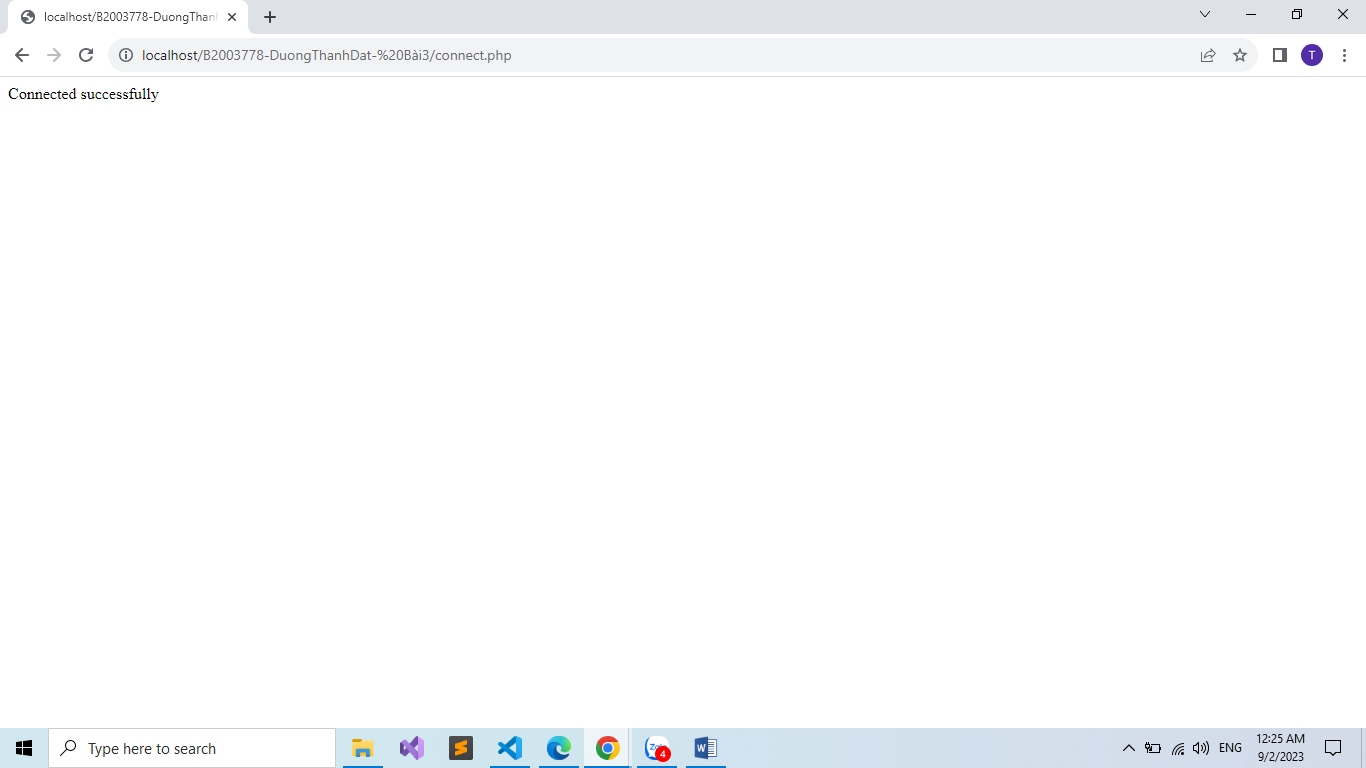
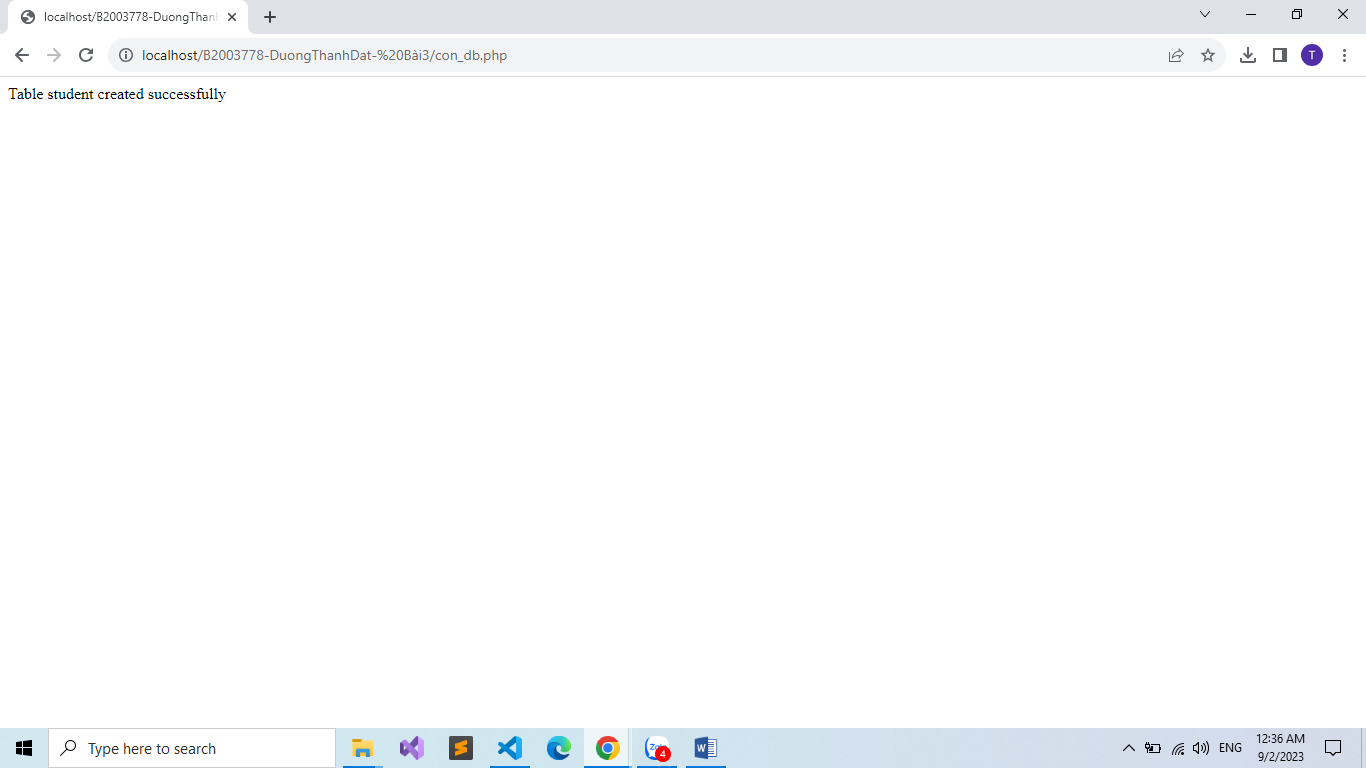
1. Bạn hãy chạy tất cả các lệnh hướng dẫn ở trên và chụp lại màn hình kết quả.

