**Bài tập thực hành 03**

Lập trình Cơ sở dữ liệu với PHP

**Mục tiêu cần đạt**:

Sau buổi các bạn cần nắm thao tác PHP với Cơ sở dữ liệu (CSDL):

* Biết cách tạo chuỗi kết nối đến Mysql, và kết nối với 1 CSDL
* Tạo 1 form cơ bản để nhập liệu và lưu vào CSDL
* Hiển thị dữ liệu từ bảng/view trong CSDL lên giao diện web (trang index)
* Thực hiện thêm/sửa/xóa dữ liệu trên giao diện web

Các bạn đọc và thực hiện viết code như hướng dẫn để làm quen, sau đó đọc trả lời các **Yêu cầu** ở dưới bài hướng dẫn để thực hiện yêu cầu bài thực hành.

**Gợi ý Tham khảo:**

* <https://www.w3schools.com/php/>
* và các nguồn khác mà sinh viên tìm được

## Tạo chuỗi kết nối đến mysql

Khi thực hiện kết nối csdl, chúng ta cần chú ý ghi nhớ tên *username* để kết nối. Thông thường thì người dùng root khi mới cài đặt xampp sẽ chưa đặt password. Khi triển khai thực tế, chúng ta nên tạo các người dùng riêng biệt với mật khẩu cho csdl này.

Tạo tập tin connect.php với các lệnh php sau để thực hiện chuỗi kết nối đến mysql:

(Tham khảo các phương thức trong class mysqli tại [[1]](#footnote-1))

<?php

// thong tin ve chuoi ket noi gom server name, username va mat khau de dang nhap vao mysql, mac dinh cua xampp la root, password rong

$servername = "localhost";

$username = "root";

$password = "";

// Create connection

$conn = new mysqli($servername, $username, $password);

// Check connection

if ($conn->connect\_error) {

//hien thi loi neu ket noi khong duoc

die("Connection failed: " . $conn->connect\_error);

}

//neu ket noi thanh cong

echo "Connected successfully";

?>

## Tạo chuỗi kết nối đến CSDL

Bạn vào phpmyadmin tạo 1 csdl tên là **qlsv**

Tạo 1 tập tin php con\_db.php với lệnh như sau

<?php

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "qlsv";

// Create connection

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

// Check connection

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

// sql to create table

$sql = "CREATE TABLE student (

id INT(6) UNSIGNED AUTO\_INCREMENT PRIMARY KEY,

fullname VARCHAR(50) NOT NULL,

email VARCHAR(50),

Birthday date,

reg\_date TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP

)";

if ($conn->query($sql) === TRUE) {

echo "Table student created successfully";

} else {

echo "Error creating table: " . $conn->error;

}

$conn->close();

?>

Chạy tập tin php trên, và quan sát trong phpmyadmin các bạn sẽ thấy bảng student được tạo trong qlsv

Trong trường hợp, chúng ta muốn tách biệt tạo csdl bên phpmyadmin, chúng ta chỉ cần chuỗi kết nối đến csdl là đủ.

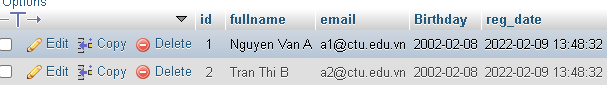
## Index: Tải dữ liệu từ bảng trong csdl đưa lên trình bày trên trang web

Viết vài lệnh sau dán vào phần SQL trong phpmyadmin để tạo 1 số dòng dữ liệu cho bảng student.

