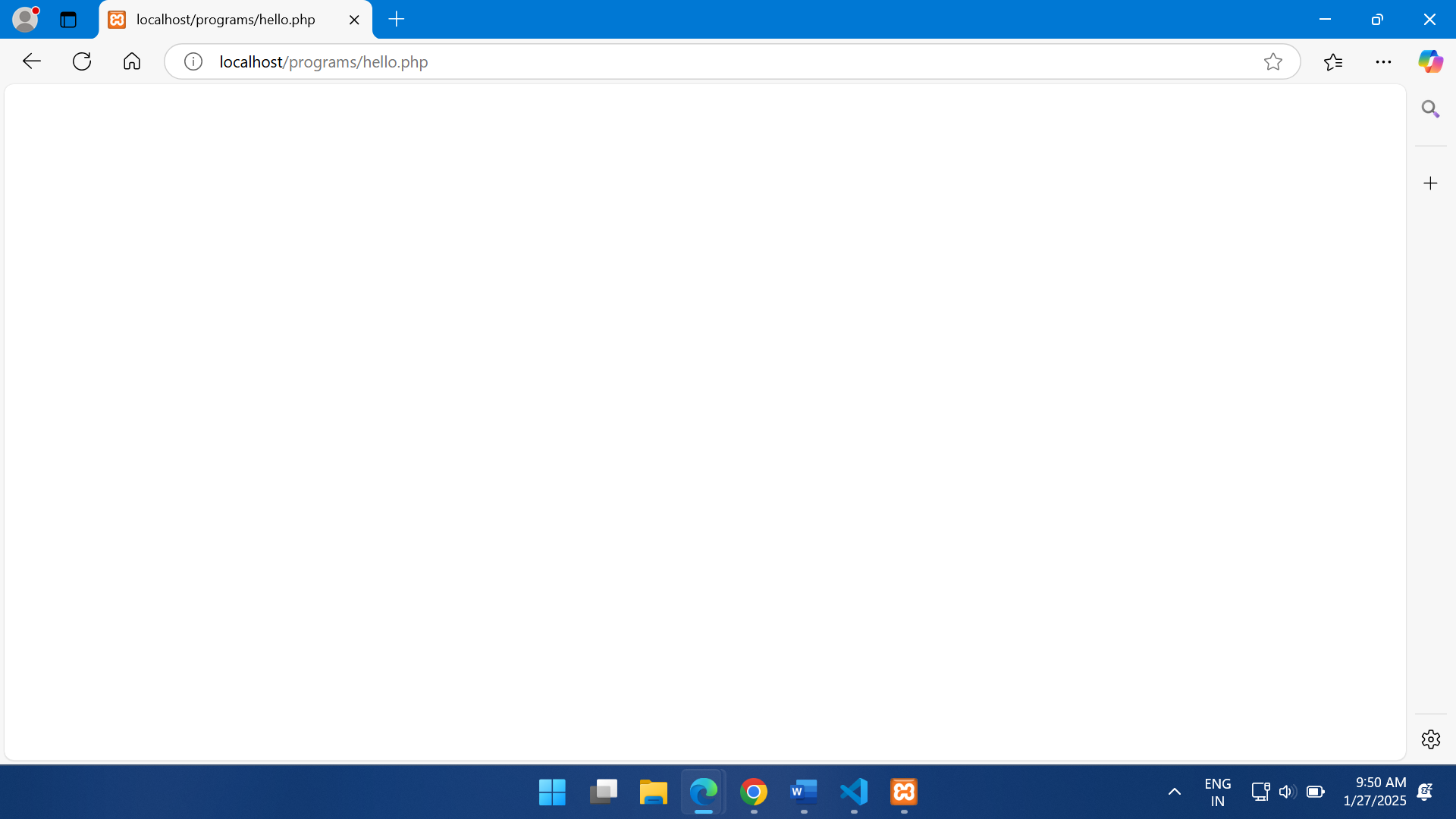
    header("Location: hello.php");

    exit();

}

?>





**Logout.html**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Logout</title>

</head>

<body>

    <h1>Thank You, <?php echo htmlspecialchars($name); ?>!</h1>

    <p>You used this application for <?php echo gmdate("H:i:s", $duration); ?> (hh:mm:ss).</p>

    <a href="index.php">Go to Login Page</a>

</body>

</html>

**Logout.php**

<?php

session\_start();

if (!isset($\_SESSION['name'])) {

    header("Location: index.php");

    exit();

}

$name = $\_SESSION['name'];

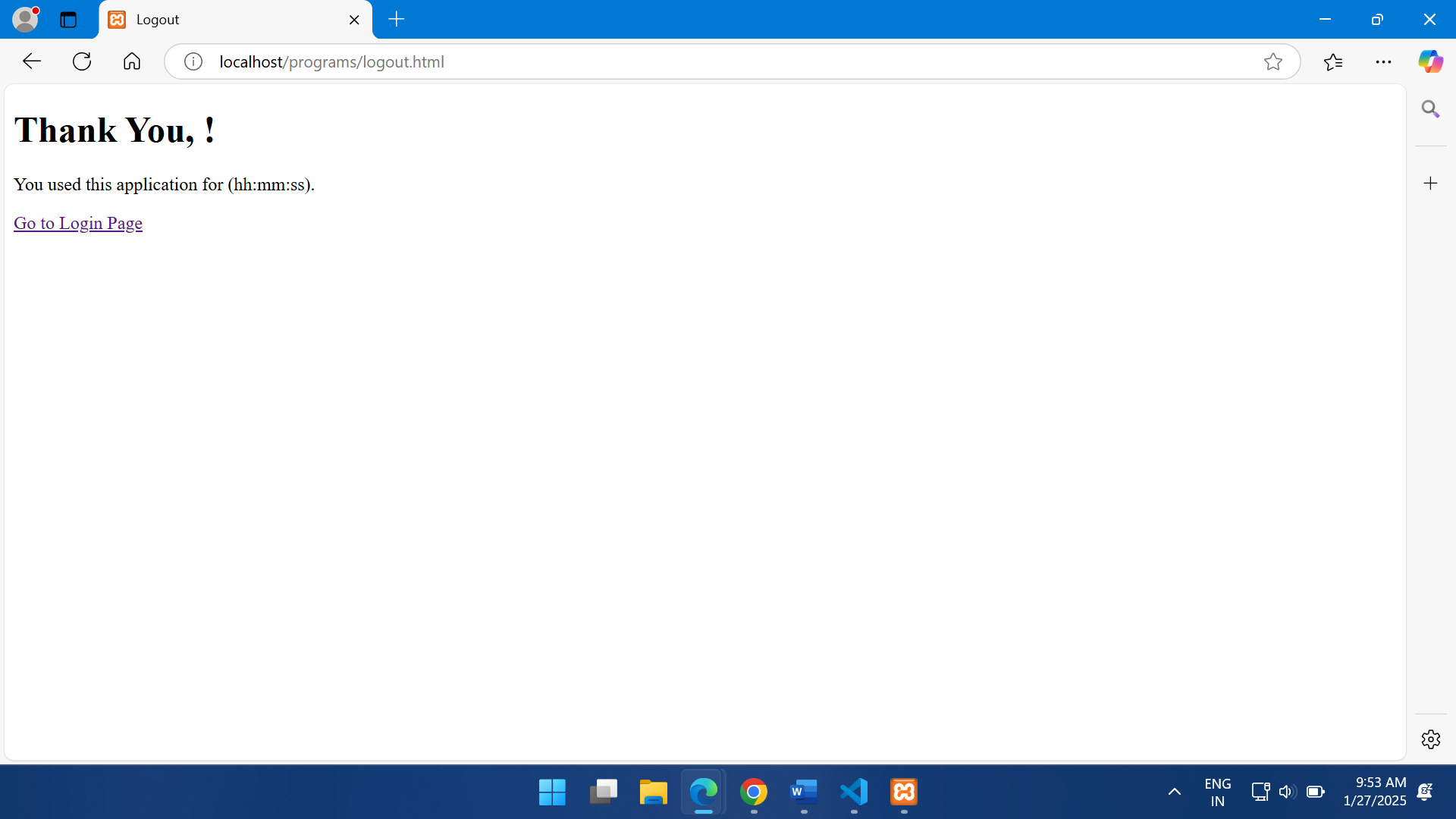
$startTime = $\_SESSION['start\_time'];

