

## Bài tập chương 4

### A. . Code đã hoàn thiện

```
<?php
$host = '127.0.0.1';
$dbname = 'cse485_web';
$username = 'root';
$password = '';

$dsn =
"mysql:host=$host;dbname=$dbname;charset=utf8mb4";

try {
    // TODO 1: Tạo đối tượng PDO để kết nối CSDL
    $pdo = new PDO($dsn, $username, $password);
    $pdo->setAttribute(PDO::ATTR_ERRMODE,
PDO::ERRMODE_EXCEPTION);
    echo "Kết nối thành công!";
} catch (PDOException $e) {
    die("Kết nối thất bại: " . $e->getMessage());
}

// TODO 2: Kiểm tra xem form đã được gửi đi (method
POST)
if ($_SERVER['REQUEST_METHOD'] === 'POST' &&
isset($_POST['ten_sinh_vien'])) {

    // TODO 3: Lấy dữ liệu 'ten_sinh_vien' và 'email' từ
$_POST
    $ten = $_POST['ten_sinh_vien'];
    $email = $_POST['email'];

    // TODO 4: Viết câu lệnh SQL INSERT với Prepared
Statement (dùng dấu ?)
    $sql = "INSERT INTO sinhvien (ten_sinh_vien, email)
VALUES (?, ?)";

    // TODO 5: Chuẩn bị (prepare) và thực thi (execute)
câu lệnh
    $stmt = $pdo->prepare($sql);
```

```

$stmt->execute([$ten, $email]);

// TODO 6: (Tùy chọn) Chuyển hướng về chính trang
này để "làm mới"
// Việc này giúp tránh gửi lại form khi người dùng
nhấn F5
header('Location: index.php');
exit;
}

// TODO 7: Viết câu lệnh SQL SELECT *
$sql_select = "SELECT * FROM sinhvien ORDER BY ngay_tao
DESC";

// TODO 8: Thực thi câu lệnh SELECT (không cần prepare
vì không có tham số)
$stmt_select = $pdo->query($sql_select);
?>

<!DOCTYPE html>
<html lang="vi">
<head>
    <meta charset="UTF-8">
    <title>PHT Chương 4 Website hướng dữ liệu</title>
    <style>
        table { width: 100%; border-collapse: collapse;
    }
        th, td { border: 1px solid #ddd; padding: 8px; }
        th { background-color: #f2f2f2; }
    </style>
</head>
<body>

    <h2>Thêm Sinh Viên Mới (Chủ đề 4.3)</h2>
    <form action="index.php" method="POST">
        Tên sinh viên: <input type="text"
name="ten_sinh_vien" required>
        Email: <input type="email" name="email"
required>
        <button type="submit">Thêm</button>

```

```

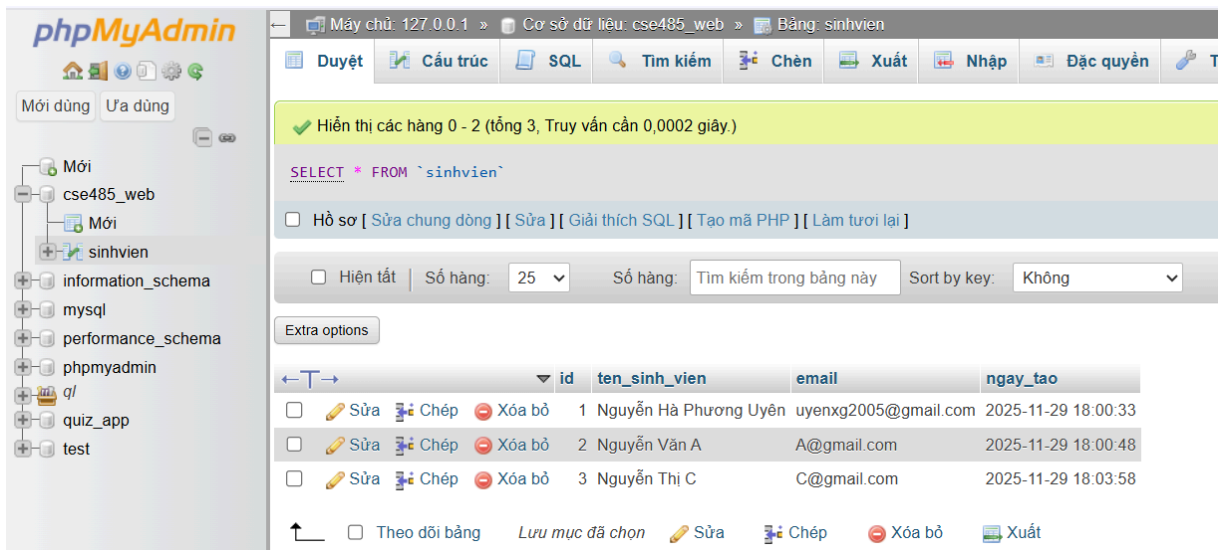
</form>

<h2>Danh Sách Sinh Viên (Chủ đề 4.2)</h2>
<table>
    <tr>
        <th>ID</th>
        <th>Tên Sinh Viên</th>
        <th>Email</th>
        <th>Ngày Tạo</th>
    </tr>
    <?php
        // TODO 9: Dùng vòng lặp (ví dụ: while) để duyệt
qua kết quả
        while ($row =
$stmt_select->fetch(PDO::FETCH_ASSOC)) {
            // TODO 10: In (echo) các dòng <tr> và <td>
chứa dữ liệu $row
            // Sử dụng htmlspecialchars để bảo mật chống
XSS như gợi ý
            echo "<tr>";
            echo "<td>" . htmlspecialchars($row['id']) .
"</td>";
            echo "<td>" .
htmlspecialchars($row['ten_sinh_vien']) . "</td>";
            echo "<td>" .
htmlspecialchars($row['email']) . "</td>";
            echo "<td>" .
htmlspecialchars($row['ngay_tao']) . "</td>";
            echo "</tr>";
        }
    ?>
</table>
</body>
</html>

```

## B. Ảnh chụp màn hình Kết quả:

### 1. Ảnh 1



### 2. Ảnh 2

Kết nối thành công!

#### Thêm Sinh Viên Mới (Chủ đề 4.3)

Tên sinh viên:  Email:

#### Danh Sách Sinh Viên (Chủ đề 4.2)

ID	Tên Sinh Viên	Email	Ngày Tạo
3	Nguyễn Thị C	C@gmail.com	2025-11-29 18:03:58
2	Nguyễn Văn A	A@gmail.com	2025-11-29 18:00:48
1	Nguyễn Hà Phương Uyên	uyenxg2005@gmail.com	2025-11-29 18:00:33

## Câu hỏi phản biện:

Vì sao khi mà người ta ấn post xong rồi thì vẫn giữ nguyên cái trạng thái post đấy để mà khi tải lại trang trình duyệt lại gửi yêu cầu post đến server mà không tự động chuyển sang trạng thái get? Dữ liệu tải lên được lưu trữ tạm thời ở đâu trong trình duyệt giúp cho lúc tải lại trang nó lại được post thêm một lần nữa?