college CDI

Database concepts (CBD)

Class 8

Writing to a database using PHP

Now that we know how to search database in order to display data into a web document, it will become quite handy to be able to write to the database in order to add or to modify data. Doing so using PHP is not quite different than it is using MySQL.

Example (MySQLi Object-oriented)

```php
<?php
// Server info
$servername = "localhost";
$username = "root";
$password = "root";
$dbname = "My_test_DB";

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

// Check connection
if ($conn->connect_error) {
        die("Connection failed: " . $conn->connect_error);
}

// Query
$sql = "INSERT INTO library (title, author, year) VALUES ('My new book', 'So Me', '2020')";

// Check data creation
if ($conn->query($sql) === TRUE) {
        echo "New record created successfully";
} else {
        echo "Error: " . $sql . "<br>" . $conn->error;
}

$conn->close();
?>
```

Explanation :
After defining the server info and having created a successful connection, the query is stored in the variable *$sql* (in this case, 3 values are added to 3 columns as a new entry). A conditional structure checks if the query was correctly executed and displays an error or a success message accordingly.

Example (MySQLi Procedural)

```php
<?php
// Server info
$servername = "localhost";
$username = "root";
$password = "root";
$dbname = "My_test_DB";

// Create connection
$conn = mysqli_connect($servername, $username, $password, $dbname);

// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}

// Query
$sql = "INSERT INTO library (title, author, year) VALUES ('My new book', 'So Me', '2020')";

// Check data creation
if (mysqli_query($conn, $sql)) {
    echo "New record created successfully";
} else {
    echo "Error: " . $sql . "<br>" . mysqli_error($conn);
}

$conn->close();
?>
```

Example (PDO)

```php
<?php
$servername = "localhost";
$username = "root";
$password = "root";
$dbname = "My_test_DB";

try {
$conn = new PDO("mysql:host=$servername;dbname=$dbname", $username, $password);

// set the PDO error mode to exception
$conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
$sql = "INSERT INTO MyGuests (title, author, year) VALUES ('My Book', 'Me', '2020')";

// use exec() because no results are returned
$conn->exec($sql);
echo "New record created successfully";
    }
catch(PDOException $e){
        echo $sql . "<br>" . $e->getMessage();
}

$conn = null;
?>
```

Writing in a database using a PHP form

Writing into a database using a form can be very useful whether it is to let a user subscribe to a mailing list or in the context of a CMS.

HTML:

```html
<form action="#" method="post">
        <label for="title">Title:</label>
        <input type="text" name="title">

        <label for="author">Author:</label>
        <input type="text" name="author">

        <label for="year">Year:</label>
        <input type="text" name="year">

        <button>Create new entry</button>
</form>
```

PHP:

```php
<?php
$db = new mysqli('localhost', 'root', 'root', 'My_test_DB');        // Connect to server

if ($db->connect_error) {                                            // Connection error message
        die("Connection failed: " . $db->connect_error);
}

$sql = "INSERT INTO library (title, author, year)                    // Query
VALUES ('$_POST[title]','$_POST[author]','$_POST[year]')";

if (!mysqli_query($db, $sql)){                                       // Query error message
        die('Error: ' . $db_error());
}

echo "1 record added";                                               // Confirmation message

$db->close();
?>
```

IMPORTANT: Always validate the form to prevent from malicious SQL injection.