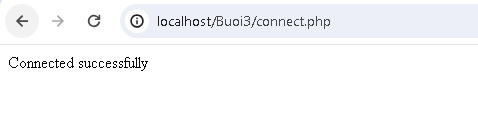
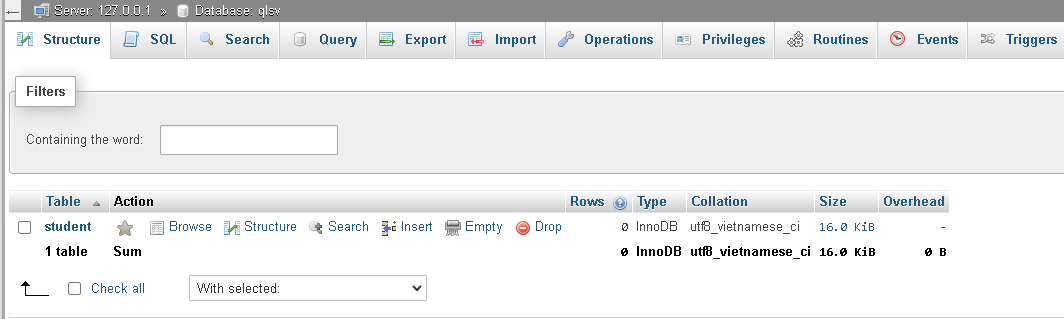
Câu 1:

* connect.php



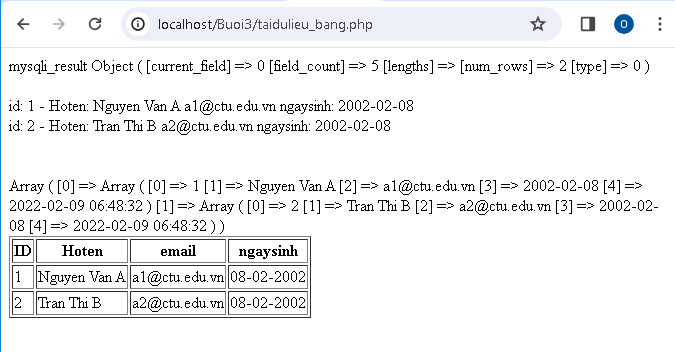
* con\_db.php



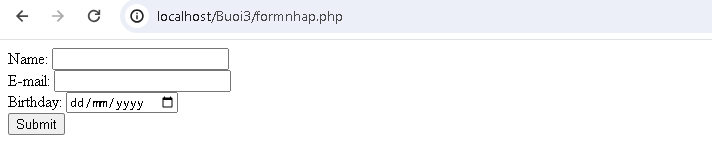


* taidulieu\_bang.php

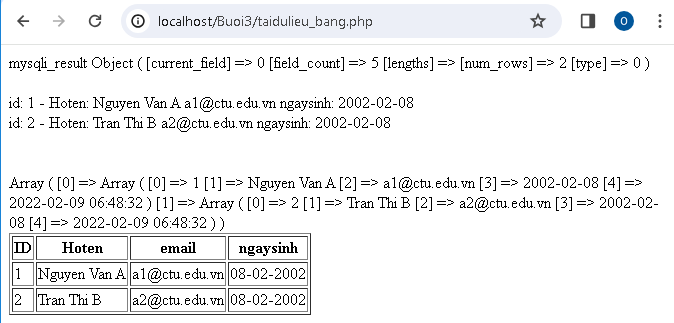




* formnhap.php



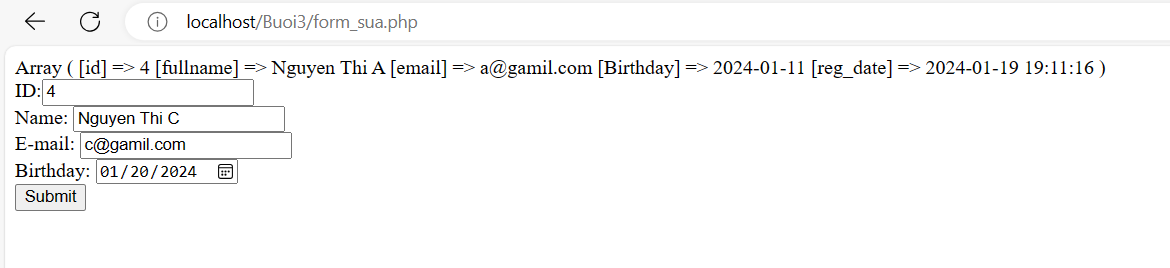
* luu.php



* taidulieu\_bang1.php



* form\_sua.php



* sua.php

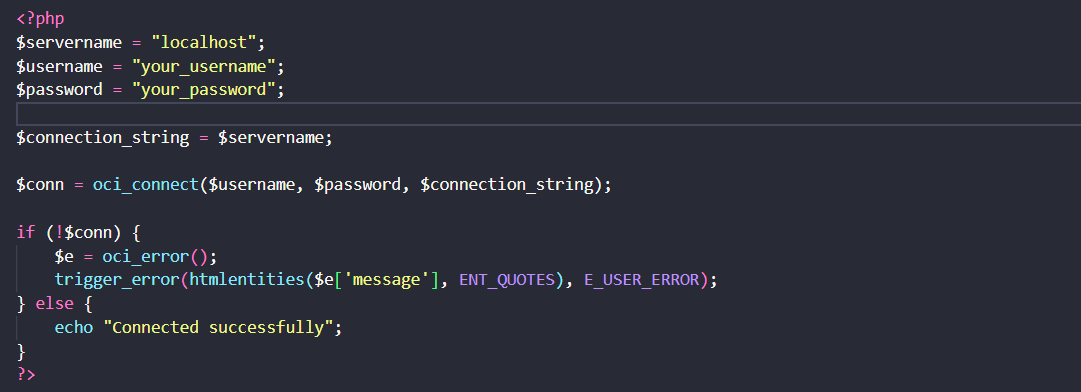


* xoa.php



Câu 2:

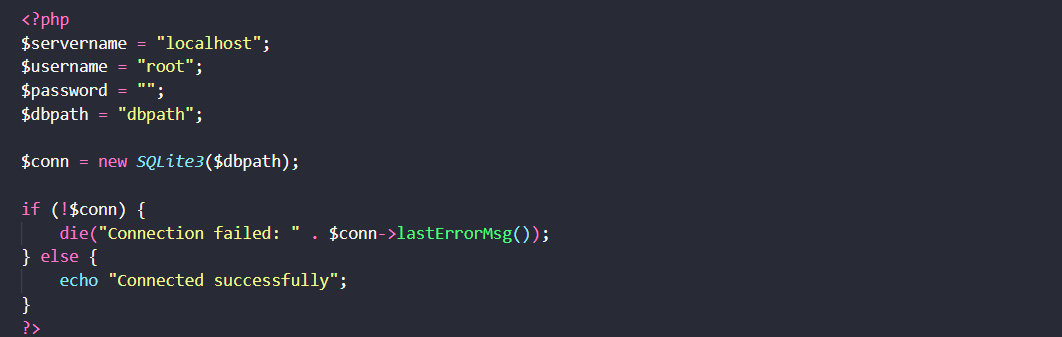
* Kết nối với Oracle



* Kết nối với SQL Server



* Kết nối với SQLite



Câu 3:

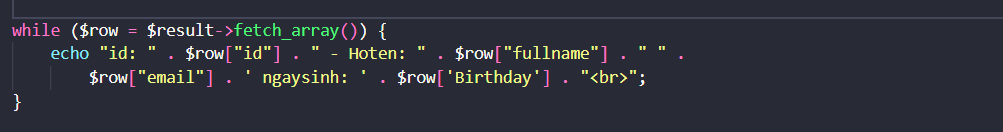
* class mysqli để hỗ trợ thực hiện kết nối đến cơ sở dữ liệu MySQL và thực hiện các thao tác truy vấn như thêm, sửa, xóa dữ liệu.
* Các phương thức trong class mysqli:
* \_\_construct(): Phương thức này dùng để khởi tạo một đối tượng mysqli
* query(): Thực hiện một truy vấn SQL trên cơ sở dữ liệu
* fetch\_assoc(): Trả về dữ liệu từ kết quả truy vấn dưới dạng mảng kết hợp, mảng số nguyên, hoặc cả hai
* fetch\_object(): Trả về dữ liệu dưới dạng đối tượng.
* num\_rows(): Trả về số hàng trong kết quả truy vấn
* affected\_rows(): Trả về số hàng ảnh hưởng bởi một truy vấn INSERT, UPDATE, hoặc DELETE
* preapare(): Chuẩn bị một truy vấn SQL để thực hiện nhiều lần với các tham số khác nhau
* error: Thuộc tính chứa thông tin về lỗi sau khi một truy vấn thất bại
* autocommit(): Bật/tắt chế độ tự động commit
* commit(): Commit giao dịch hiện tại
* rollback(): Rollback giao dịch hiện tại
* get\_server\_info(): Trả về thông tin về phiên bản MySQL Server
* ping(): Kiểm tra xem kết nối đến server có còn hoạt động không.
* info(): Thuộc tính chứa thông tin về trạng thái thực hiện của truy vấn.
* close(): Đóng kết nối đến MySQL Server
* connect(): Kết nối đến cơ sở dữ liệu MySQL
* real\_escape\_string(): Nhận một chuỗi và trả về một phiên bản đã được xử lý để tránh các lỗi SQL injection
* insert\_id(): Trả về ID tự động được tạo bởi câu lệnh INSERT gần nhất
* 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\_row(): Trả về một mảng chứa dữ liệu từ hàng tiếp theo trong kết quả, chỉ sử dụng kiểu số nguyên
* fetch\_field(): Trả về thông tin về một cột từ kết quả truy vấn dưới dạng đối tượng
* errno(): Trả về mã lỗi của lần gọi phương thức gần nhất
* init(): Tạo một đối tượng MySQLi mới và trả về nó. Được sử dụng để khởi tạo một đối tượng MySQLi không cần kết nối ngay lập tức
* real\_connect(): Kết nối đến server MySQL với các tham số được cung cấp
* change\_user(): Thay đổi người dùng và cơ sở dữ liệu kết nối đến MySQL Server
* select\_db(): Chọn cơ sở dữ liệu làm cơ sở dữ liệu hiện tại
* set\_charset(): Thiết lập bộ ký tự cho kết nối
* multi\_query(): Thực hiện nhiều truy vấn SQL trong một lời gọi

Câu 4:

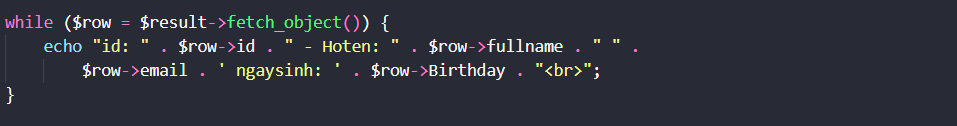
* Ý nghĩa của phương thức die(): được sử dụng để kết thúc chương trình và hiển thị một thông báo lỗi tùy chọn. Thường được sử dụng để xử lý các tình huống không mong muốn hoặc lỗi trong mã nguồn PHP.
* Cách dùng: Nếu kết nối không thành công, hàm die() sẽ được gọi và thông báo “Connection failed” sẽ được hiển thị và chương trình sẽ kết thúc ngay lập tức mà không thực hiện các lệnh tiếp theo. Điều này giúp ngăn chặn việc thực hiện các lệnh không mong muốn trong trường hợp kết nối thất bại.