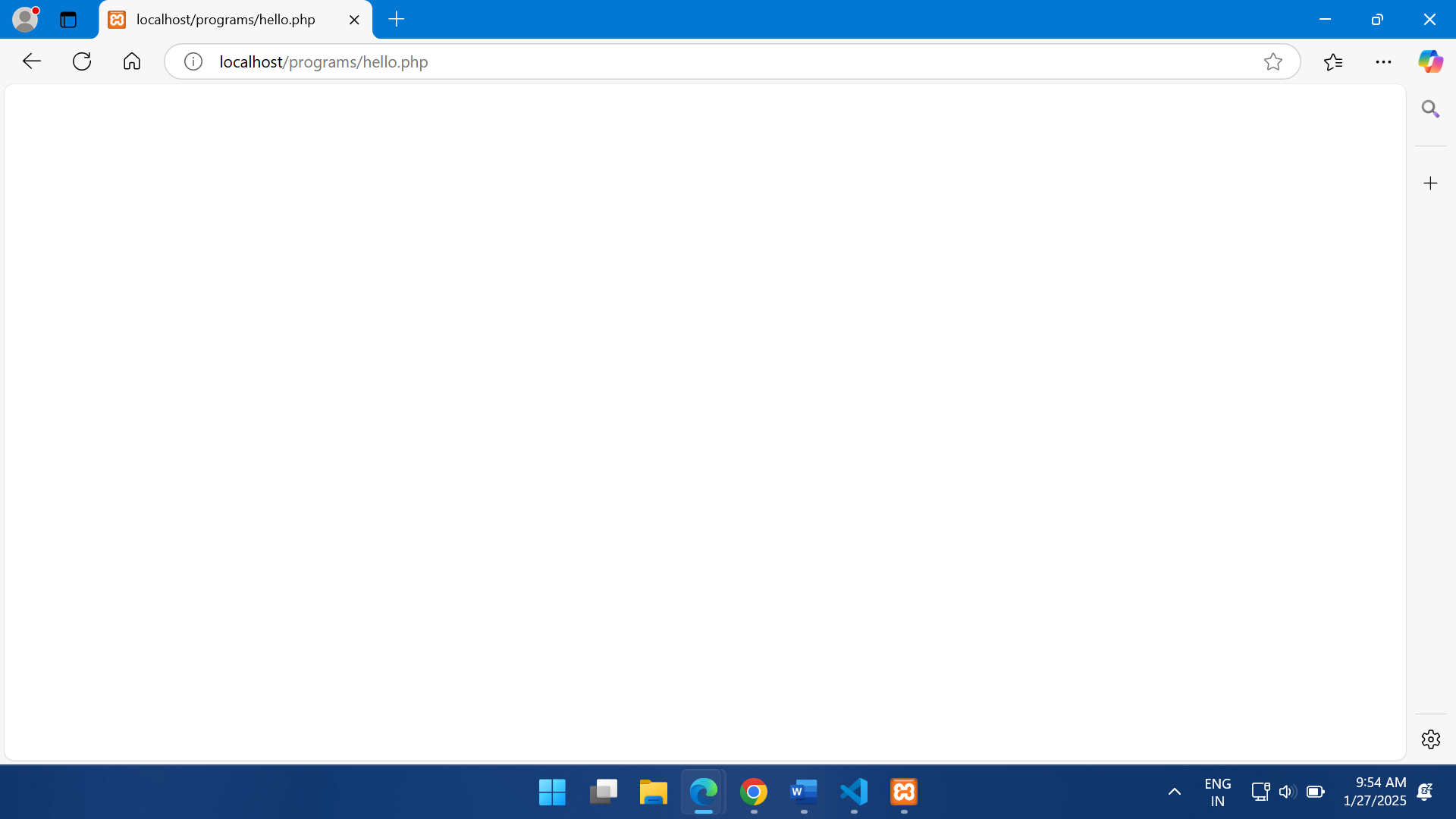
$endTime = time();

$duration = $endTime - $startTime;

session\_destroy();

?>





4. A web application that takes name and age from an HTML page. If the age is less than it should send a page with “Hello, you are not authorized to visit this site”message, where should bereplaced with the entered name. Otherwise it should send “Welcome to this site”message.

**Pro4.html**

<!DOCTYPE html>

<html>

<head>

<title>Age Authorization</title>

</head>

<body>

    <h1>Enter Your Details</h1>

    <form action="pro4.php" method="post">

        <label for="name">Name:</label>

        <input type="text" id="name" name="name" required>

        <br><br>

        <label for="age">Age:</label>

        <input type="number" id="age" name="age" required>

        <br><br>

        <button type="submit">Submit</button>

    </form>

</body>

</html>

**Pro4.php**

<?php

if ($\_SERVER['REQUEST\_METHOD'] == 'POST') {

    // Retrieve form data

    $name = htmlspecialchars($\_POST['name']);

    $age = intval($\_POST['age']);

    // Check age

    if ($age < 18) {

        echo "<h1>Hello $name, you are not authorized to visit this site.</h1>";

    } else {

        echo "<h1>Welcome to this site, $name!</h1>";