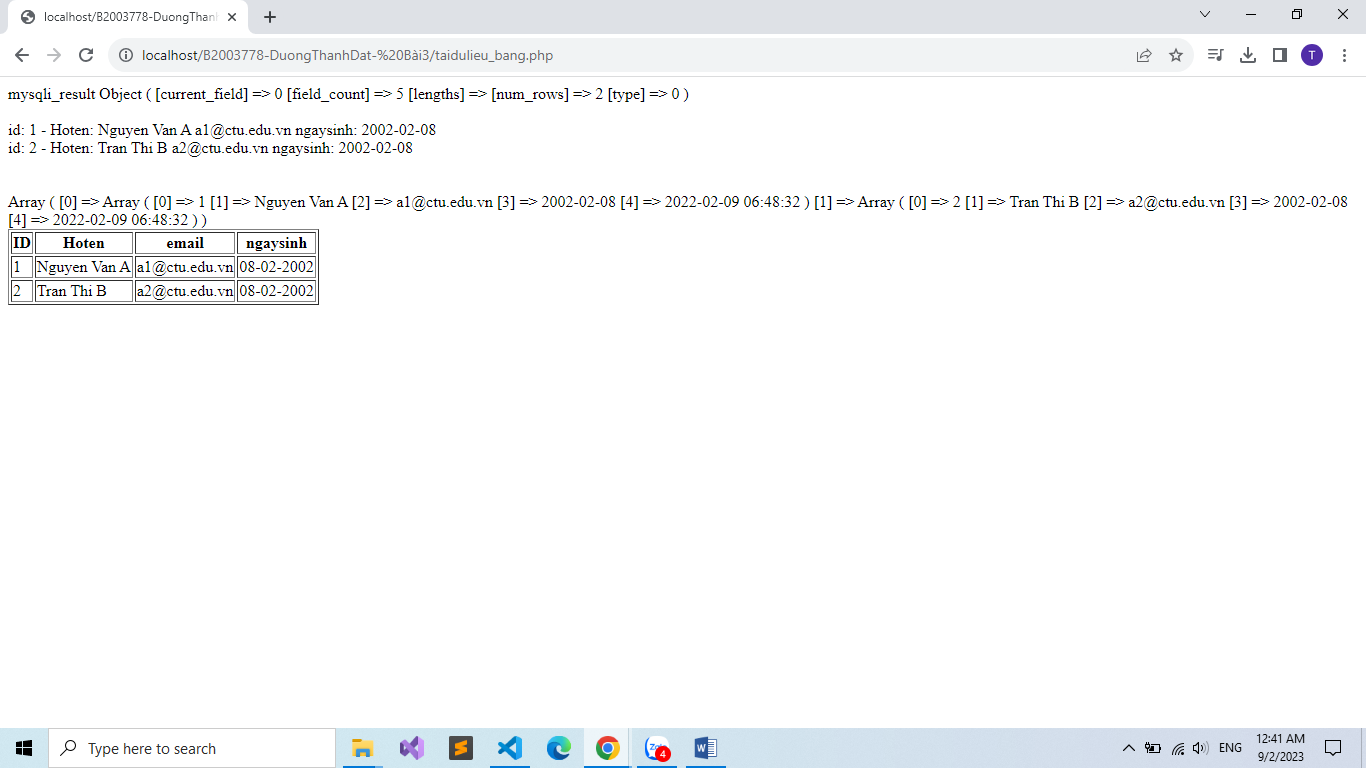
* connect.php



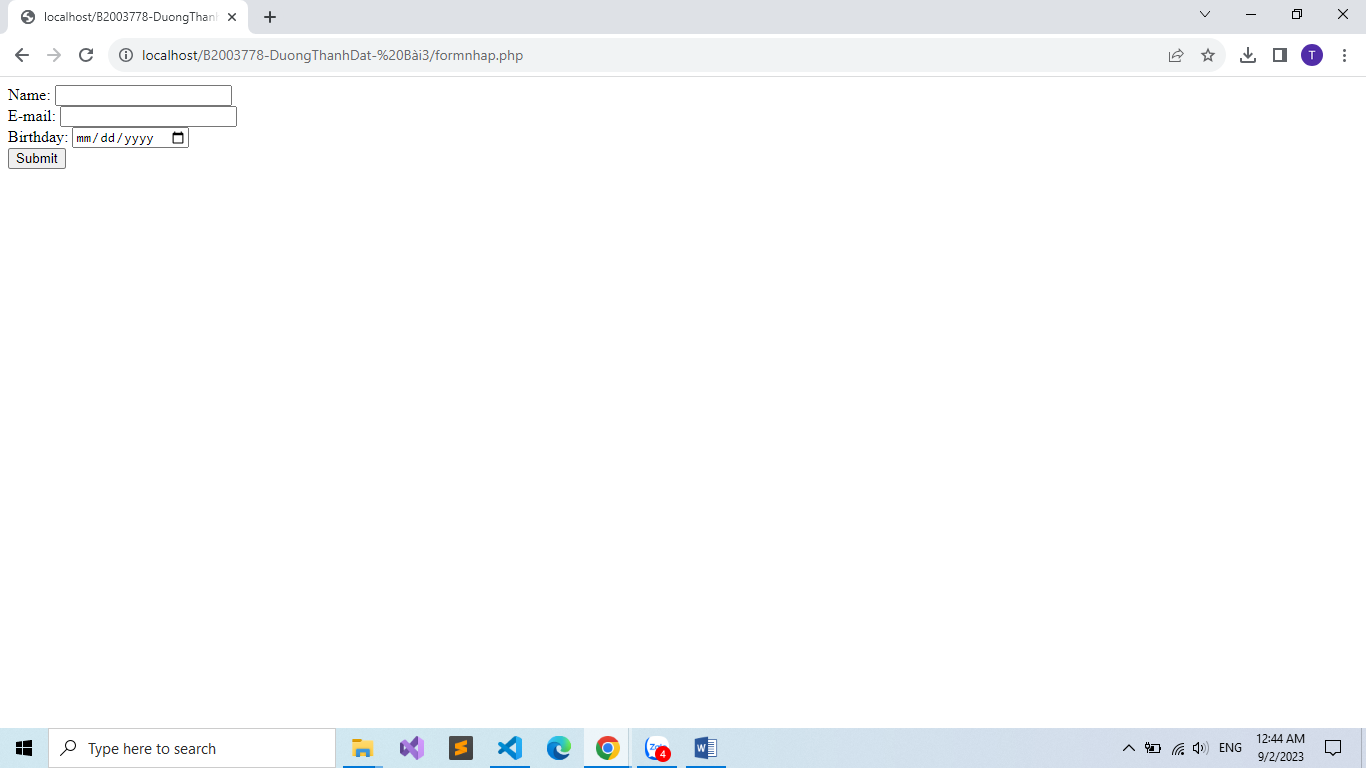
* con\_db.php



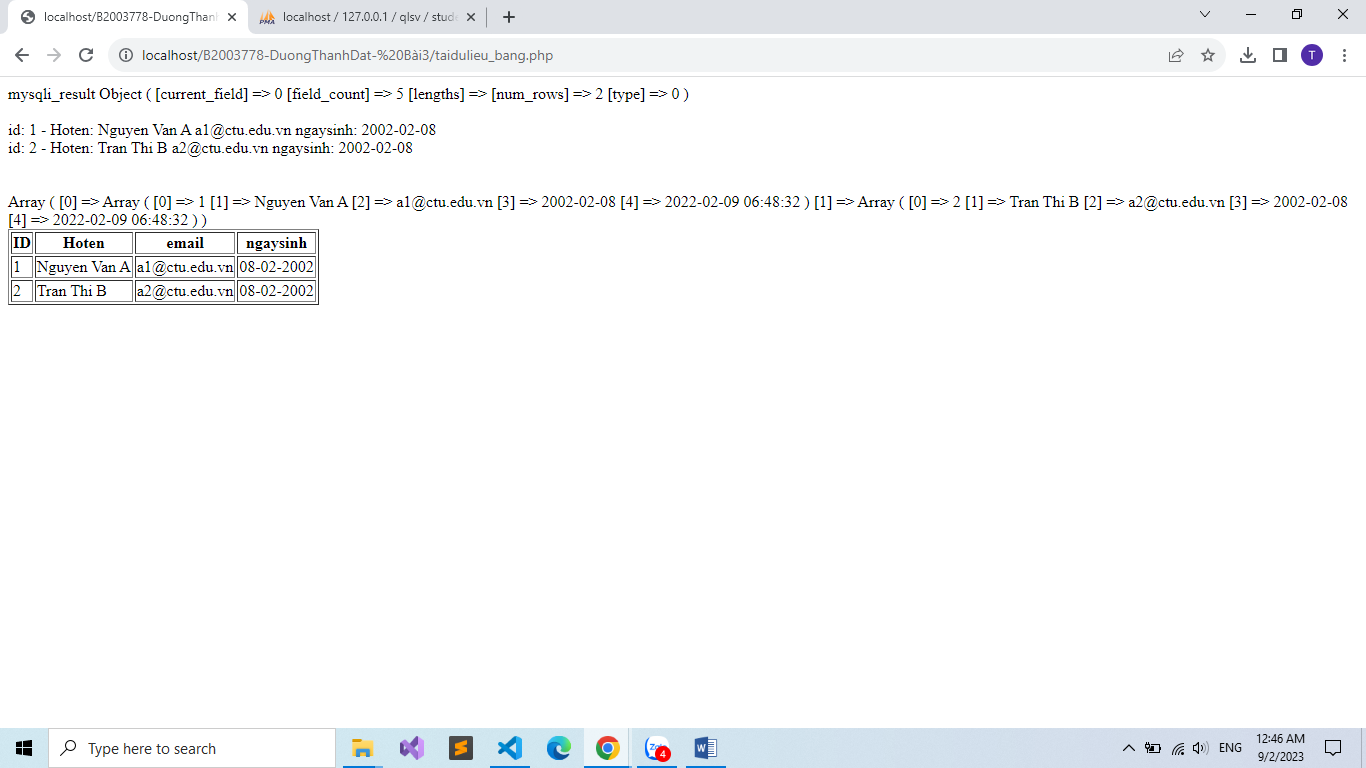
* taidulieu\_bang.php



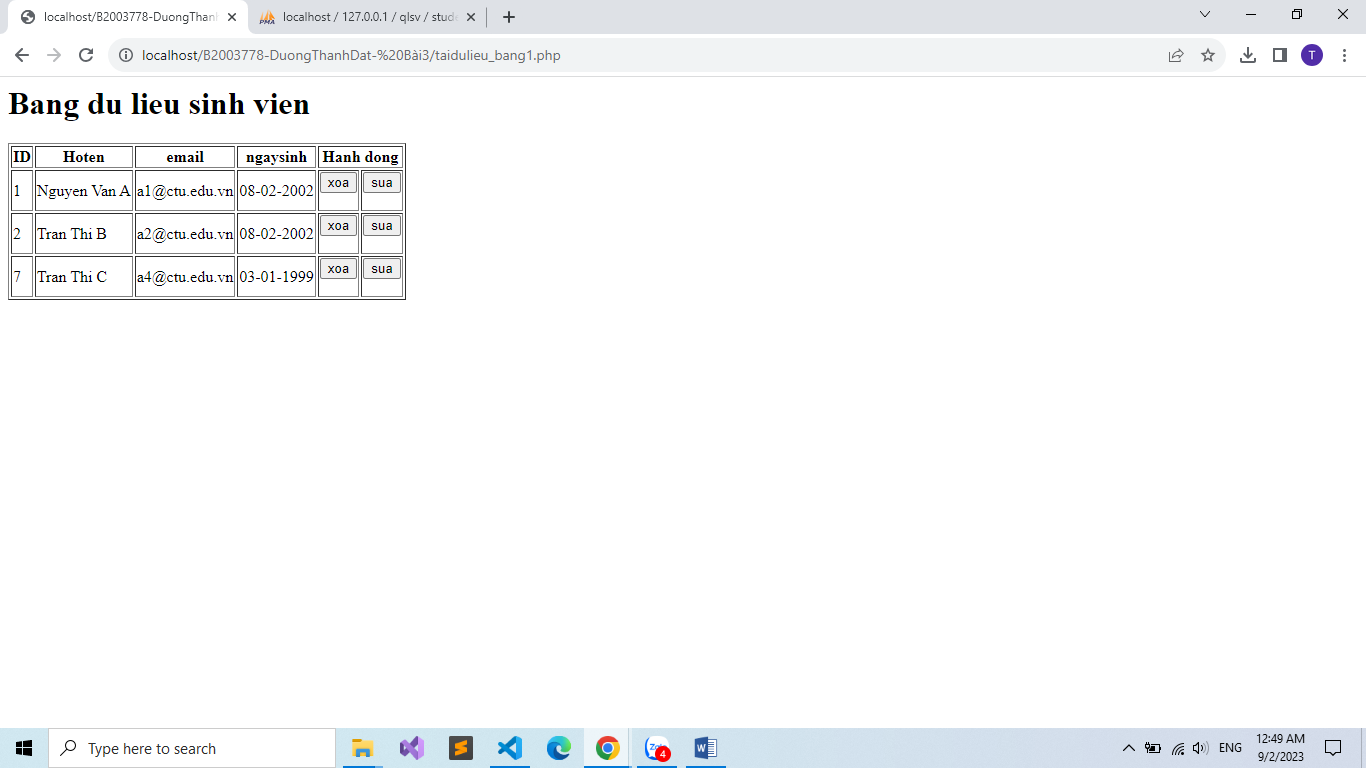
* formnhap.php



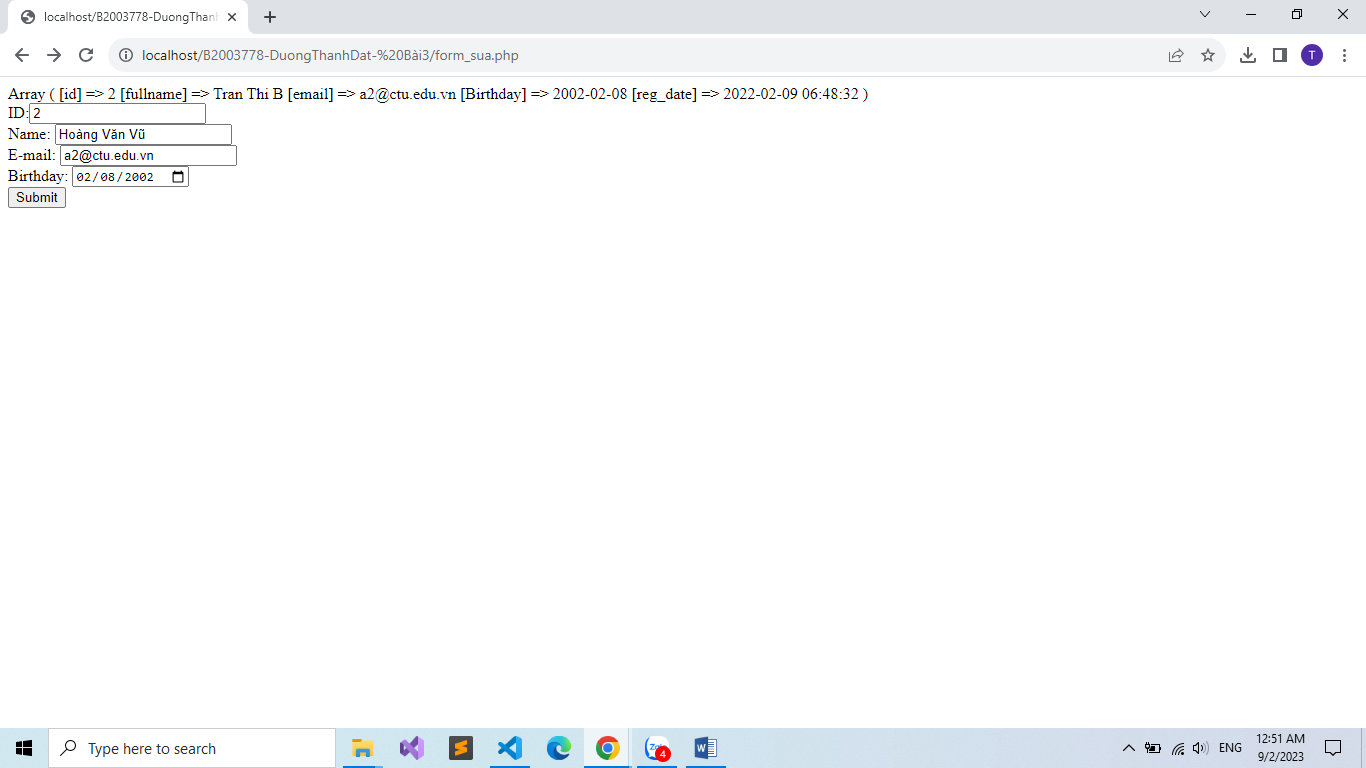
* luu.php



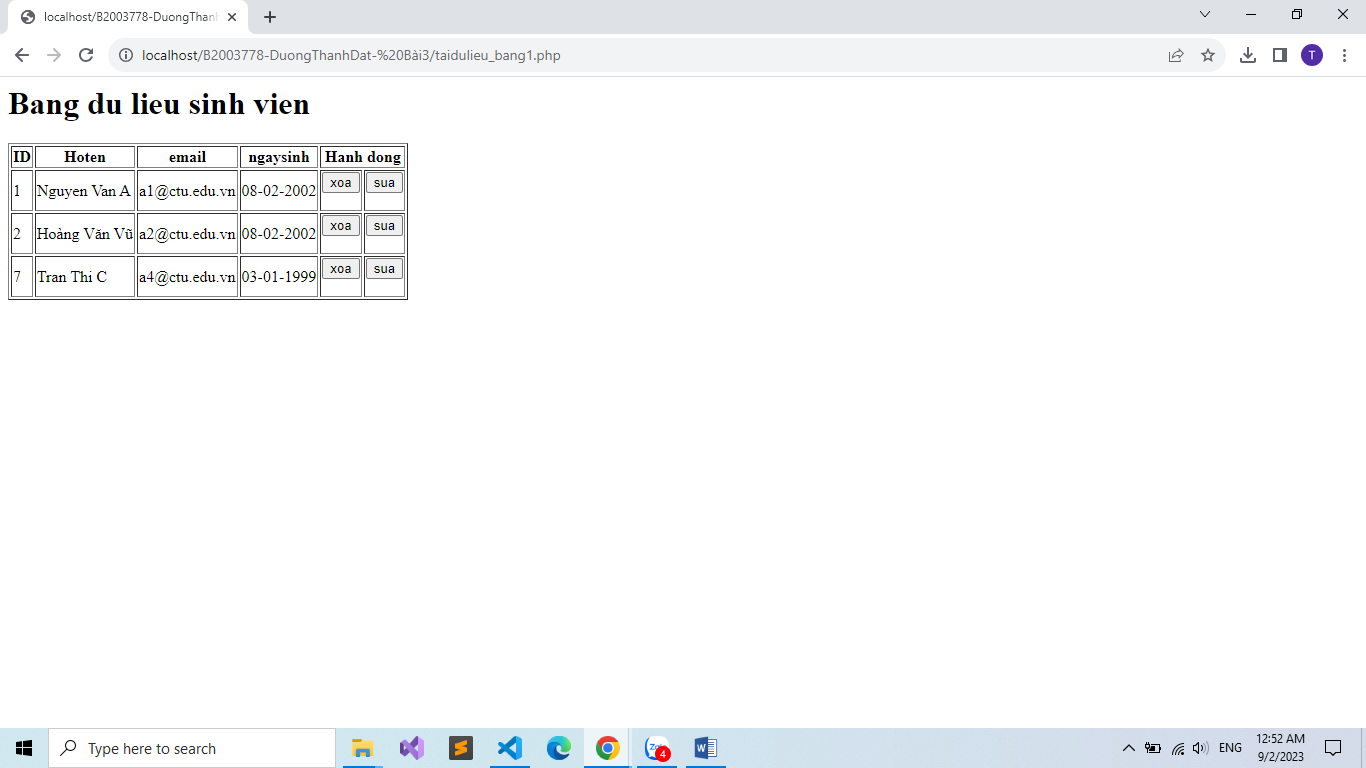
* taidulieu\_bang1.php



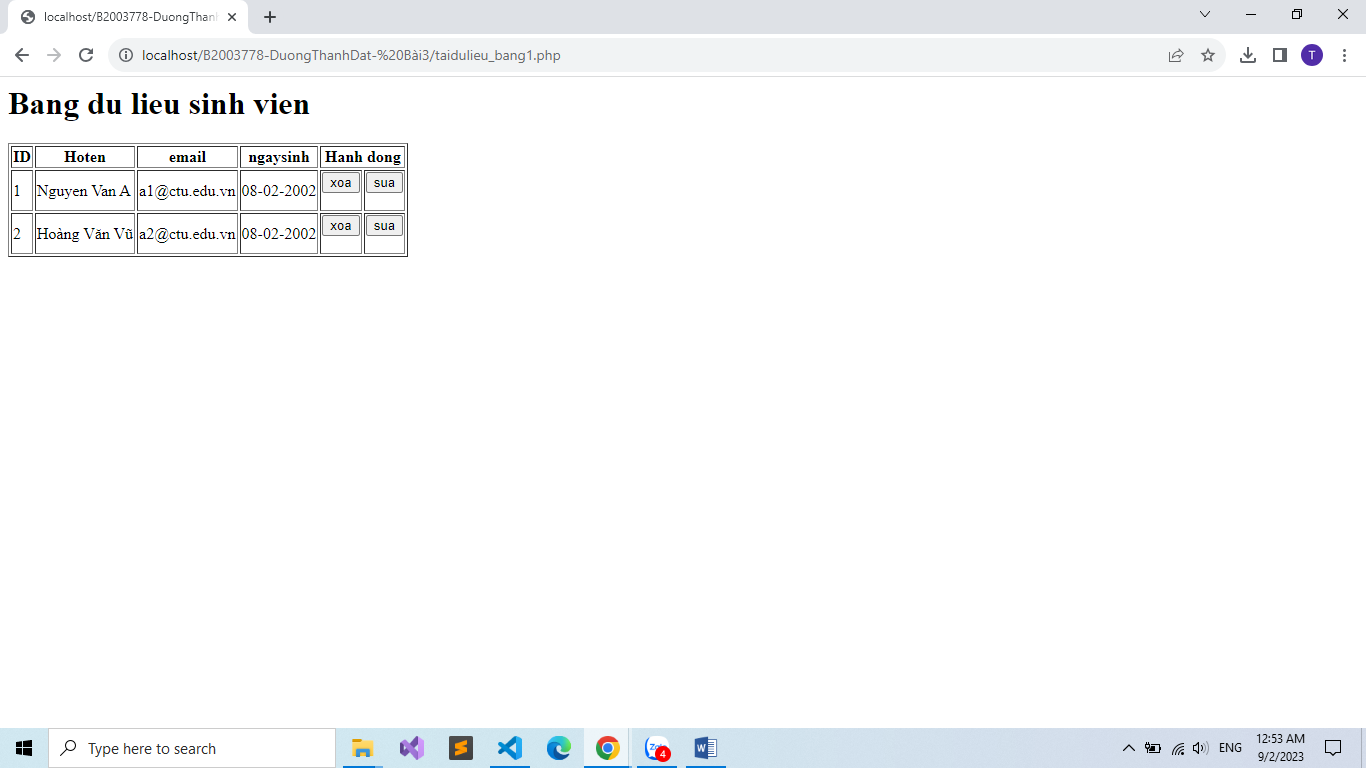
* form\_sua.php



* sua.php



* xoa.php



2. Từ code kết nối với MySQL, bạn hãy tìm và trình bày code kết nối với các hệ quản trị CSDL khác như Oracle, SQL server, SQLite.

* Code kết nối với Oracle:

<?php

$servername = "localhost";

$username = "root";

$password = "";

$database = "dbname";

$conn = oci\_connect($username, $password, $servername/$database);

if (!$conn) {

    $e = oci\_error();

    trigger\_error(htmlentities($e['message'], ENT\_QUOTES), E\_USER\_ERROR);

}

?>

* Code kết nối với SQL Server:

<?php

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "dbname";

$conn = new PDO("sqlsrv:Server=$servername;Database=$dbname", $username, $password);

if (!$conn) {

    die("Connection failed: " . print\_r(sqlsrv\_errors(), true));

}

?>

* Code kết nối với SQLite:

<?php

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "dbname";

$conn = new PDO("sqlite:$dbname");

if (!$conn) {

    die("Connection failed: " . $conn->lastErrorMsg());

}

?>

3. Cho biết class mysqli để hỗ trợ thực hiện những điều gì. Hãy liệt kê và mô tả các phương thức trong class mysqli (tham khảo ở 2 hoặc các tài liệu khác mà bạn tìm được).