Câu 5:

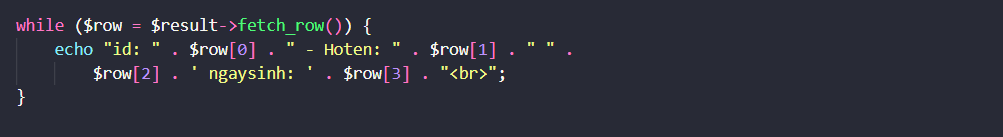
* Các phương thức khác trong mysqli có chức năng tương tự
* fetch\_object(): Trả về dữ liệu dưới dạng đối tượng.
* fetch\_array(): Trả về một mảng chứa dữ liệu từ hàng tiếp theo trong kết quả, với cả hai kiểu kết hợp và số nguyên
* fetch\_row(): Trả về một mảng chứa dữ liệu từ hàng tiếp theo trong kết quả, chỉ sử dụng kiểu số nguyên
* Phương thức fetch\_array()



* Phương thức fetch\_object()



* Phương thức fetch\_row()



Câu 6:

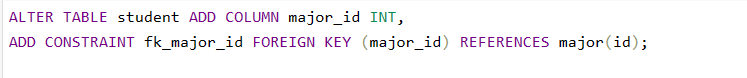
* Hàm header trong PHP được sử dụng để gửi một tiêu đề HTTP tới trình duyệt, thường được sử dụng để chuyển hướng người dùng từ một trang web tới một trang web khác. Điều này thường được thực hiện sau khi một hành động cụ thể, ví dụ như submit một form hoặc xử lý dữ liệu, để chuyển hướng người dùng sang một trang web mới để tương tác tiếp hoặc hiển thị một thông báo thành công.
* Cách sử dụng hàm header:
* Sau khi thực hiện thành công câu lệnh SQL để chèn dữ liệu vào cơ sở dữ liệu ($conn->query($sql) == TRUE), thông báo "Them sinh vien thanh cong" được hiển thị.
* Tiếp theo, câu lệnh header('Location: taidulieu\_bang.php'); được sử dụng để chuyển hướng người dùng tới trang taidulieu\_bang.php. Khi server gửi tiêu đề Location với giá trị là URL mới, trình duyệt sẽ tự động chuyển hướng người dùng tới trang đó.
* Nếu có lỗi trong quá trình thực hiện câu lệnh SQL, thông báo lỗi cùng với thông báo "Error" sẽ được hiển thị.
* Chức năng chính của header('Location: ...'); là chuyển hướng người dùng, và nó phải được gọi trước khi có bất kỳ đầu ra nào khác được gửi đến trình duyệt. Nếu đã có đầu ra, sẽ xuất hiện lỗi.

Câu 7:

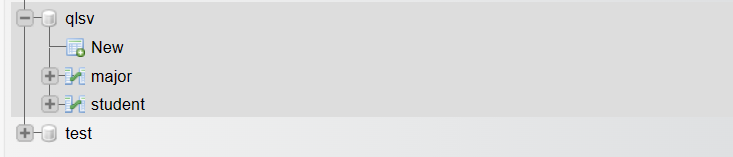
* Tạo bảng



* Thêm khóa ngoại

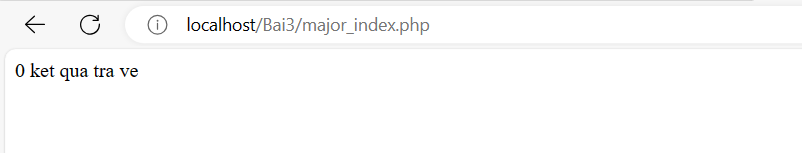


* Kết quả

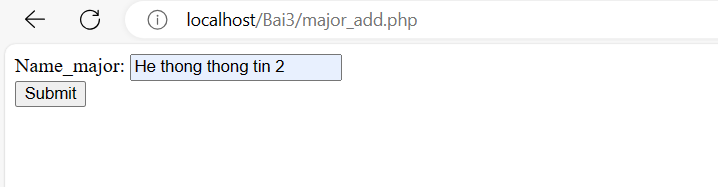


Câu 8:

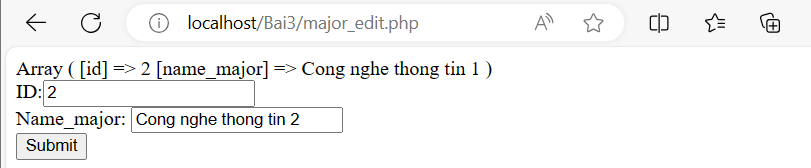
* major\_index.php



* major\_add.php



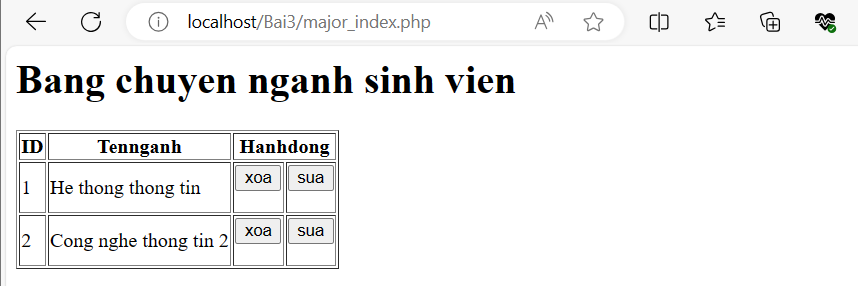
* major\_edit.php



* major\_edit\_save.php

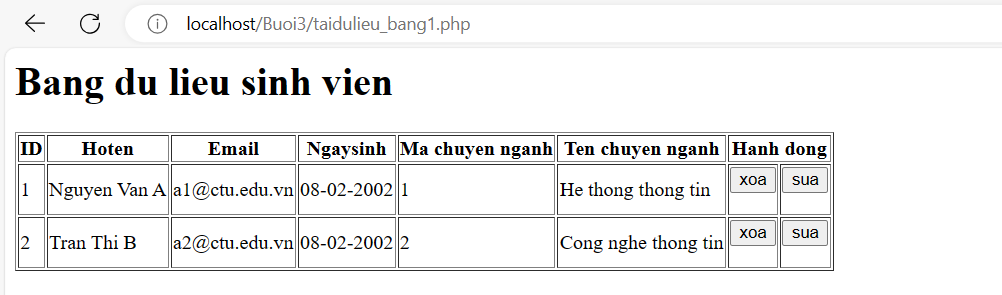


* major\_xoa.php



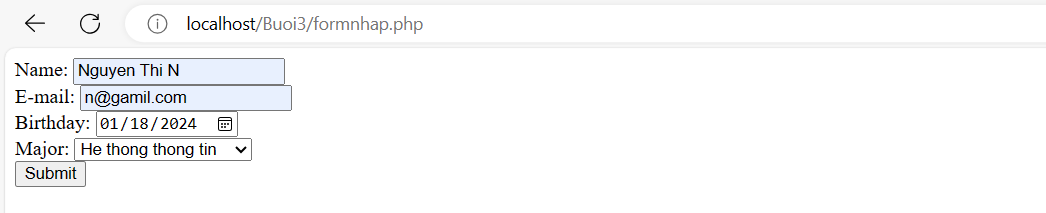
Câu 9:

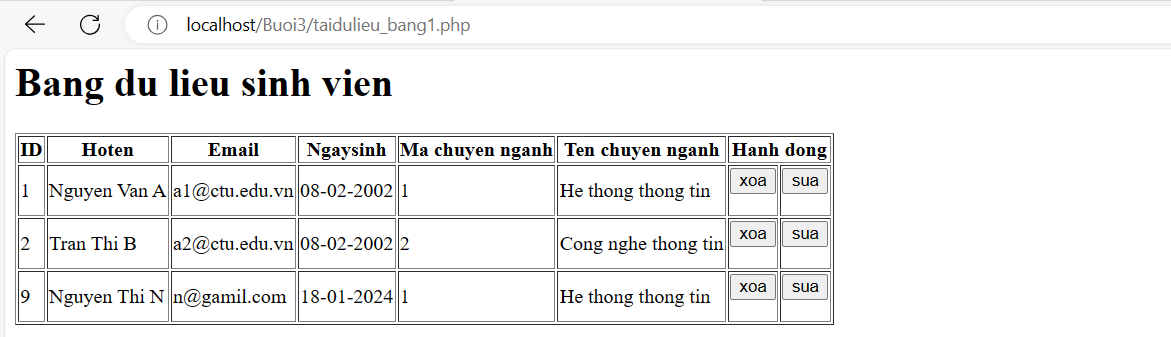
* taidulieu\_bang1.php



Câu 10:

* formnhap.php





* formsua.php