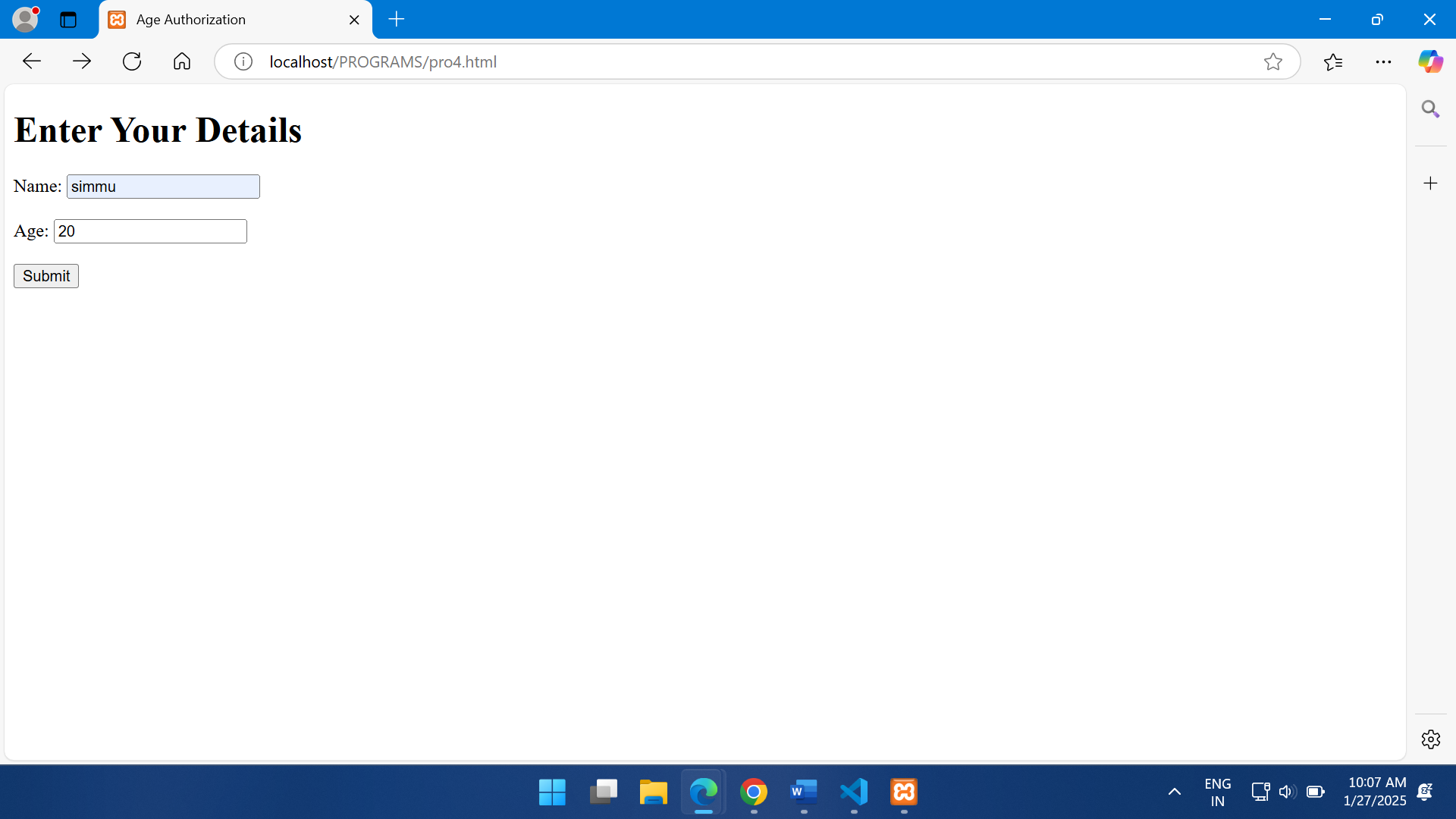
    }

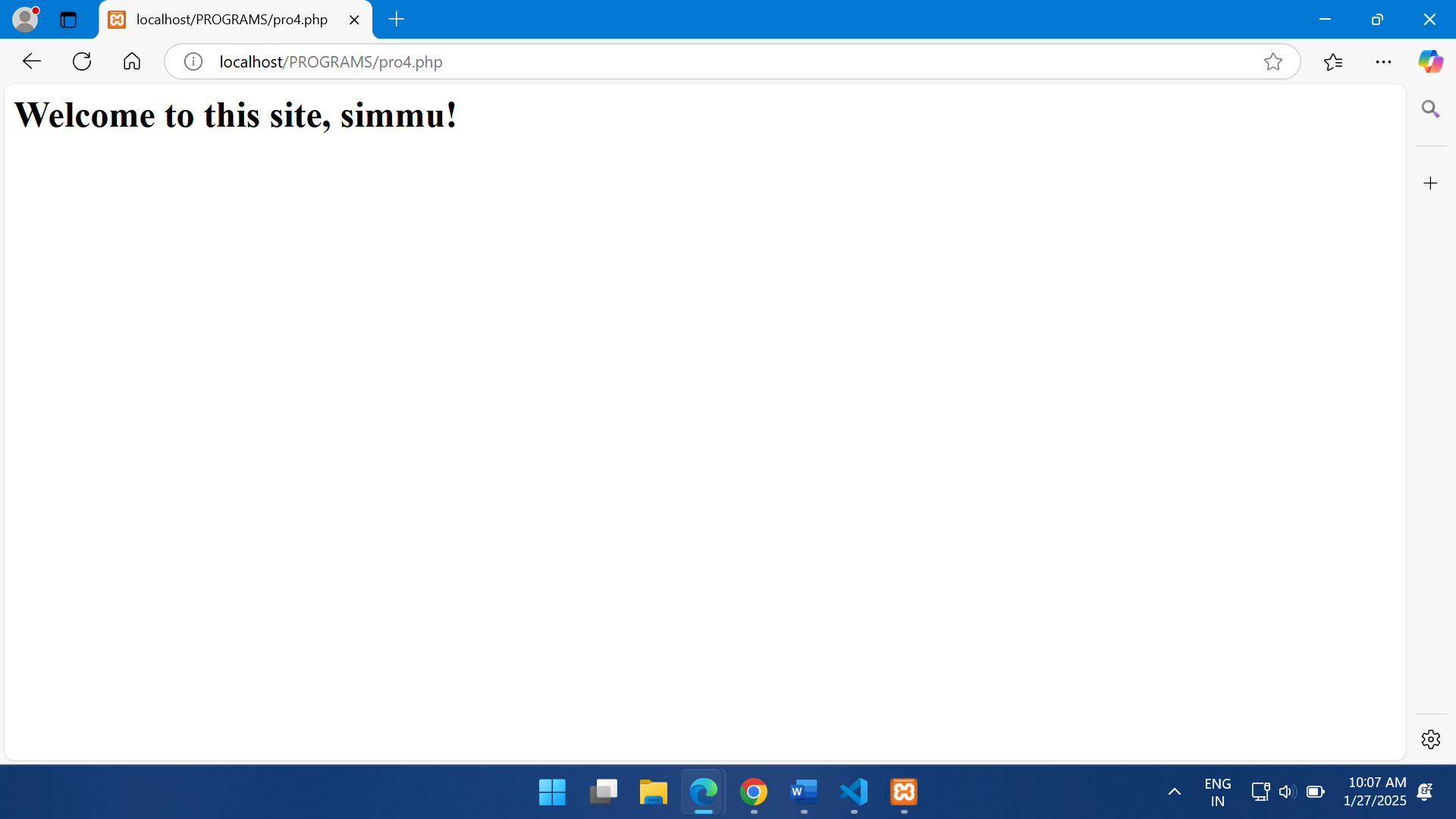
} else {

    echo "<h1>Invalid request method.</h1>";

}

?>





**5.** **A web application for implementation: The user is first served a login page which takes users name and password. After submitting the details, the server checks these values against the data from a database and takes the following decisions. If name and password matches, serves a welcome page with users full name. If name matches and password does not match, then serves “password mismatch” page If name is not found in the database, serves a registration page,** **where user’s full name is asked and on submitting the full name, it stores, the login name, password and full name in the database (hint: use session for storing the submitted login name and password).**

**CREATE TABLE users ( id INT AUTO\_INCREMENT PRIMARY KEY, username VARCHAR(255) UNIQUE NOT NULL, password VARCHAR(255) NOT NULL, full\_name VARCHAR(255) NOT NULL );**

**Login.html**

<?php

session\_start();

?>

<!DOCTYPE html>

<html>

<head>

<title>Login</title>

</head>

<body>

<form action="process\_login.php" method="POST">

<label for="username">Username:</label>

<input type="text" id="username" name="username" required>

<br>

<label for="password">Password:</label>

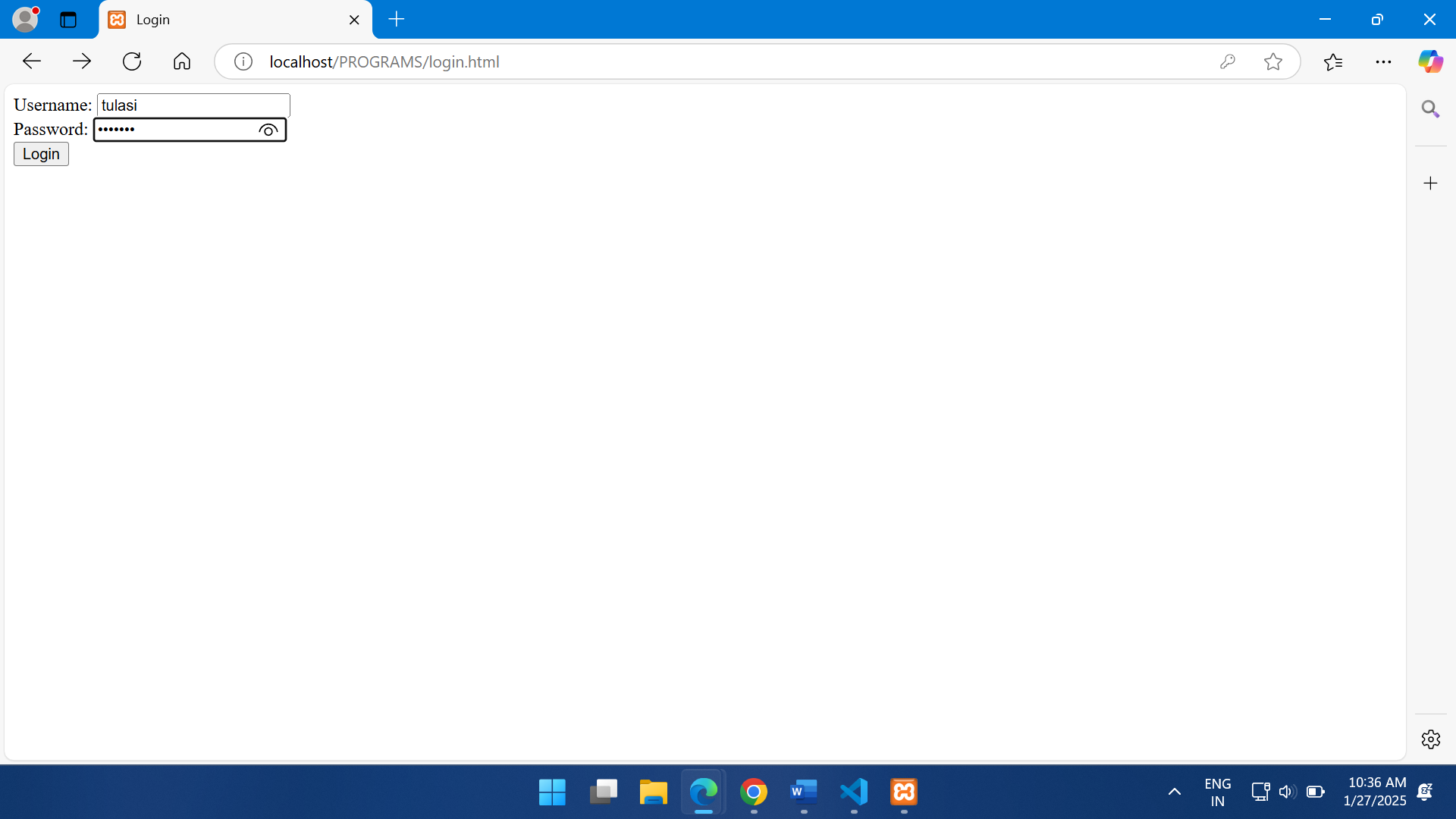
<input type="password" id="password" name="password" required>

<br>

<button type="submit">Login</button>

</form>

</body>

</html>

**pro5process\_login.php**

<?php

session\_start();

$conn = new mysqli('localhost', 'root', '', 'your\_database\_name');

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

$username = $\_POST['username'];

$password = $\_POST['password'];

$sql = "SELECT \* FROM users WHERE username = ?";

$stmt = $conn->prepare($sql);

$stmt->bind\_param('s', $username);

$stmt->execute();

$result = $stmt->get\_result();

$user = $result->fetch\_assoc();

if ($user) {

if (password\_verify($password, $user['password'])) {

$\_SESSION['full\_name'] = $user['full\_name'];

header("Location: welcome.php");

} else {

echo "Password mismatch. <a href='login.php'>Go back</a>";

}

} else {

$\_SESSION['username'] = $username;

$\_SESSION['password'] = password\_hash($password, PASSWORD\_DEFAULT);

header("Location: register.php");

}

?>

**Pro5register.php**

<?php

session\_start();

if ($\_SERVER['REQUEST\_METHOD'] == 'POST') {

$conn = new mysqli('localhost', 'root', '', 'your\_database\_name');

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

$username = $\_SESSION['username'];

$password = $\_SESSION['password'];