INSERT INTO `student` (`id`, `fullname`, `email`, `Birthday`, `reg\_date`) VALUES

(1, 'Nguyen Van A', 'a1@ctu.edu.vn', '2002-02-08', '2022-02-09 06:48:32'),

(2, 'Tran Thi B', 'a2@ctu.edu.vn', '2002-02-08', '2022-02-09 06:48:32');



Kết nối CSDL truy cập vào bảng student đưa dữ liệu bảng vào mảng $st

Tạo tập tin taidulieu\_bang.php. Ở đây trình bày các cách để thể hiện dữ liệu, bao gồm cách trình bày dữ liệu dạng bảng kết hợp HTML. Chúng ta chú ý kiểu dữ liệu dạng date sẽ cần chỉnh sửa format lại để hiển thị theo định dạng phù hợp.

<?php

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "qlsv";

// Create connection

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

// Check connection

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

//tao chuoi luu cau lenh sql

$sql = "SELECT \* FROM student";

//thuc thi cau lenh sql va dua doi tuong vao $result

$result = $conn->query($sql);

if ($result->num\_rows > 0) {

//cach 1: show du lieu nhu bien

//show gia tri trong mang

print\_r ($result);

echo '<br>';

echo '<br>';

//Cach 2: show theo tung dong voi for

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

echo "id: " . $row["id"]. " - Hoten: " . $row["fullname"]. " " . $row["email"].' ngaysinh: '.$row['Birthday']. "<br>";

}

echo '<br>';

echo '<br>';

//xoa ket qua cu tu o tren

$result -> free\_result();

//Cach 3: trinh bay voi bang html

//load du lieu moi len dua vao bien result

$result = $conn->query($sql);

$result\_all = $result -> fetch\_all();

print\_r($result\_all);

// trinh bay du lieu trong 1 bang html

//tieu de bang

echo "<table border=1><tr><th>ID</th><th>Hoten</th><th>email</th><th>ngaysinh</th></tr>";

// output data of each row

foreach ($result\_all as $row) {

//dinh dang de hien thi ngay thang theo dd-mm-yyyy

$date = date\_create($row[3]);

echo "<tr><td>" . $row[0]. "</td><td>" . $row[1]. "</td><td>" . $row[2]. "</td><td>" .

$date ->format('d-m-Y')

. "</td></tr>";

}

echo "</table>";

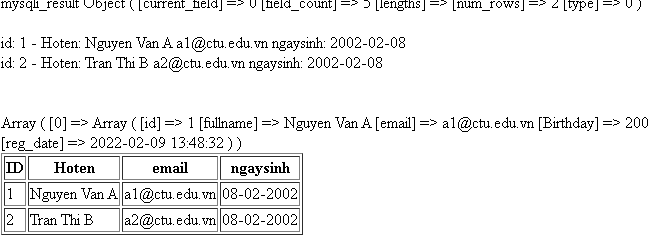
} else {

echo "0 ket qua tra ve";

}

$conn->close();

?>



## Add: Form để thêm dữ liệu vào CSDL và chuyển đến trang thể hiện dữ liệu

Tạo form giao diện để nhập: formnhap.php

<!DOCTYPE HTML>

<html>

<body>

<form action="luu.php" method="post">

Name: <input type="text" name="name"><br>

E-mail: <input type="text" name="email"><br>

Birthday: <input type="date" name="birth"><br>

<input type="submit">

</form>

</body>

</html>

Tạo tập tin php xử lý hành động lưu: luu.php

<?php

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "qlsv";

// Create connection

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

// Check connection

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

$date = date\_create($\_POST["birth"]);

$sql = "INSERT INTO student (fullname, email, birthday) VALUES ('".$\_POST["name"] ."', '".$\_POST["email"] ."', '".$date ->format('Y-m-d') ."')";

if ($conn->query($sql) == TRUE) {

echo "Them sinh vien thanh cong";

//neu thuc hien thanh cong, chung ta se cho di chuyen den taidulieu\_bang.php

header('Location: taidulieu\_bang.php');

} else {

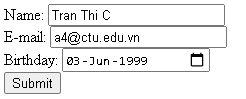
echo "Error: " . $sql . "<br>" . $conn->error;

}

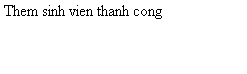
$conn->close();