Lớp **mysqli** trong PHP cung cấp các phương thức để thực hiện các thao tác với cơ sở dữ liệu MySQL. Dưới đây là một số phương thức quan trọng trong lớp **mysqli**:

* \_\_construct(): Phương thức khởi tạo để tạo đối tượng mysqli.
* connect(): Kết nối đến cơ sở dữ liệu MySQL.
* close(): Đóng kết nối đến cơ sở dữ liệu.
* query(): Thực thi một câu lệnh SQL và trả về kết quả dưới dạng một đối tượng mysqli\_result hoặc false nếu có lỗi.
* prepare(): Chuẩn bị một câu lệnh SQL để thực thi và trả về một đối tượng mysqli\_stmt.
* real\_escape\_string(): Nhận một chuỗi và trả về một phiên bản đã được xử lý để tránh các lỗi SQL injection.
* affected\_rows(): Trả về số hàng bị ảnh hưởng bởi câu lệnh INSERT, UPDATE hoặc DELETE gần nhất.
* insert\_id(): Trả về ID tự động được tạo bởi câu lệnh INSERT gần nhất.
* num\_rows(): Trả về số hàng trong kết quả của một câu lệnh SELECT.
* fetch\_assoc(): Trả về một mảng kết hợp chứa dữ liệu từ hàng tiếp theo trong kết quả.
* fetch\_array(): Trả về một mảng chứa dữ liệu từ hàng tiếp theo trong kết quả, với cả hai kiểu kết hợp và số nguyên.
* fetch\_object(): Trả về một đối tượng chứa dữ liệu từ hàng tiếp theo trong kết quả.
* fetch\_row(): Trả về một mảng chứa dữ liệu từ hàng tiếp theo trong kết quả, chỉ sử dụng kiểu số nguyên.
* error(): Trả về thông báo lỗi của lần gọi phương thức gần nhất.
* errno(): Trả về mã lỗi của lần gọi phương thức gần nhất.

4. Trong tập tin connect.php, Bạn hãy cho biết ý nghĩa của phương thức die() và cách dùng.

Phương thức **die()** trong PHP được sử dụng để dừng thực thi chương trình và hiển thị một thông báo lỗi nếu có. Nó có thể được sử dụng để xử lý lỗi hoặc thông báo trạng thái và dừng chương trình ngay lập tức.

Cú pháp của phương thức **die()** là:

die(message);

Trong đó:

* **message** là thông báo lỗi hoặc thông báo trạng thái sẽ được hiển thị.

Khi phương thức **die()** được gọi, chương trình sẽ dừng lại và không thực hiện bất kỳ mã lệnh nào nằm sau đó.

Trong đoạn mã mà bạn đã cung cấp, nếu kết nối tới cơ sở dữ liệu MySQL không thành công, phương thức **die()** được sử dụng để dừng chương trình và hiển thị thông báo lỗi kết nối không thành công. Nếu kết nối thành công, thông báo "Connected successfully" sẽ được hiển thị.

5. Trong tập tin taidulieu\_bang.php có dùng phương thức fetch\_assoc và fetch\_all để lấy dữ liệu từ đối tượng $result. Ngoài 2 phương thức này, các bạn hãy liệt kê các phương thức khác trong mysqli có chức năng tương tự, và hãy chạy thử các phương thức này như các cách hiển thị dữ liệu cách 4, cách 5, cách 6… trong tập tin taidulieu\_bang.php.

Một số phương thức khác trong lớp mysqli có chức năng tương tự:

* fetch\_array(): Trả về một mảng chứa dữ liệu từ hàng tiếp theo trong kết quả, với cả hai kiểu kết hợp và số nguyên.
* fetch\_object(): Trả về một đối tượng chứa dữ liệu từ hàng tiếp theo trong kết quả.
* fetch\_row(): Trả về một mảng chứa dữ liệu từ hàng tiếp theo trong kết quả, chỉ sử dụng kiểu số nguyên.

Cách sử dụng các phương thức này:

* Cách 4: Sử dụng **fetch\_array()**

while ($row = $result->fetch\_array()) {

    echo "id: " . $row["id"] . " - Hoten: " . $row["fullname"] . " " .

        $row["email"] . ' ngaysinh: ' . $row['Birthday'] . "";

}

* Cách 5: Sử dụng **fetch\_object()**

while ($row = $result->fetch\_object()) {

    echo "id: " . $row->id . " - Hoten: " . $row->fullname . " " .

        $row->email . ' ngaysinh: ' . $row->Birthday . "";

}

* Cách 6: Sử dụng **fetch\_row()**

while ($row = $result->fetch\_row()) {

    echo "id: " . $row[0] . " - Hoten: " . $row[1] . " " .

        $row[2] . ' ngaysinh: ' . $row[3] . "";

}

6. Phương thức header trong hàm luu.php để di chuyển đến trang khác sau khi chúng ta thực hiện xong các hành động nào đó (như di chuyển đến trang chủ khi xong hành động sửa/xóa,...).Hãy mô tả chi tiết cách sử dụng, và công dụng của header (Tham khảo 3 ).

Hàm **header()** trong PHP được sử dụng để gửi một tiêu đề HTTP tới trình duyệt, điều hướng người dùng đến một trang web hoặc tài nguyên khác. Nó thường được sử dụng để chuyển hướng người dùng đến một trang mới sau khi thực hiện xong một hành động nhất định, ví dụ như thêm, sửa hoặc xóa dữ liệu.

Cú pháp của hàm **header()** là:

header(string $header, bool $replace = true, int $http\_response\_code = 0)

Trong đó:

* **$header** là chuỗi tiêu đề HTTP sẽ được gửi.
* **$replace** là một cờ chỉ định xem liệu tiêu đề đã được gửi trước đó có được thay thế hay không. Mặc định là true.
* **$http\_response\_code** là mã HTTP response code. Mặc định là 0.

Trong file **luu.php,** hàm **header()** được sử dụng để chuyển hướng người dùng đến trang **taidulieu\_bang.php** sau khi thêm sinh viên thành công vào cơ sở dữ liệu. Điều này giúp người dùng xem kết quả mới nhất sau khi thực hiện hành động thêm.

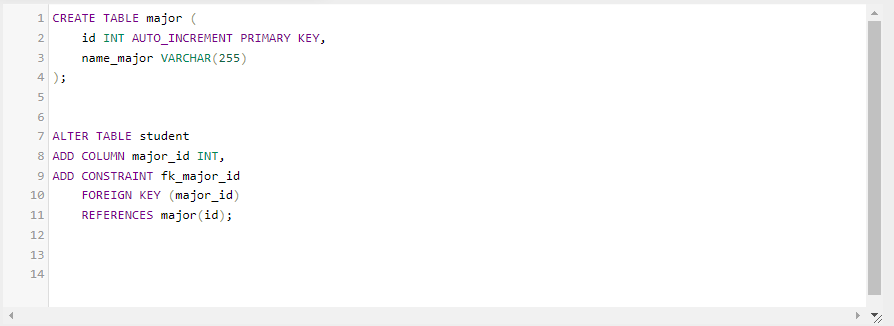
Vì **header()** phải được gọi trước khi bất kỳ đầu ra nào được gửi đến trình duyệt, nên nó thường được sử dụng trước khi in bất kỳ mã HTML hoặc dữ liệu nào khác.

Lưu ý :

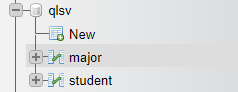
* Không được gửi bất kỳ đầu ra nào trước khi gọi hàm header(). Nếu có đầu ra trước đó, bạn có thể sử dụng hàm ob\_start() để bắt đầu bộ đệm đầu ra và sau đó gọi ob\_end\_flush() để gửi đầu ra và tiếp tục chuyển hướng.
* Sau khi gọi hàm header() và chuyển hướng, bạn nên sử dụng exit hoặc die để dừng thực thi chương trình ngay lập tức và ngăn chặn bất kỳ mã lệnh nào tiếp theo được thực thi.