$full\_name = $\_POST['full\_name'];

$sql = "INSERT INTO users (username, password, full\_name) VALUES (?, ?, ?)";

$stmt = $conn->prepare($sql);

$stmt->bind\_param('sss', $username, $password, $full\_name);

if ($stmt->execute()) {

echo "Registration successful. <a href='login.php'>Login now</a>";

} else {

echo "Error: " . $stmt->error;

}

$stmt->close();

$conn->close();

session\_destroy();

} else {

?>

<!DOCTYPE html>

<html>

<head>

<title>Register</title>

</head>

<body>

<form action="register.php" method="POST">

<label for="full\_name">Full Name:</label>

<input type="text" id="full\_name" name="full\_name" required>

<br>

<button type="submit">Register</button>

</form>

</body>

</html>

<?php

}

?>

**Welcome.php**

<?php

session\_start();

if (isset($\_SESSION['full\_name'])) {

echo "<h1>Welcome, " . htmlspecialchars($\_SESSION['full\_name']) . "!</h1>";

echo "<a href='logout.php'>Logout</a>";

} else {

header("Location: login.php");

}

?>

**Logout.php**

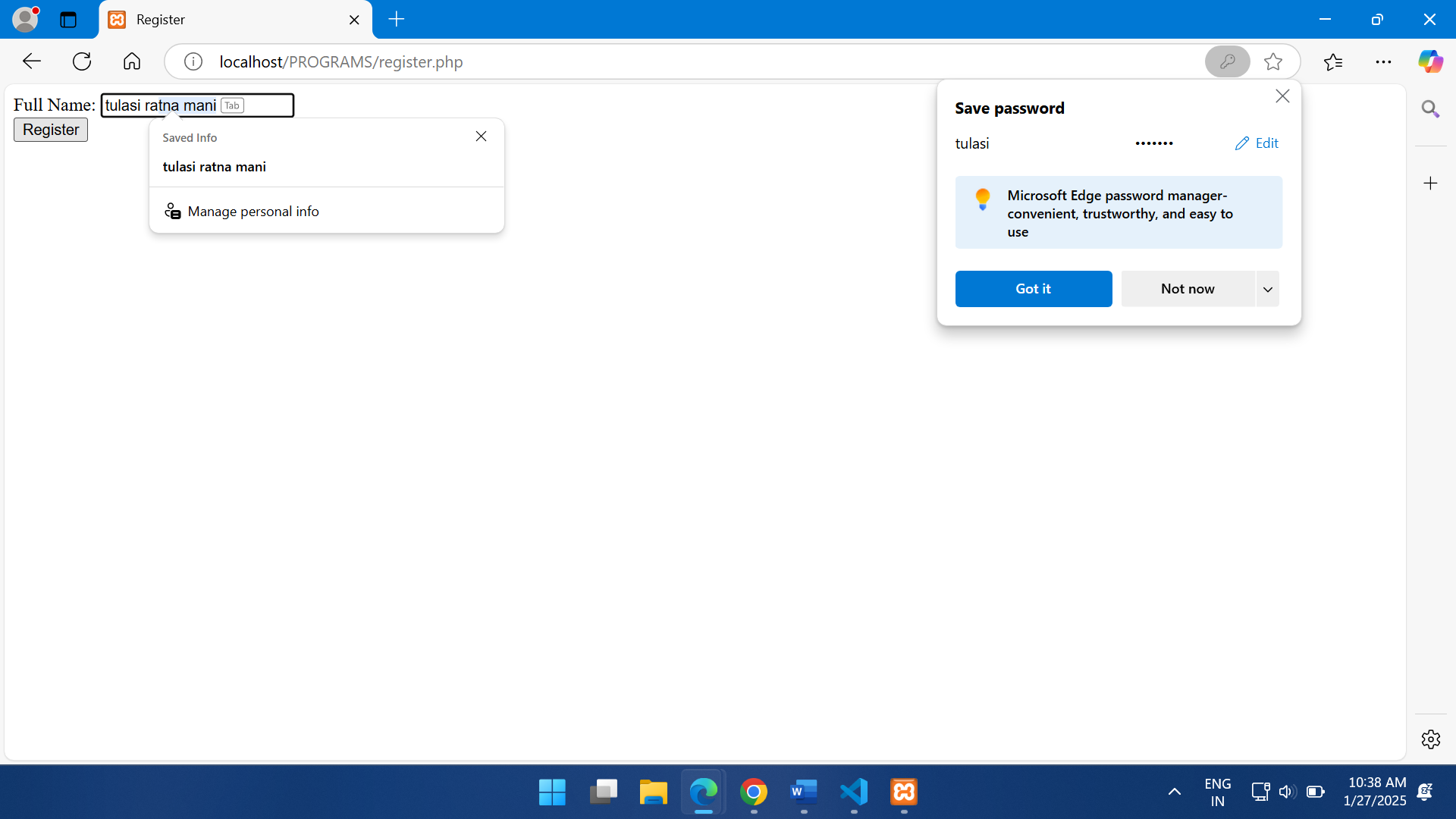
<?php

session\_start();

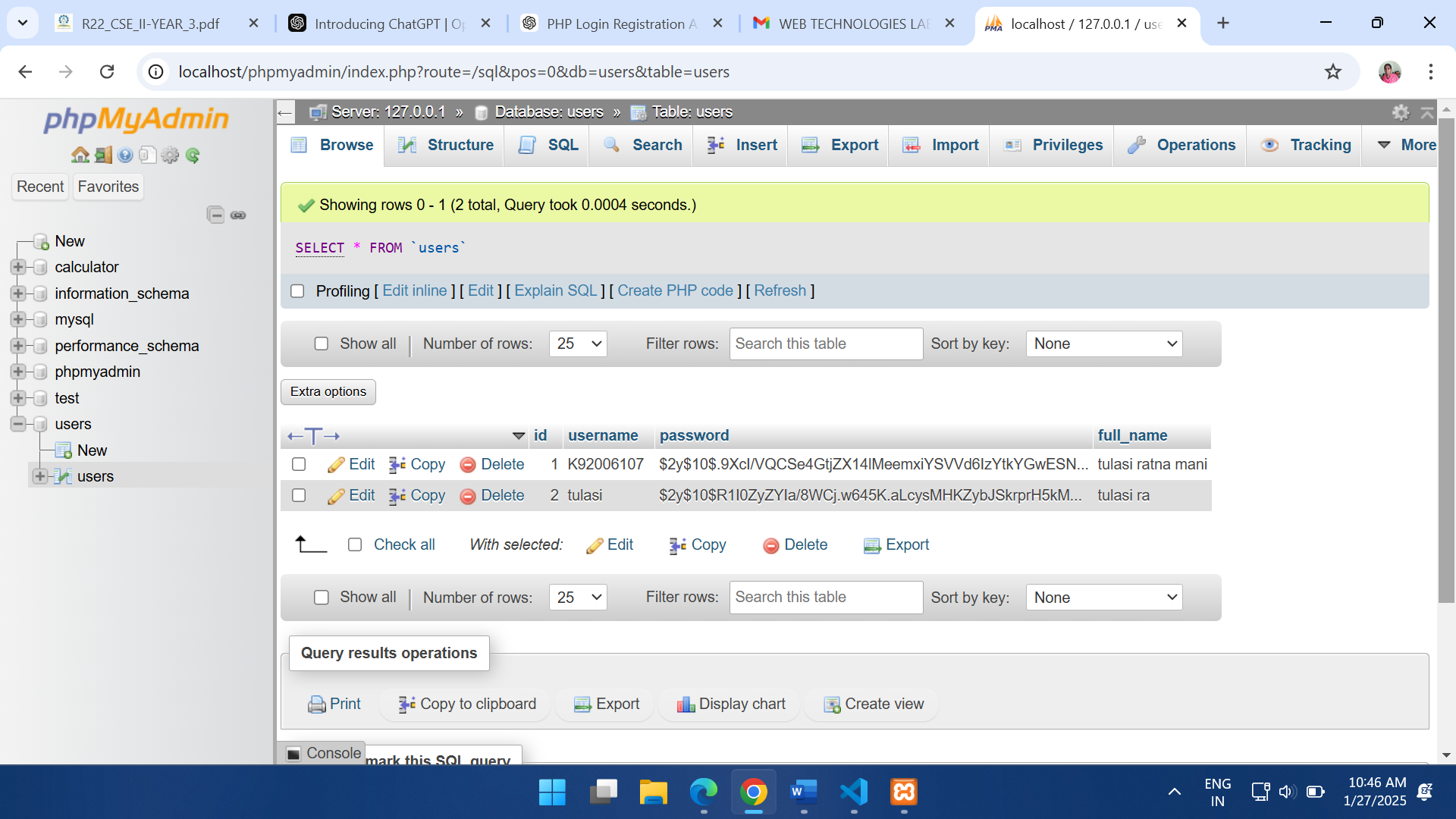
session\_destroy();

header("Location: login.php");

?>







**6.** **A web application that lists all cookies stored in the browser on clicking “List cookies button”Add cookies if necessary.**