?>

Các bạn chạy formnhap.php trên trình duyệt và thấy kết quả sau:



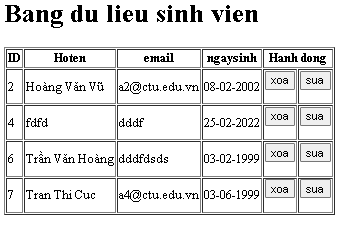
Bấm submit và chúng ta đã nhập liệu thành công



## Edit: Chỉnh sửa dữ liệu từ danh sách

Kết nối CSDL truy cập vào bảng student đưa dữ liệu bảng vào mảng $st

Tạo tập tin taidulieu\_bang1.php (như taidulieu\_bang.php nhưng gọn hơn và có thêm nút Xóa/sửa kế bên).



Code để thực hiện giao diện trên trong taidulieu\_bang1.php

<?php

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "qlsv";

// Create connection

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

// Check connection

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

$sql = "SELECT \* FROM student";

$result = $conn->query($sql);

if ($result->num\_rows > 0) {

// trinh bay voi bang html

//load du lieu moi len dua vao bien result

$result = $conn->query($sql);

$result\_all = $result -> fetch\_all(MYSQLI\_ASSOC);

//print\_r($result\_all);

// trinh bay du lieu trong 1 bang html

//tieu de bang

?>

<h1>Bang du lieu sinh vien</h1>

<table border=1><tr><th>ID</th><th>Hoten</th><th>email</th><th>ngaysinh</th><th colspan="2">Hanh dong</th></tr>

<?php

// output data of each row

foreach ($result\_all as $row) {

$date = date\_create($row['Birthday']);

echo "<tr><td>" . $row["id"]. "</td><td>" . $row["fullname"]. "</td><td>" . $row["email"]. "</td><td>" .

$date ->format('d-m-Y')

. "</td><td>";

?>

<form method="post" action="xoa.php">

<input type="submit" name="action" value="xoa"/>

<input type="hidden" name="id" value="<?php echo $row['id']; ?>"/>

</form>

<?php

echo "</td>";

echo "<td>";

?>

<form method="post" action="form\_sua.php">

<input type="submit" name="action" value="sua"/>

<input type="hidden" name="id" value="<?php echo $row['id']; ?>"/>

</form>

<?php

echo "</td></tr>";

}

echo "</table>";

} else {

echo "0 ket qua tra ve";

}

$conn->close();

?>

Khi chúng ta chọn sửa dữ liệu, tập tin sau (form\_sua.php) sẽ thực thi

<!DOCTYPE HTML>

<html>

<?php

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "qlsv";

// Create connection

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

// Check connection

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

$id = $\_POST['id'];

$sql = "select \* FROM student WHERE ID='".$id."'";

$result = $conn->query($sql);

$row = $result->fetch\_assoc();

?>

<body>

<?php print\_r($row)?>

<form action="sua.php" method="post">

ID:<input type="text" name="id" value="<?php echo $row['id'];?>"><br>

Name: <input type="text" name="fullname" value="<?php echo $row['fullname'];?>"><br>

E-mail: <input type="text" name="email" value="<?php echo $row['email'];?>"><br>

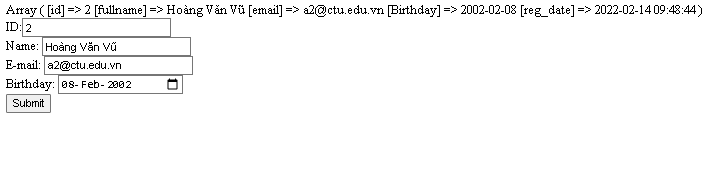
Birthday: <input type="date" name="birth" value="<?php echo $row['Birthday'];?>"><br>

<input type="submit">

</form>

</body>

</html>



Khi bấm nút submit sẽ gọi qua tập tin sua.php sau:

<?php

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "qlsv";

// Create connection

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

// Check connection

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

$id = $\_POST['id'];

$date = date\_create($\_POST["birth"]);

$sql = "UPDATE student set fullname = '".$\_POST['fullname']."', email = '".$\_POST['email']."',birthday = '".$date ->format('Y-m-d')."'";

$sql = $sql. " WHERE ID='".$id."'";

if ($conn->query($sql) == TRUE) {

header('Location: taidulieu\_bang1.php');

} else {

echo "Error: " . $sql . "<br>" . $conn->error;

}

$conn->close();

?>

Sau khi thực hiện sửa xong, sẽ tải trở lại file taidulieu\_bang1.php

## Delete: Xóa dòng dữ liệu trong bảng

Giữ nguyên tập tin như taidulieu\_bang1.php, chúng ta tạo tập tin xoa.php để xử lý khi người dùng bấm vào nút xóa tương ứng với dòng dữ liệu trong bảng

<?php

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "qlsv";

// Create connection

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

// Check connection

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

$id = $\_POST['id'];

$sql = "DELETE FROM student WHERE ID='".$id."'";

if ($conn->query($sql) == TRUE) {

header('Location: taidulieu\_bang1.php');

} else {

echo "Error: " . $sql . "<br>" . $conn->error;

}

$conn->close();