7. Vào CSDL qlsv, bạn hãy viết lệnh tạo thêm 1 bảng sau để có thêm thông tin về chuyên ngành (kiểu dữ liệu các bạn tự cho sao cho phù hợp): major(id, name\_major) Và thêm cột khóa ngoại major\_id vào bảng student student(id, fullname, email, birthday, major\_id)

Lệnh tạo bảng và thêm khóa ngoại:

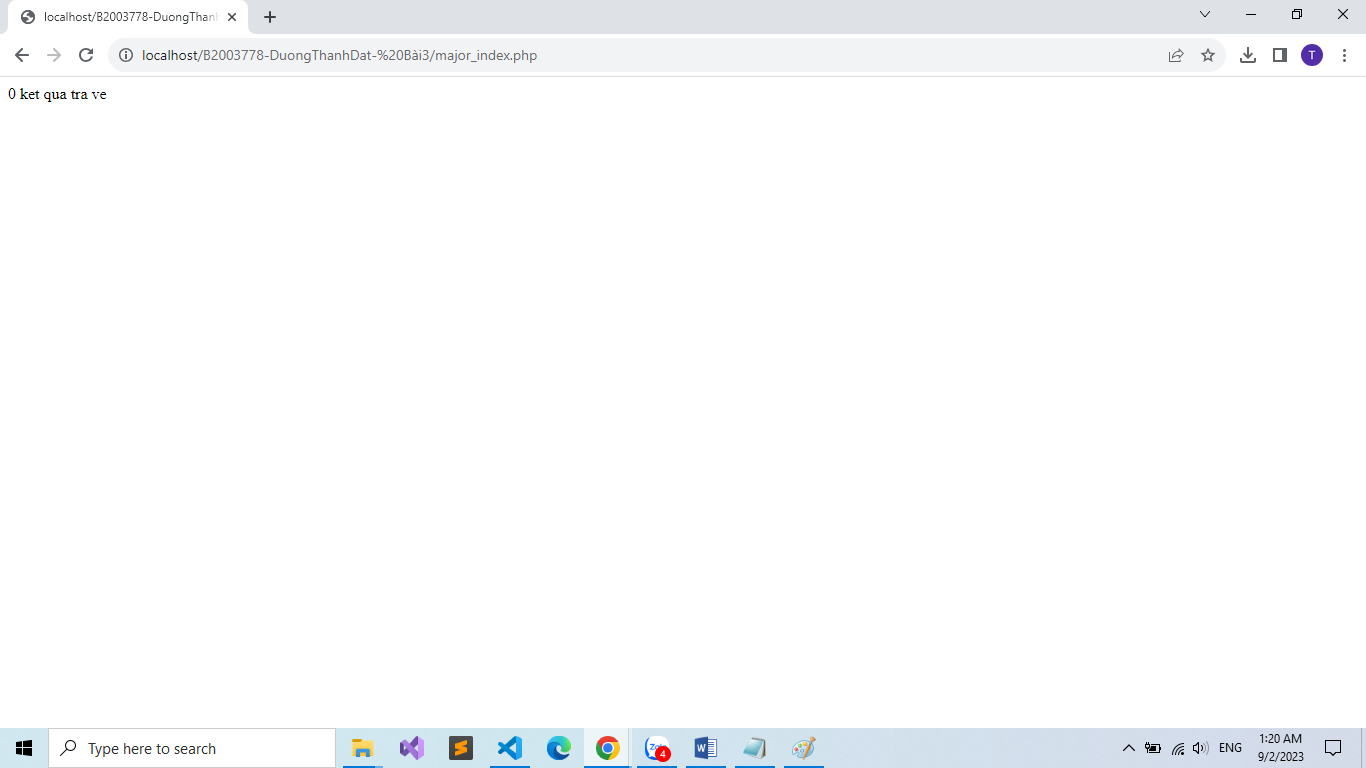


Kết quả:

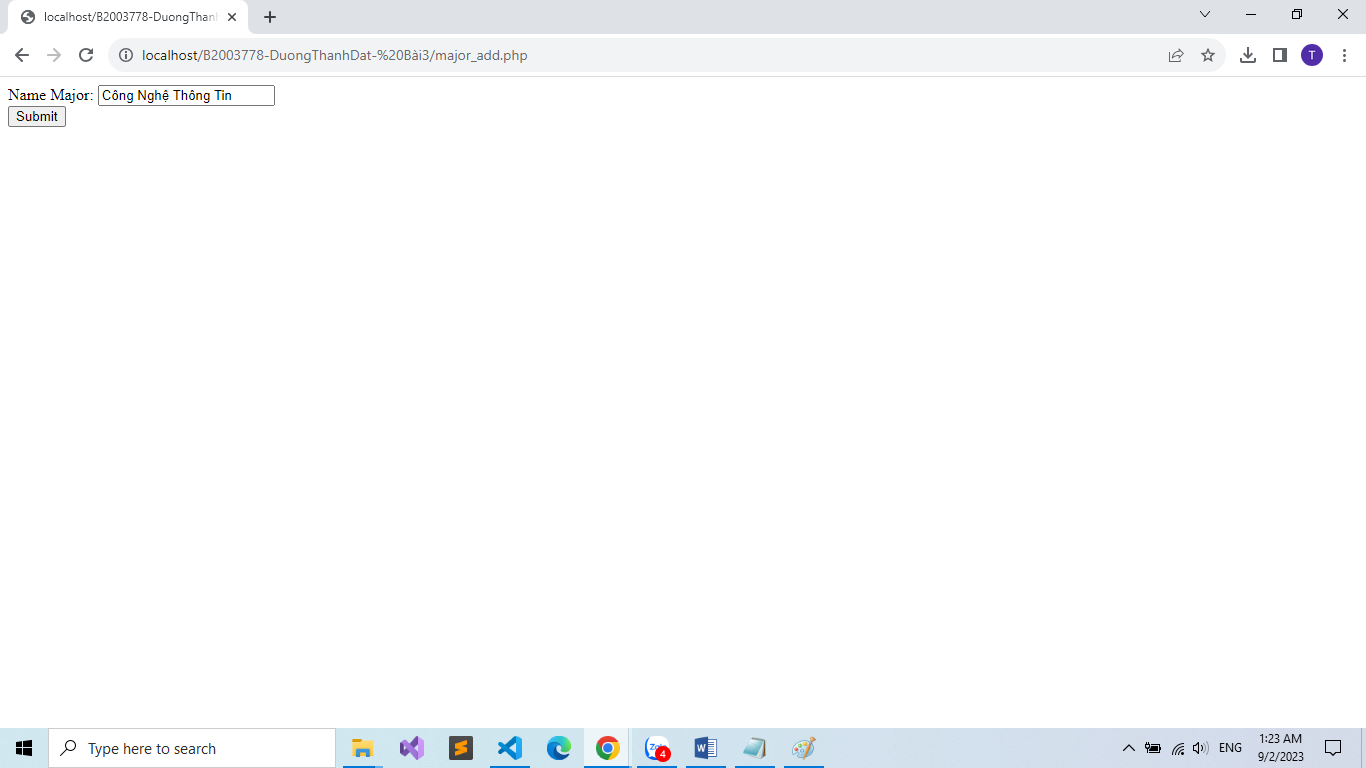


8. Tạo các tập tin major\_index.php, major\_add.php, major\_edit (và major\_edit\_save.php để thực hiện hành động sửa trên csdl), major.xoa để có các trang hiển thị danh sách, thêm, sửa, xóa đối với bảng major. Gợi ý: Tham khảo taidulieu\_bang1.php, form\_sua.php, sua.php, xoa.php