**cookies\_app.php**

<?php

// Handle adding cookies

if ($\_SERVER['REQUEST\_METHOD'] === 'POST' && isset($\_POST['cookie\_name'], $\_POST['cookie\_value'])) {

    $cookie\_name = preg\_replace('/[=,; \t\r\n\013\014]/', '\_', $\_POST['cookie\_name']); // Replace invalid characters with '\_'

    $cookie\_value = htmlspecialchars($\_POST['cookie\_value'], ENT\_QUOTES, 'UTF-8'); // Sanitize value for safety

    if ($cookie\_name) {

        // Set cookie for 1 day

        setcookie($cookie\_name, $cookie\_value, time() + 86400, "/");

        header("Location: cookies\_app.php"); // Redirect to avoid form resubmission

        exit;

    } else {

        echo "Invalid cookie name. Please use a valid name.";

    }

}

?>

<!DOCTYPE html>

<html>

<head>

    <title>Cookie Manager</title>

</head>

<body>

    <h1>Cookie Manager</h1>

    <!-- Form to Add a New Cookie -->

    <form method="POST" action="cookies\_app.php">

        <label for="cookie\_name">Cookie Name:</label>

        <input type="text" id="cookie\_name" name="cookie\_name" required>

        <br>

        <label for="cookie\_value">Cookie Value:</label>

        <input type="text" id="cookie\_value" name="cookie\_value" required>

        <br>

        <button type="submit">Add Cookie</button>

    </form>

    <br>

    <!-- Button to List Cookies -->

    <button onclick="location.href='?list\_cookies=true'">List Cookies</button>

    <h2>Stored Cookies:</h2>

    <ul>

        <?php

        // List cookies if the button is clicked

        if (isset($\_GET['list\_cookies'])) {

            if (!empty($\_COOKIE)) {

                foreach ($\_COOKIE as $key => $value) {

                    echo "<li><strong>" . htmlspecialchars($key) . ":</strong> " . htmlspecialchars($value) . "</li>";

                }

            } else {

                echo "<li>No cookies found.</li>";

            }

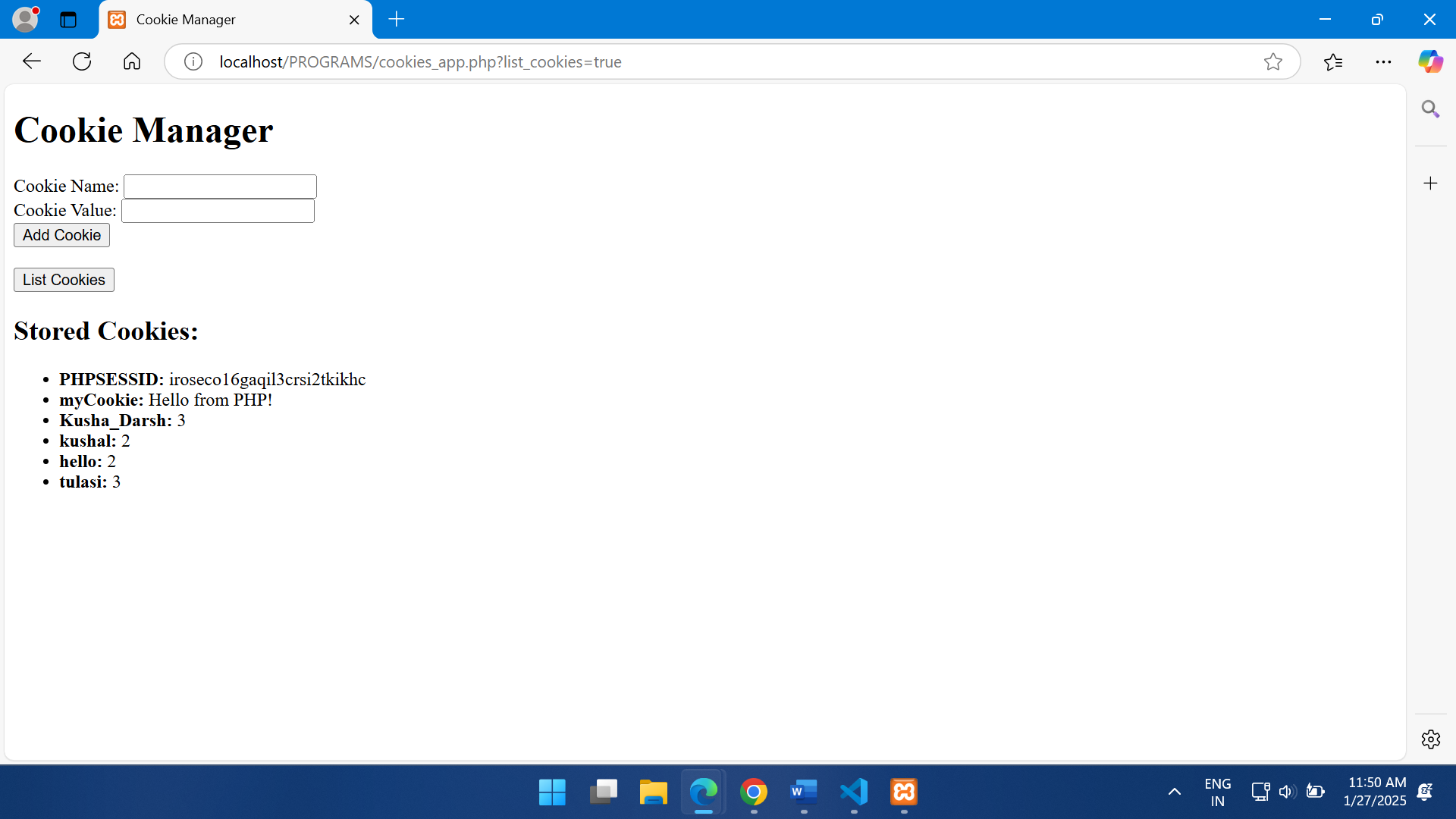
        }

        ?>

    </ul>

</body>

</html>



**7. A user validation web application, where the user submits the login name and password to the server. The name and password are checked against the data already available in Database and if the data matches, a successful login page is returned. Otherwise a failure message is shown to the user.**

**Users.xml**

<?xml version="1.0"?>

<users>

<user>

<userid>madhu</userid>

<password>12345</password>

</user>

<user>

<userid>naveen</userid>

<password>12345</password>

</user>

<user>

<userid>durga</userid>

<password>12345</password>

</user>

<user>

<userid>ramesh</userid>

<password>12345</password>

</user>

</users>

**Loginxml.php**

<html>

<head>

<title>Login </title>

</head>

<body>

<form name="form1" method="post">

<table align="center" width="80%" border="0" cellpadding="3" cellspacing="1">

<tr >

<td colspan=2 align="center">

<b><h3>Member Login</h3> </b></td>

</tr>

<br/>

<tr>

<td>Username  : </td>

<td><input name="uname" type="text"></td>

</tr>

<tr>

<td>Password  : </td>

<td><input name="pwd" type="password"></td>

</tr>

<tr>

<td></td>

<td ><input type="submit" name="Submit" value="Login"></td>

</tr>

</table>

</form>

<?php

if(isset($\_POST["Submit"]))

{

$uname=$\_POST["uname"];

$pwd=$\_POST["pwd"];

$xml=simplexml\_load\_file("users.xml") or die("Error: Cannot create object");

foreach($xml->children() as $users)

{

    if($users->userid==$uname)

    {

        if($users->password==$pwd)

        {

            echo "<center><b>Hello..".$uname . ", ";

            echo "Login Successful</b></center> ";

            return;

        }

        else

        {

            echo "<center><b>Invalid Password...</b></center>";

            return;