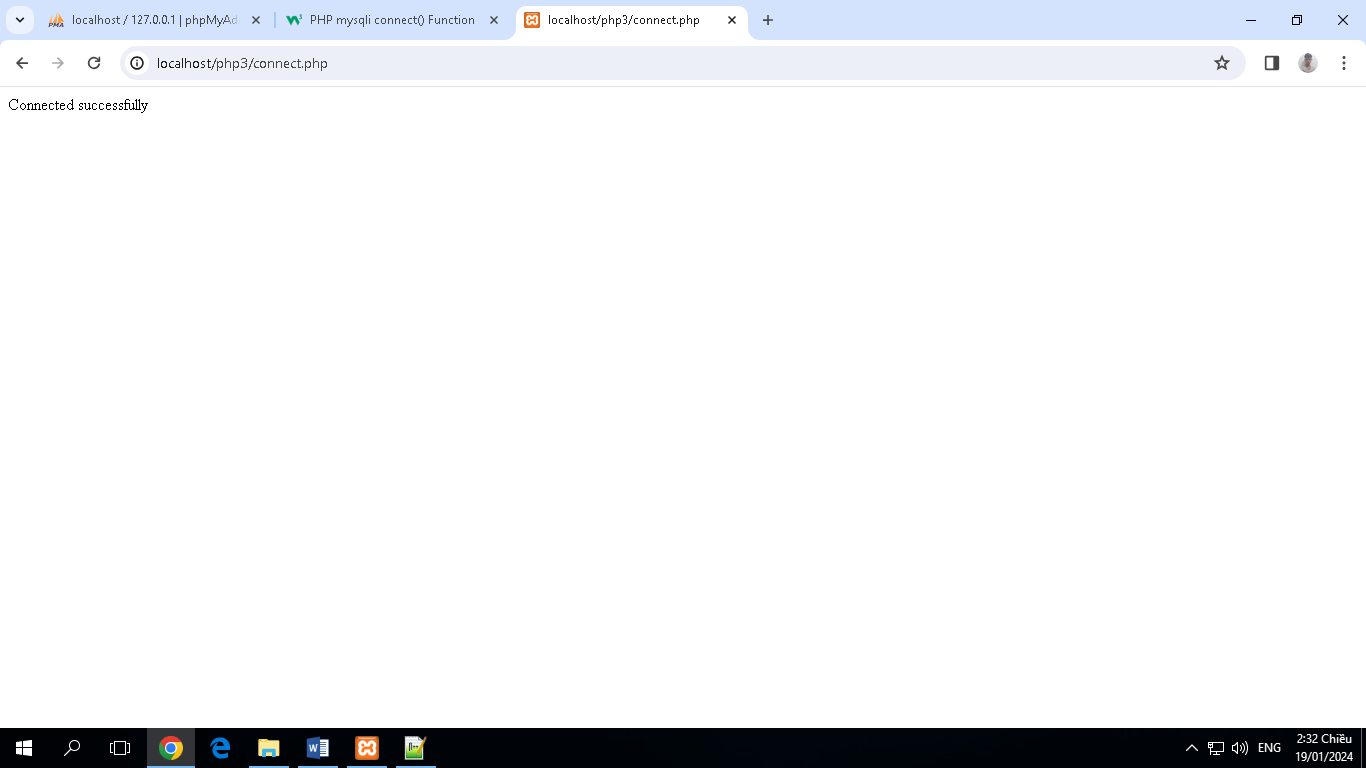
?>

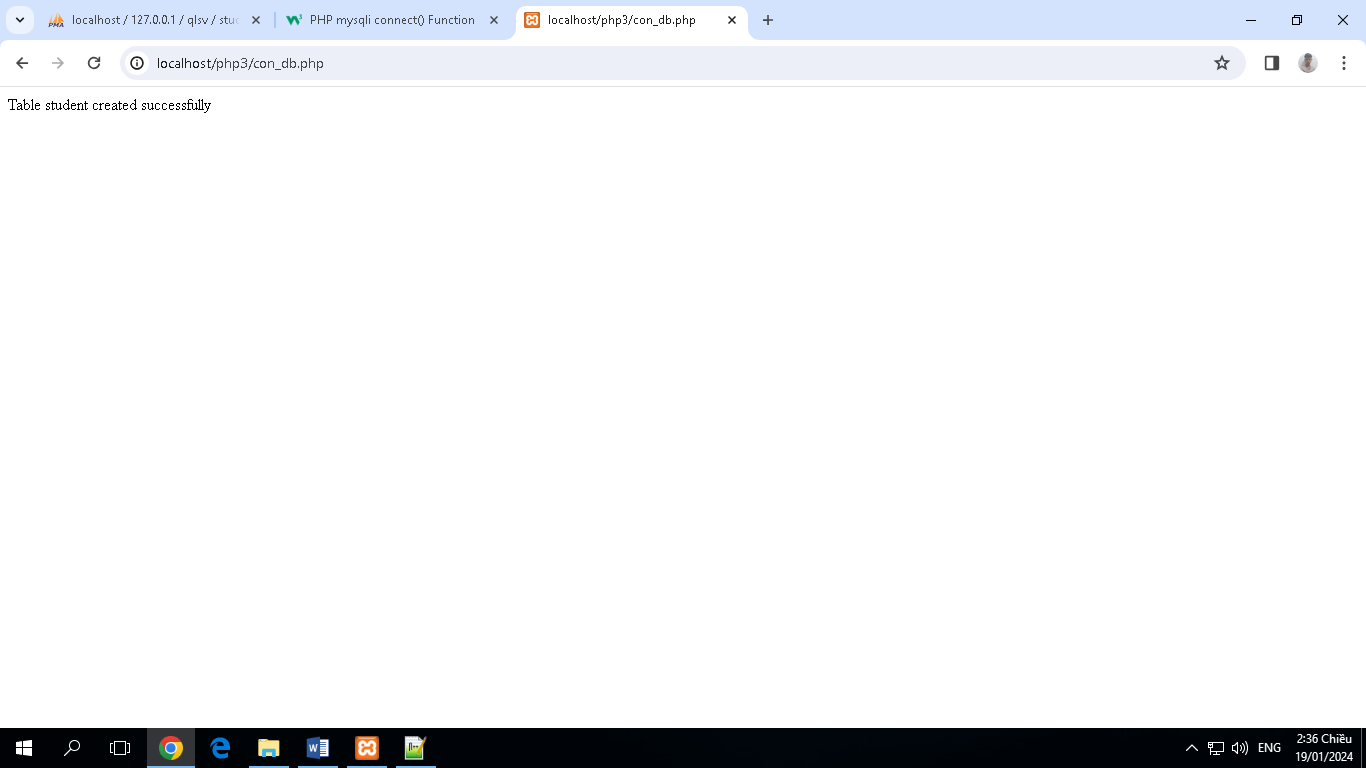
Sau khi xóa xong sẽ trở lại taidulieu\_bang1.php

