Training Day 11 Report

30 June 2025

The eleventh day of Web Development training focused on inserting form data into a MySQL database using PHP. We explored how to capture form inputs and safely store them in relevant tables. The concept of handling form submission with error and success messages was also introduced.

1. Inserting Data into Database

We learned how to use the INSERT INTO SQL command in PHP to add user-submitted data into MySQL tables.

Example:

```
<?php
$conn = mysqli_connect("localhost", "root", "", "alumni_portal");

$name = $_POST['name'];
$email = $_POST['email'];
$year = $_POST['graduation_year'];

$sql = "INSERT INTO alumni (name, email, graduation_year) VALUES ('$name', '$email', '$year')";

if (mysqli_query($conn, $sql)) {
  echo "Record inserted successfully";
} else {
  echo "Error: " . mysqli_error($conn);
}
?>
```

2. Handling Form Submissions

Collected data from input fields using $_POST

Wrote PHP scripts to handle the insert operation

Displayed success or error messages after form submission

Applied mysqli_real_escape_string() to prevent SQL injection (introduced briefly)

3. Hands-on Practice

Designed a simple registration form for alumni

Connected the form to a PHP script for database insertion

Verified data insertion using phpMyAdmin

Handled edge cases like empty input and duplicate records

4. Relevance to Project

This functionality is directly related to alumni registration. When alumni fill out the registration form, their data will be stored in the system's database and can later be retrieved for various features.

5. Learning Outcomes

Learned how to insert user input into MySQL using PHP

Practiced real-time form processing and data storage

Understood the importance of validating and escaping inputs

Made the portal more interactive and data-driven

By: Rida

URN: 2302647

CRN: 2315192

D3(CSE-D)