        }

    }

}

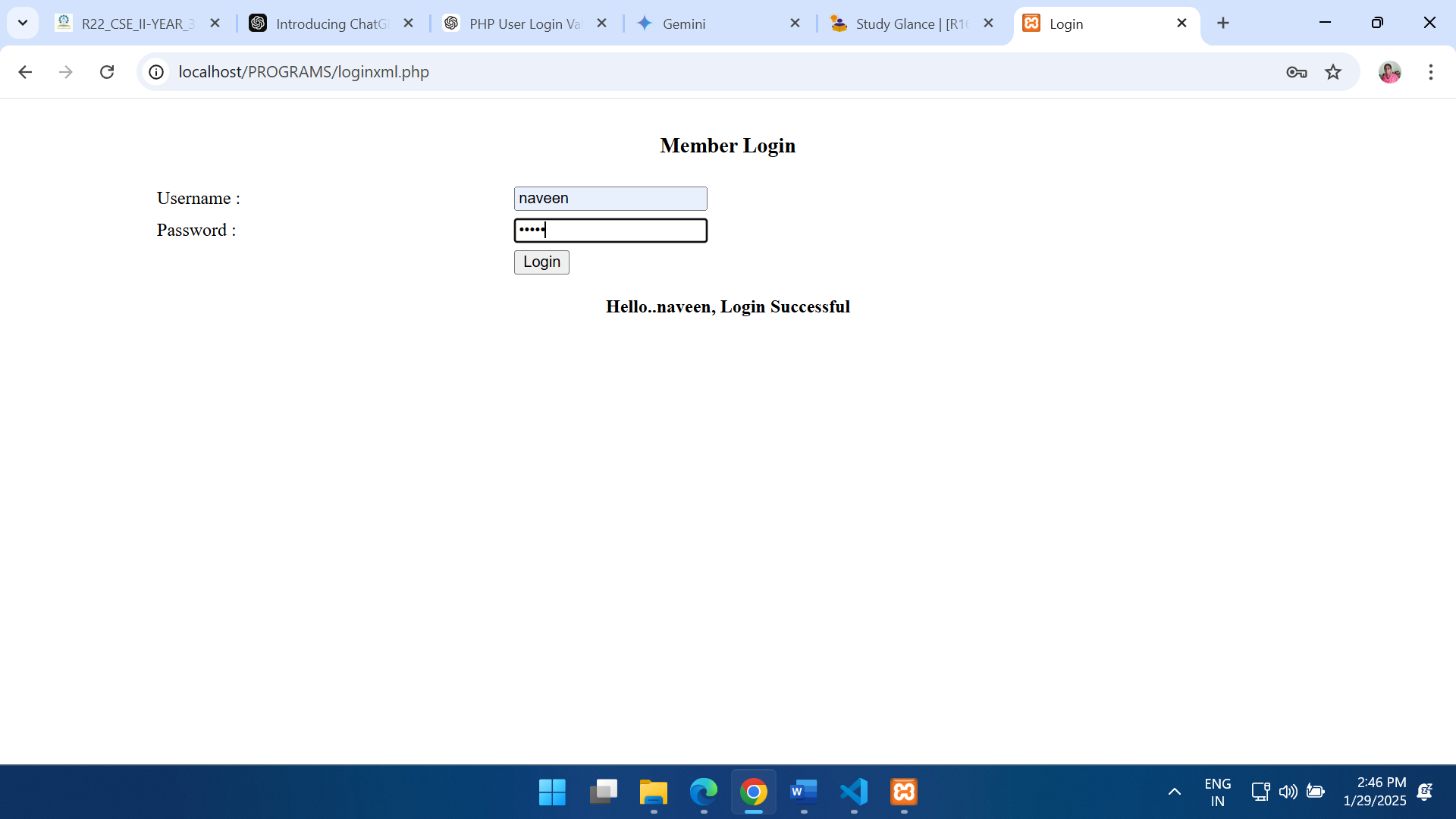
        echo "<center><b>Invalid Login..</b></center>";

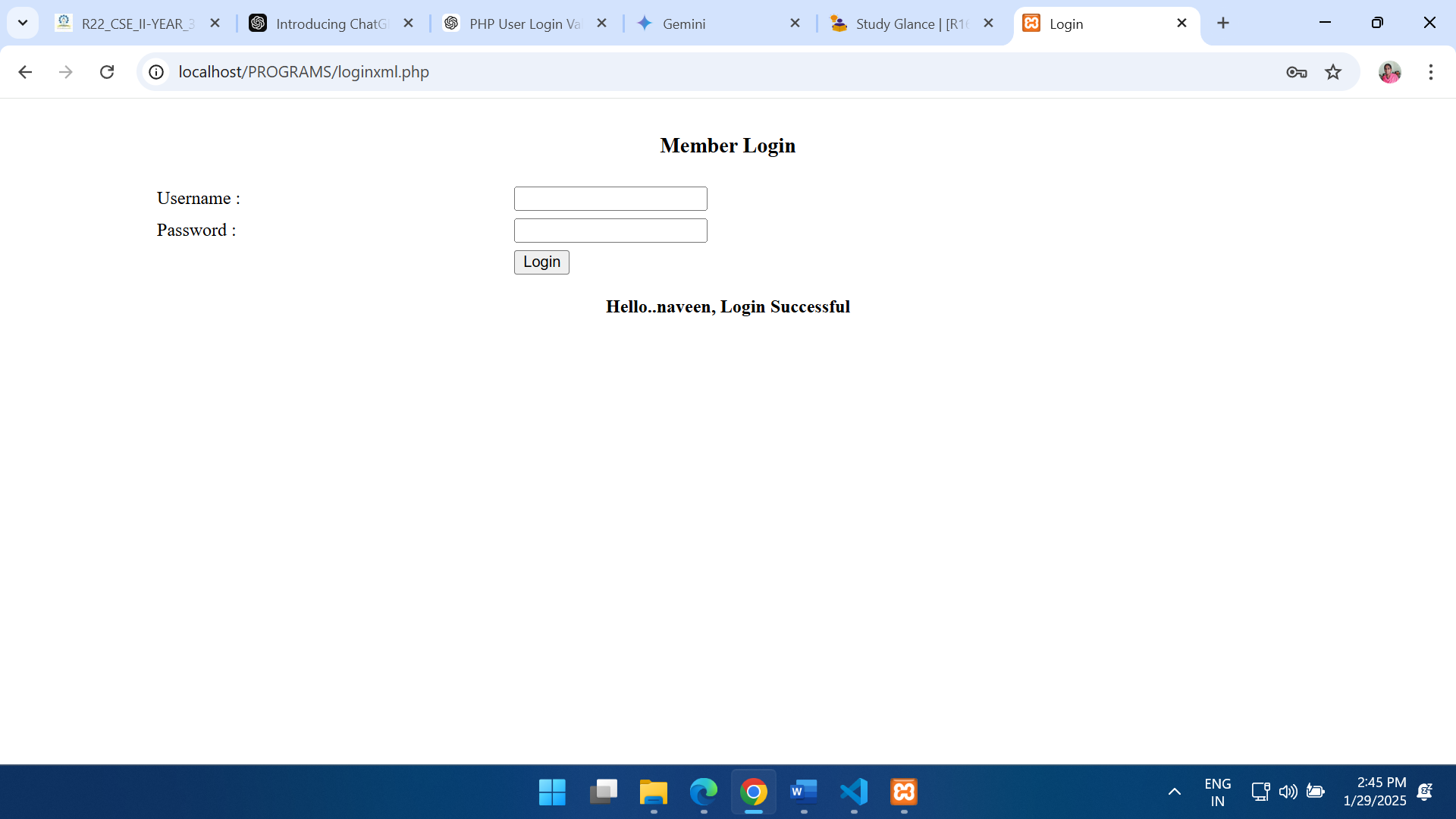
}

?>

</body>

</html>





**8.** **Modify the 7th program to use an XML file instead of database.**

**User.xml**

<?xml version="1.0" encoding="UTF-8"?>

<users>

    <user>

        <username>admin</username>

        <password>admin123</password>

    </user>

    <user>

        <username>user1</username>

        <password>password1</password>

    </user>

    <user>

        <username>user2</username>

        <password>password2</password>

    </user>

</users>

**Inde.html**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Login</title>

</head>

<body>

    <h2>Login</h2>

    <form action="log.php" method="POST">

        <label for="username">Username:</label>

        <input type="text" id="username" name="username" required>

        <br><br>

        <label for="password">Password:</label>

        <input type="password" id="password" name="password" required>

        <br><br>

        <button type="submit">Login</button>

    </form>

</body>

</html>

**Log.php**

<?php

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

    $username = $\_POST['username'];

    $password = $\_POST['password'];

    // Load XML file

    $xml = simplexml\_load\_file("user.xml") or die("Error: Cannot load XML file.");

    $isValid = false;

    // Check credentials

    foreach ($xml->user as $user) {

        if ($user->username == $username && $user->password == $password) {

            $isValid = true;

            break;

        }

    }

    if ($isValid) {

        echo "<h2>Login Successful! Welcome, " . htmlspecialchars($username) . ".</h2>";