# Yêu cầu bài thực hành:

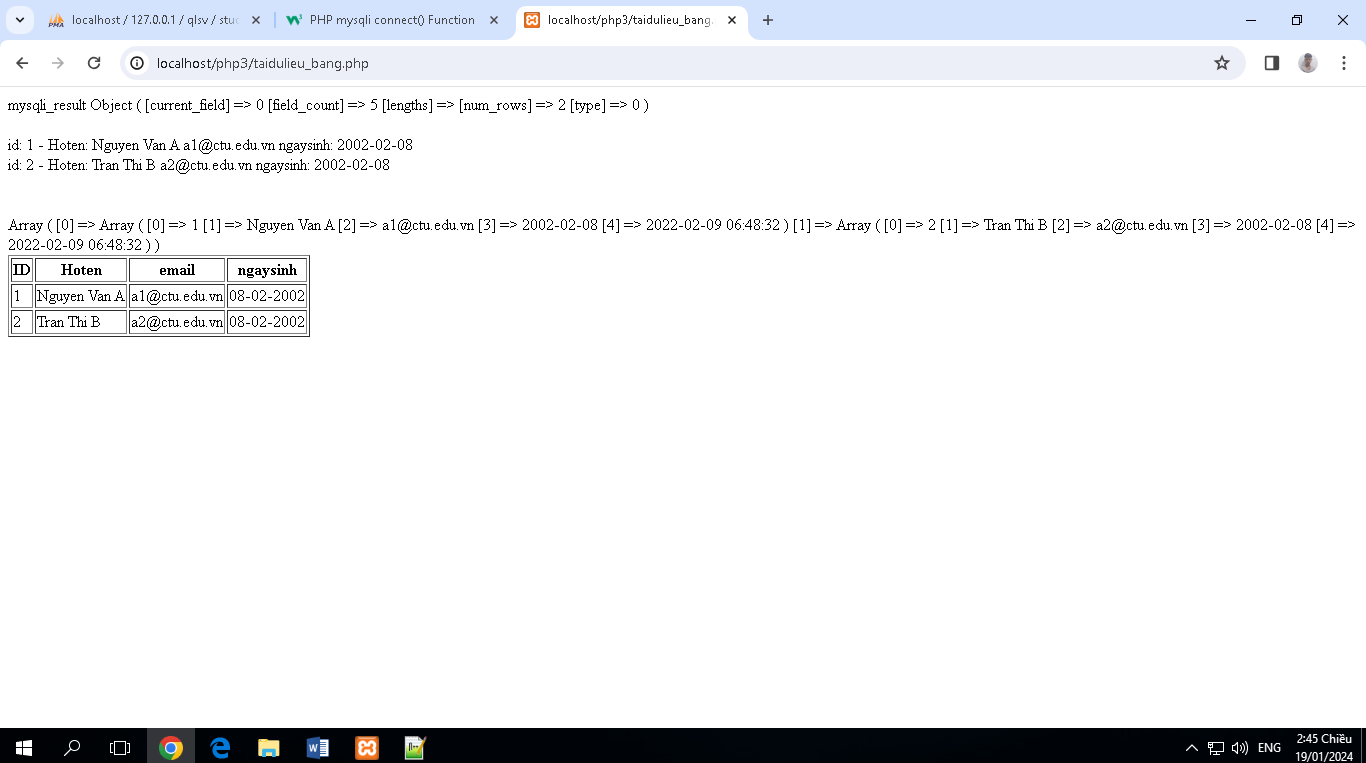
1. Bạn hãy chạy tất cả các lệnh hướng dẫn ở trên và chụp lại màn hình kết quả.