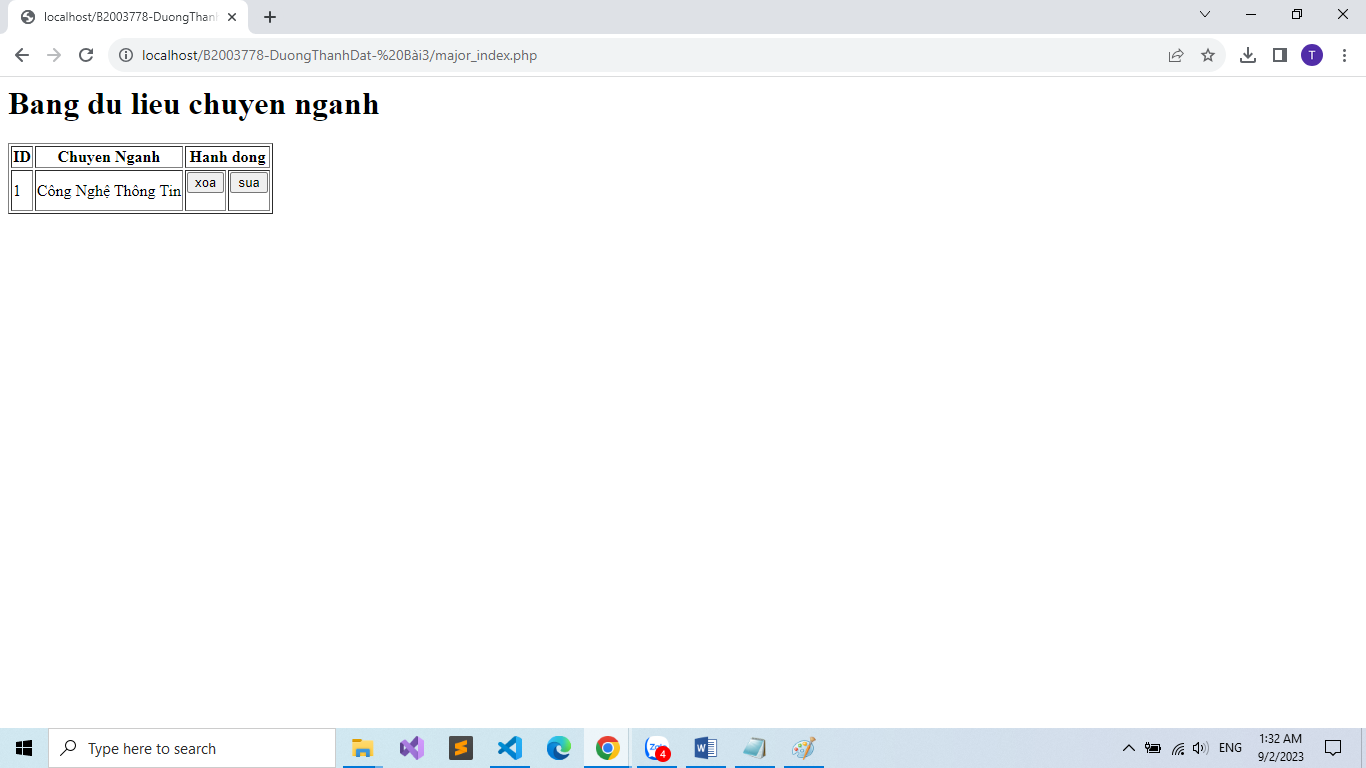
* index\_major.php



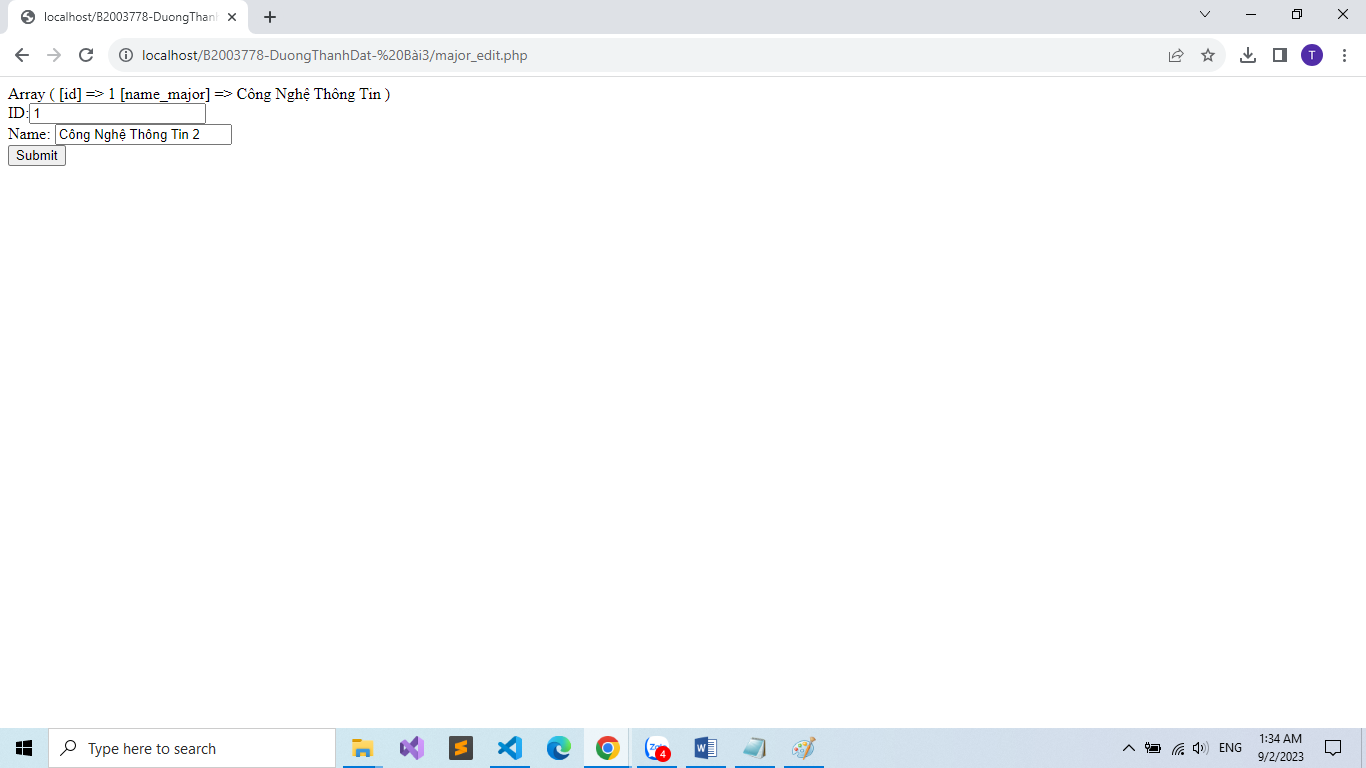
* major\_add.php



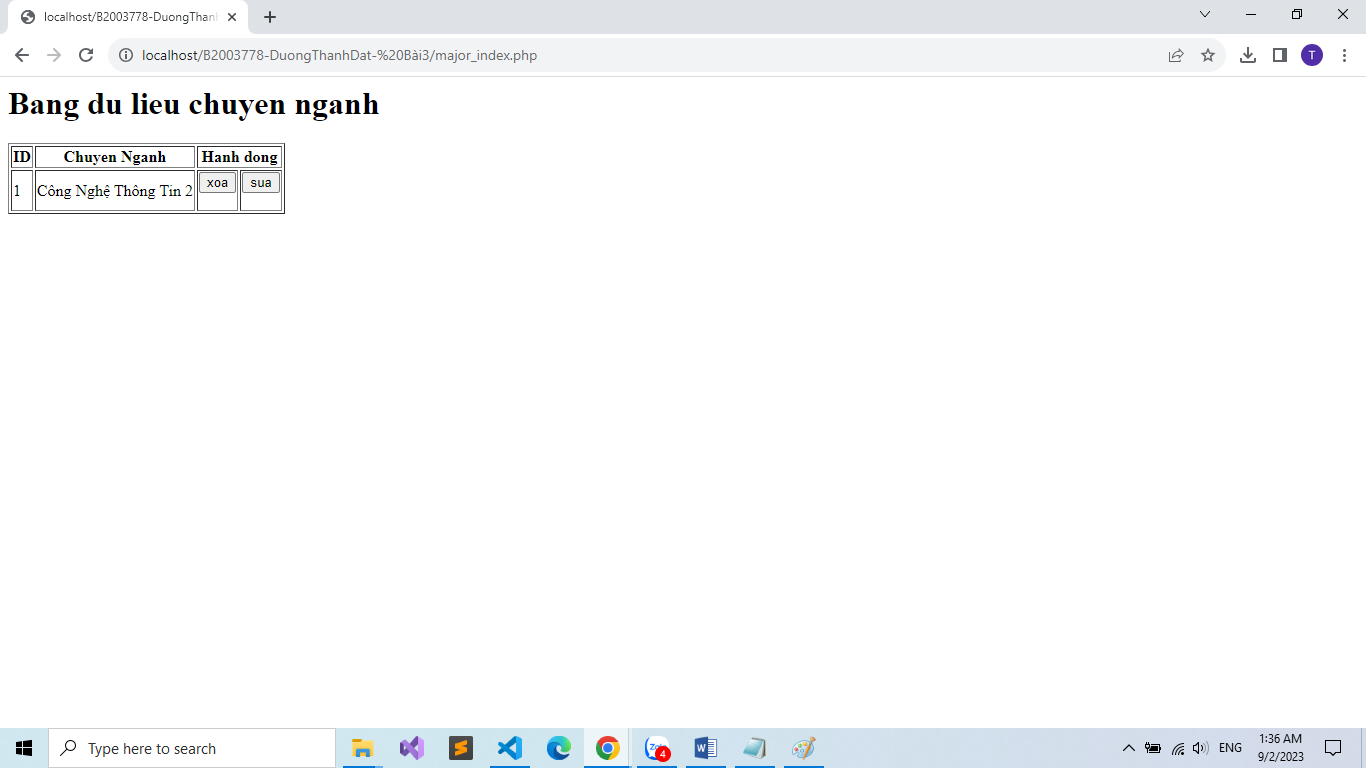
* major\_save.php



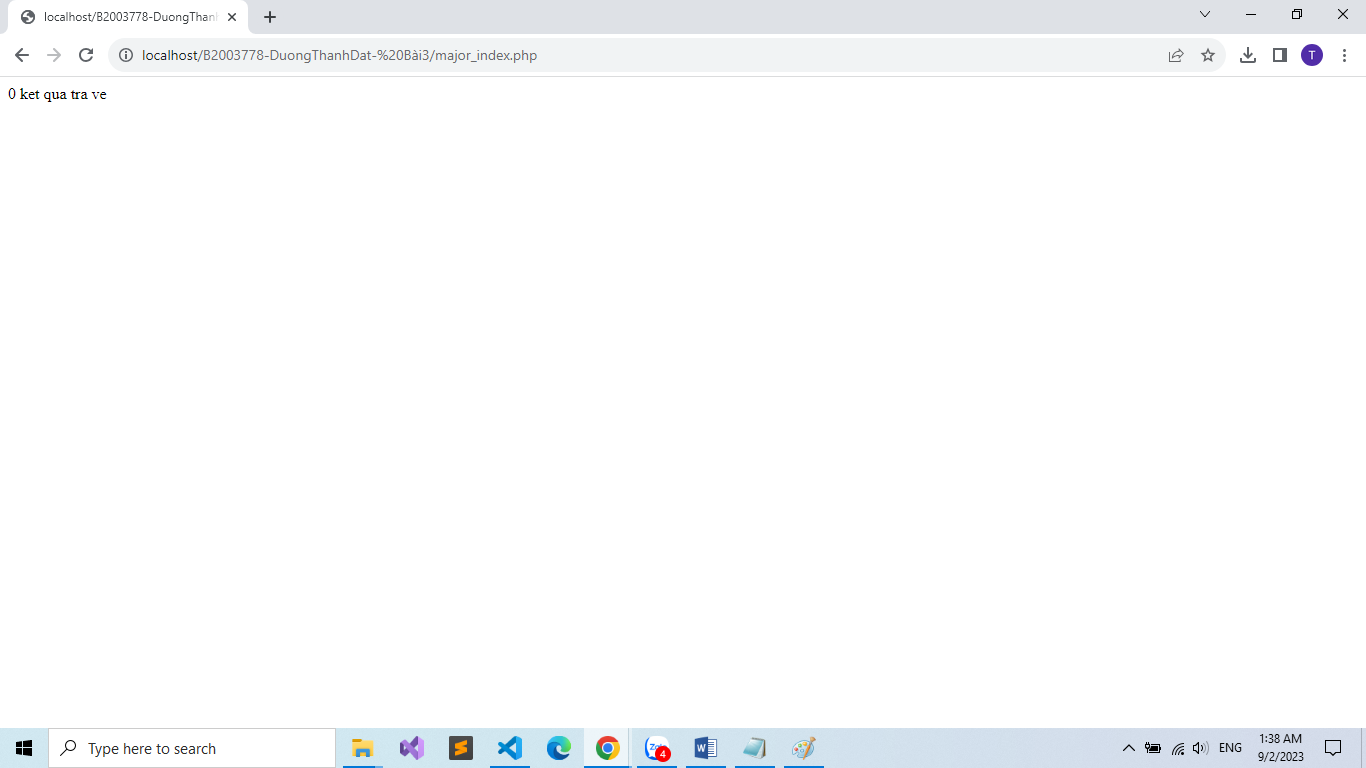
* major\_edit.php



* major\_edit\_save.php

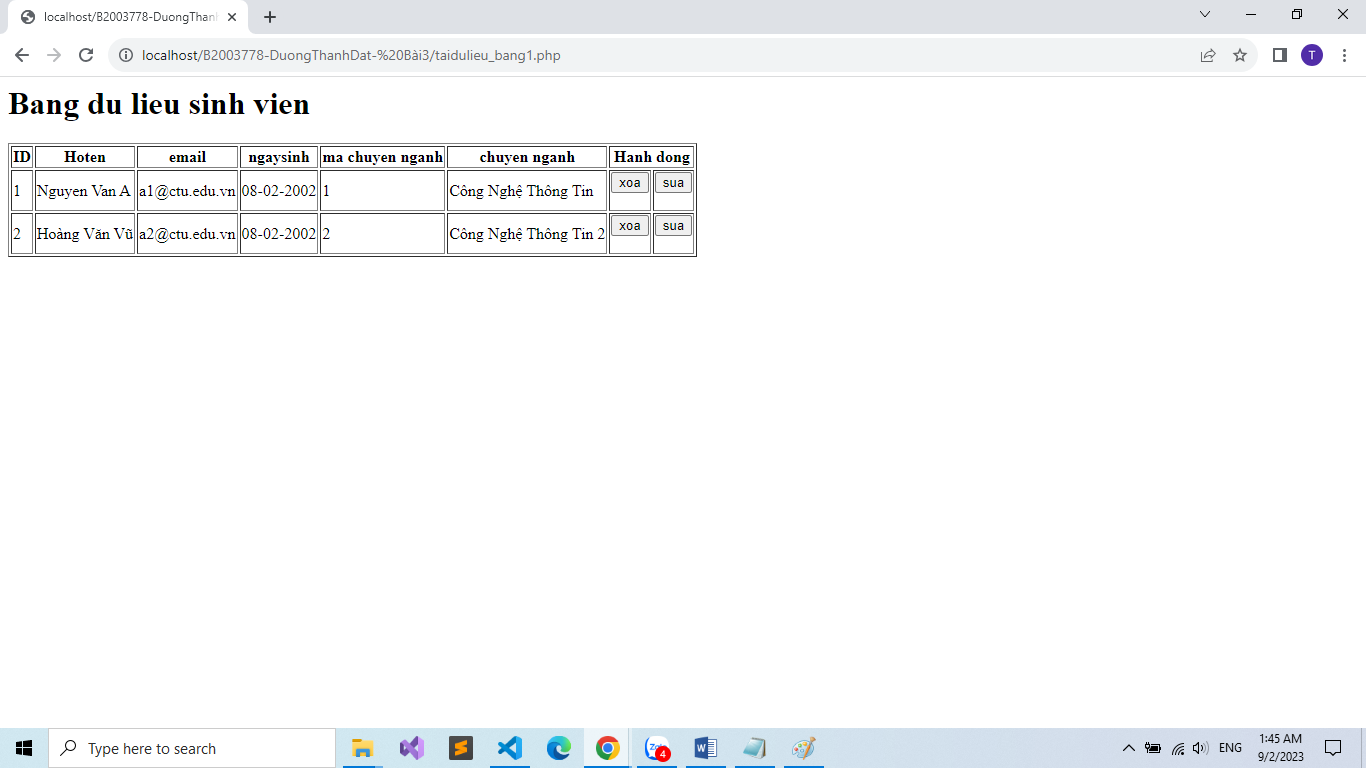


* major\_xoa.php



9. Cập nhật lại taidulieu\_bang1.php thêm hiển thị mã chuyên ngành và tên chuyên ngành tương ứng của sinh viên. Gợi ý: các bạn sửa câu lệnh sql kết nối thêm bảng chuyên ngành.

* taidulieu\_bang1.php



10. Cập nhật tương ứng ở tập tin formnhap.php, form\_sua.php sau khi thêm phần nhập khóa ngoại dạng combobox lấy dữ liệu từ bảng major. Gợi ý: Select dữ liệu từ bảng major và đưa dạng list với tag option <https://www.w3schools.com/tags/tag_option.asp>