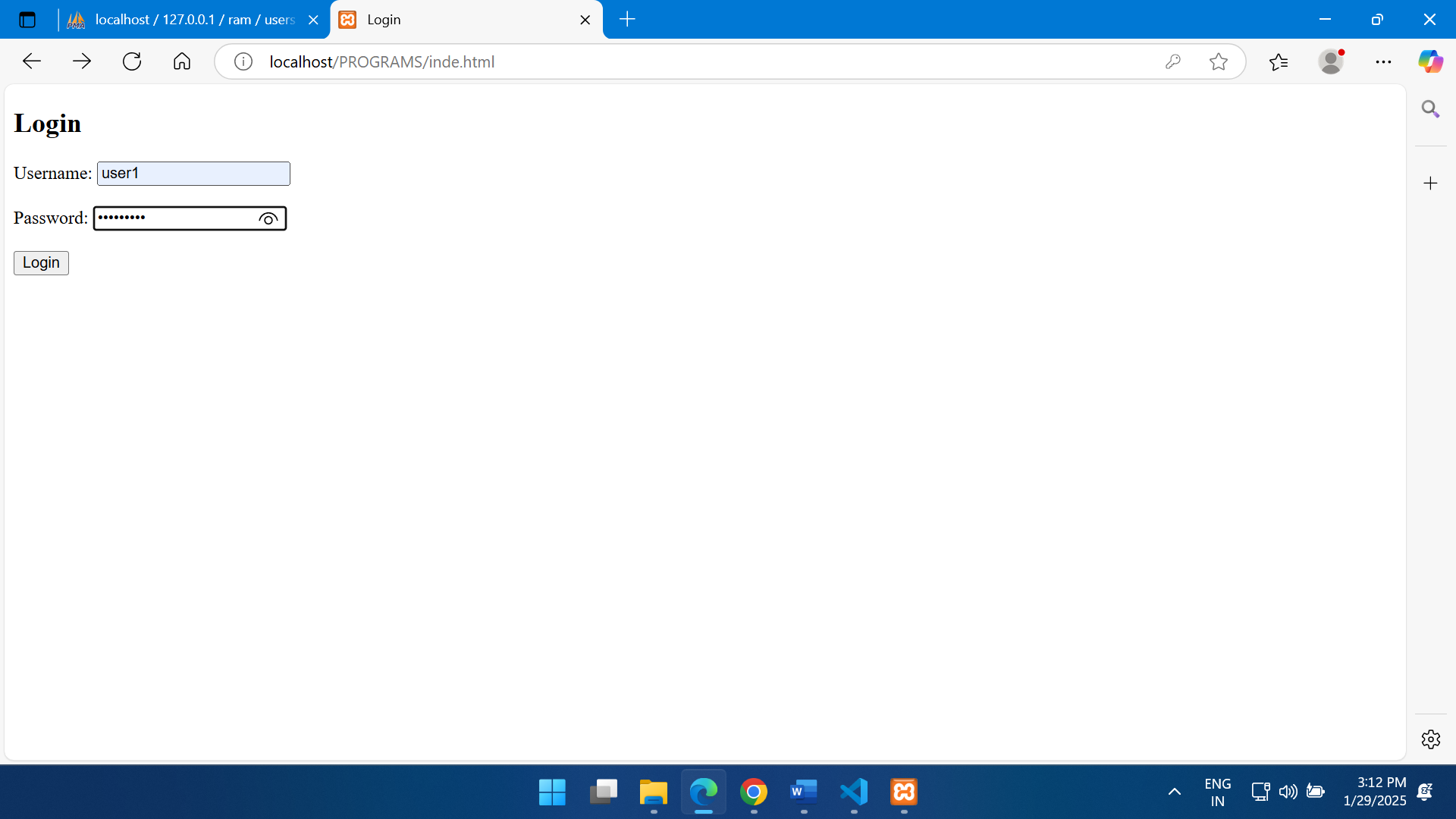
    } else {

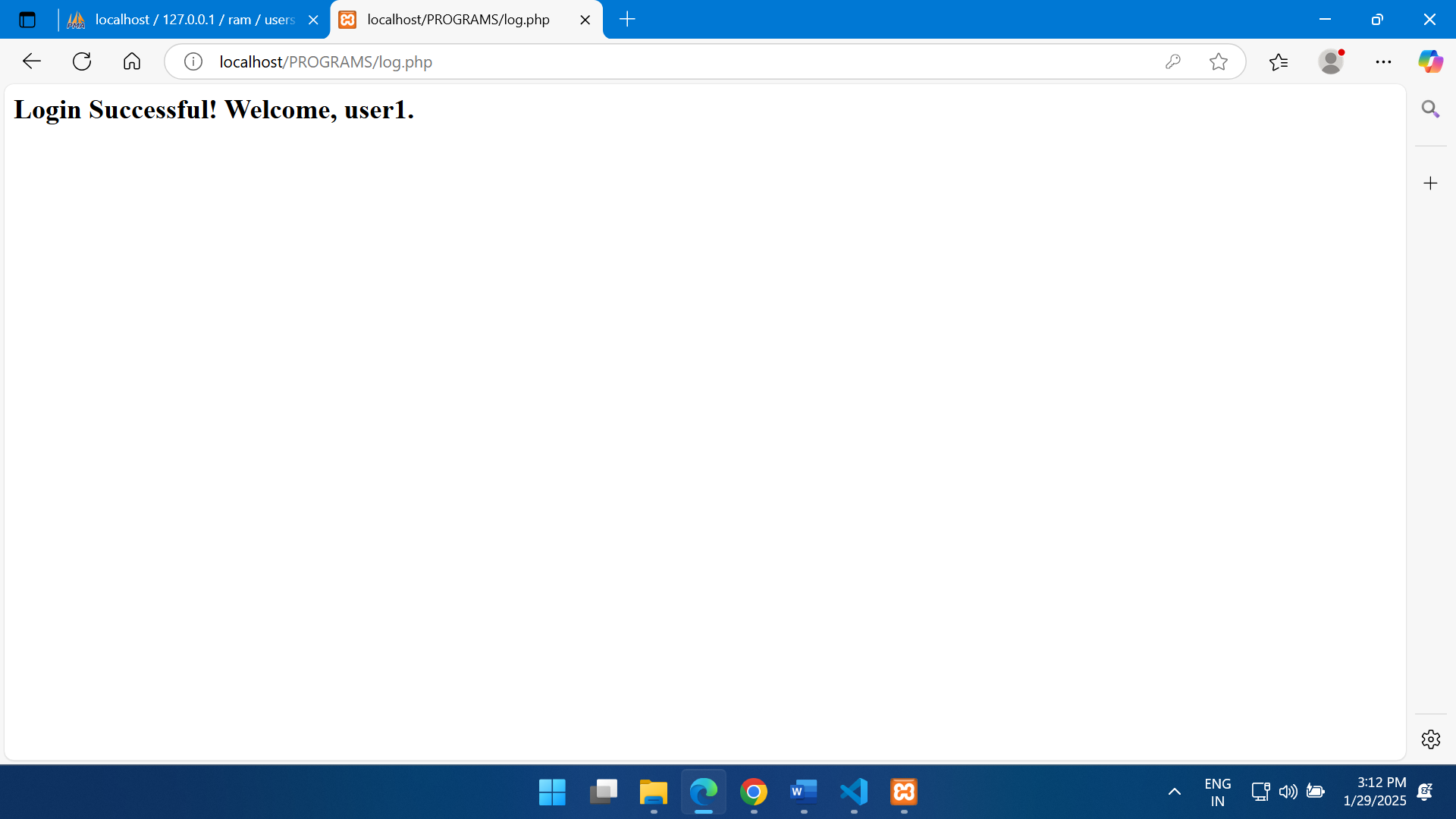
        echo "<h2>Invalid username or password. Please try again.</h2>";

    }

}

?>





**9.Create an XML document that contains 10 users information. Write a Java program, which takes User Id as input and returns the user details by taking the user information from the XML document using a DOM Parser**

**Use.xml**

<?xml version="1.0" encoding="UTF-8"?>

<users>

    <user>

        <id>1</id>

        <name>John Doe</name>

        <email>johndoe@example.com</email>

        <age>30</age>

        <city>New York</city>

    </user>

    <user>

        <id>2</id>

        <name>Jane Smith</name>

        <email>janesmith@example.com</email>

        <age>25</age>

        <city>Los Angeles</city>

    </user>

    <user>

        <id>3</id>

        <name>Robert Brown</name>

        <email>robertbrown@example.com</email>

        <age>35</age>

        <city>Chicago</city>

    </user>

    <user>

        <id>4</id>

        <name>Emily White</name>

        <email>emilywhite@example.com</email>

        <age>28</age>

        <city>Houston</city>

    </user>

    <user>

        <id>5</id>

        <name>Michael Johnson</name>

        <email>michaeljohnson@example.com</email>

        <age>40</age>

        <city>Phoenix</city>

    </user>

    <user>

        <id>6</id>

        <name>Sarah Williams</name>

        <email>sarahwilliams@example.com</email>

        <age>22</age>

        <city>San Diego</city>

    </user>

    <user>

        <id>7</id>

        <name>David Miller</name>

        <email>davidmiller@example.com</email>

        <age>38</age>

        <city>San Antonio</city>

    </user>

    <user>

        <id>8</id>

        <name>Laura Wilson</name>

        <email>laurawilson@example.com</email>

        <age>27</age>

        <city>Dallas</city>

    </user>

    <user>

        <id>9</id>

        <name>James Anderson</name>

        <email>jamesanderson@example.com</email>

        <age>33</age>

        <city>San Francisco</city>

    </user>

    <user>

        <id>10</id>

        <name>Emma Thomas</name>

        <email>emmathomas@example.com</email>

        <age>29</age>

        <city>Seattle</city>

    </user>

</users>

**Use.java**

import org.w3c.dom.\*;

import javax.xml.parsers.\*;

import java.io.\*;

import java.util.Scanner;

public class UserSearch {

    public static void main(String[] args) {

        try {

            // Load and parse the XML file

            File xmlFile = new File("use.xml");

            DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();

            DocumentBuilder builder = factory.newDocumentBuilder();

            Document document = builder.parse(xmlFile);

            // Normalize the document

            document.getDocumentElement().normalize();

            // Get user input for ID search

            Scanner scanner = new Scanner(System.in);

            System.out.print("Enter User ID: ");

            String searchId = scanner.nextLine();

            scanner.close();

            // Get list of all user nodes

            NodeList nodeList = document.getElementsByTagName("user");

            boolean userFound = false;

            // Iterate through user nodes

            for (int i = 0; i < nodeList.getLength(); i++) {

                Node node = nodeList.item(i);

                if (node.getNodeType() == Node.ELEMENT\_NODE) {

                    Element element = (Element) node;

                    String userId = element.getElementsByTagName("id").item(0).getTextContent();

                    if (userId.equals(searchId)) {

                        userFound = true;

                        String name = element.getElementsByTagName("name").item(0).getTextContent();

                        String email = element.getElementsByTagName("email").item(0).getTextContent();

                        String age = element.getElementsByTagName("age").item(0).getTextContent();

                        String city = element.getElementsByTagName("city").item(0).getTextContent();

                        System.out.println("\nUser Details Found:");

                        System.out.println("ID: " + userId);

                        System.out.println("Name: " + name);

                        System.out.println("Email: " + email);

                        System.out.println("Age: " + age);

                        System.out.println("City: " + city);

                        break;

                    }

                }

            }

            if (!userFound) {

                System.out.println("User with ID " + searchId + " not found.");

            }

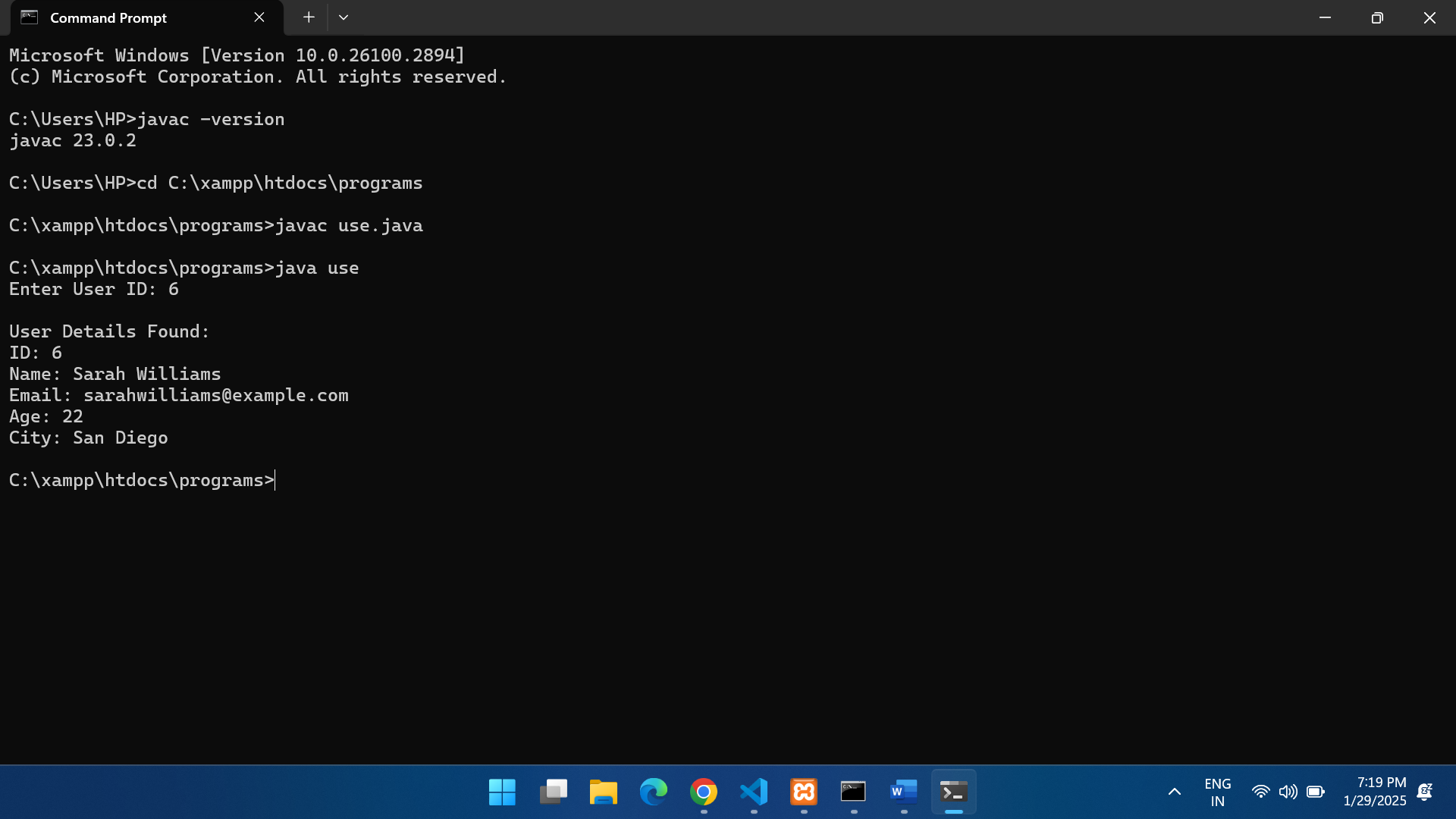
        } catch (Exception e) {

            e.printStackTrace();

        }

    }

}



**9.Create an XML document that contains 10 users information. Write a Java program, which takes User Id as input and returns the user details by taking the user information from the XML document using a**

**b. SAX parser**

**pro9b.xml**

<?xml version="1.0" encoding="UTF-8"?>

<users>

    <user>

        <id>1</id>

        <name>John Doe</name>

        <email>john.doe@example.com</email>

        <age>30</age>

    </user>

    <user>

        <id>2</id>

        <name>Jane Smith</name>

        <email>jane.smith@example.com</email>

        <age>25</age>

    </user>

    <user>

        <id>3</id>

        <name>Robert Brown</name>

        <email>robert.brown@example.com</email>

        <age>40</age>

    </user>

    <user>

        <id>4</id>

        <name>Emily Davis</name>

        <email>emily.davis@example.com</email>

        <age>22</age>

    </user>

    <user>

        <id>5</id>

        <name>Michael Johnson</name>

        <email>michael.johnson@example.com</email>

        <age>35</age>

    </user>

    <user>

        <id>6</id>

        <name>Sarah Wilson</name>

        <email>sarah.wilson@example.com</email>

        <age>28</age>

    </user>

    <user>

        <id>7</id>

        <name>David Lee</name>

        <email>david.lee@example.com</email>

        <age>33</age>

    </user>

    <user>

        <id>8</id>

        <name>Emma White</name>

        <email>emma.white@example.com</email>

        <age>27</age>

    </user>

    <user>

        <id>9</id>

        <name>James Anderson</name>

        <email>james.anderson@example.com</email>

        <age>45</age>

    </user>

    <user>

        <id>10</id>

        <name>Olivia Martin</name>

        <email>olivia.martin@example.com</email>

        <age>29</age>

    </user>

</users>

**Pro9b.java**

**import org.xml.sax.Attributes;**

**import org.xml.sax.SAXException;**

**import org.xml.sax.helpers.DefaultHandler;**

**import javax.xml.parsers.SAXParser;**

**import javax.xml.parsers.SAXParserFactory;**

**import java.io.File;**

**import java.util.Scanner;**

**class UserHandler extends DefaultHandler {**

**private String currentElement = "";**

**private String userId, name, email, age;**

**private boolean isUserFound = false;**

**private String searchId;**

**public UserHandler(String searchId) {**

**this.searchId = searchId;**

**}**

**@Override**

**public void startElement(String uri, String localName, String qName, Attributes attributes) throws SAXException {**

**currentElement = qName;**

**}**

**@Override**

**public void characters(char[] ch, int start, int length) throws SAXException {**

**String data = new String(ch, start, length).trim();**

**if (!data.isEmpty()) {**

**switch (currentElement) {**

**case "id":**

**userId = data;**

**if (userId.equals(searchId)) {**

**isUserFound = true;**

**} else {**

**isUserFound = false;**

**}**

**break;**

**case "name":**

**if (isUserFound) name = data;**

**break;**

**case "email":**

**if (isUserFound) email = data;**

**break;**

**case "age":**

**if (isUserFound) age = data;**

**break;**

**}**

**}**

**}**

**@Override**

**public void endElement(String uri, String localName, String qName) throws SAXException {**

**if (qName.equals("user") && isUserFound) {**

**System.out.println("User Details:");**

**System.out.println("ID: " + userId);**

**System.out.println("Name: " + name);**

**System.out.println("Email: " + email);**

**System.out.println("Age: " + age);**

**System.out.println("----------------------");**

**isUserFound = false; // Reset for next user**

**}**

**}**

**}**

**public class pro9b {**

**public static void main(String[] args) {**

**try {**

**Scanner scanner = new Scanner(System.in);**

**System.out.print("Enter User ID to search: ");**

**String searchId = scanner.nextLine();**

**File inputFile = new File("pro9b.xml");**

**SAXParserFactory factory = SAXParserFactory.newInstance();**

**SAXParser saxParser = factory.newSAXParser();**

**UserHandler userHandler = new UserHandler(searchId);**

**saxParser.parse(inputFile, userHandler);**

**} catch (Exception e) {**

**e.printStackTrace();**

**}**

**}**

**}**