Run tập tin connect.php :

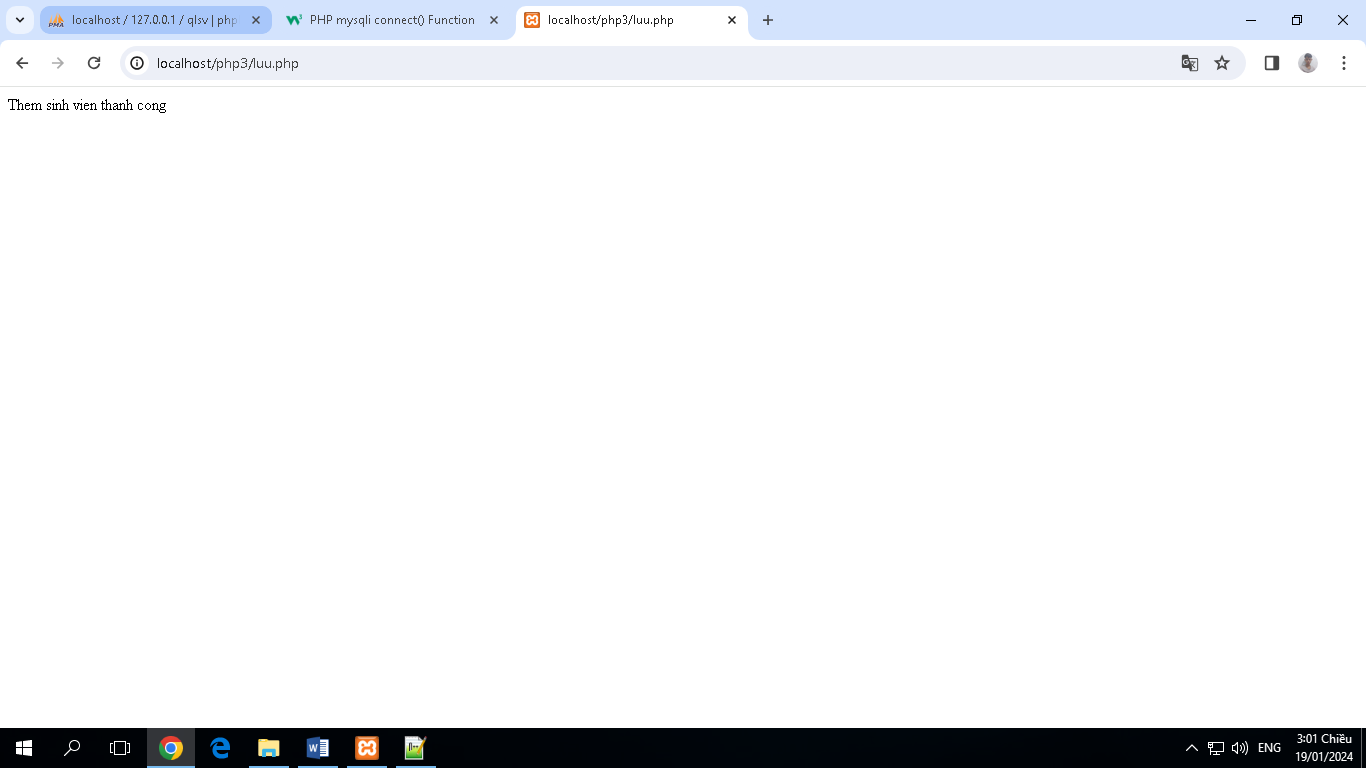
  
Run tập tin php con\_db.php :

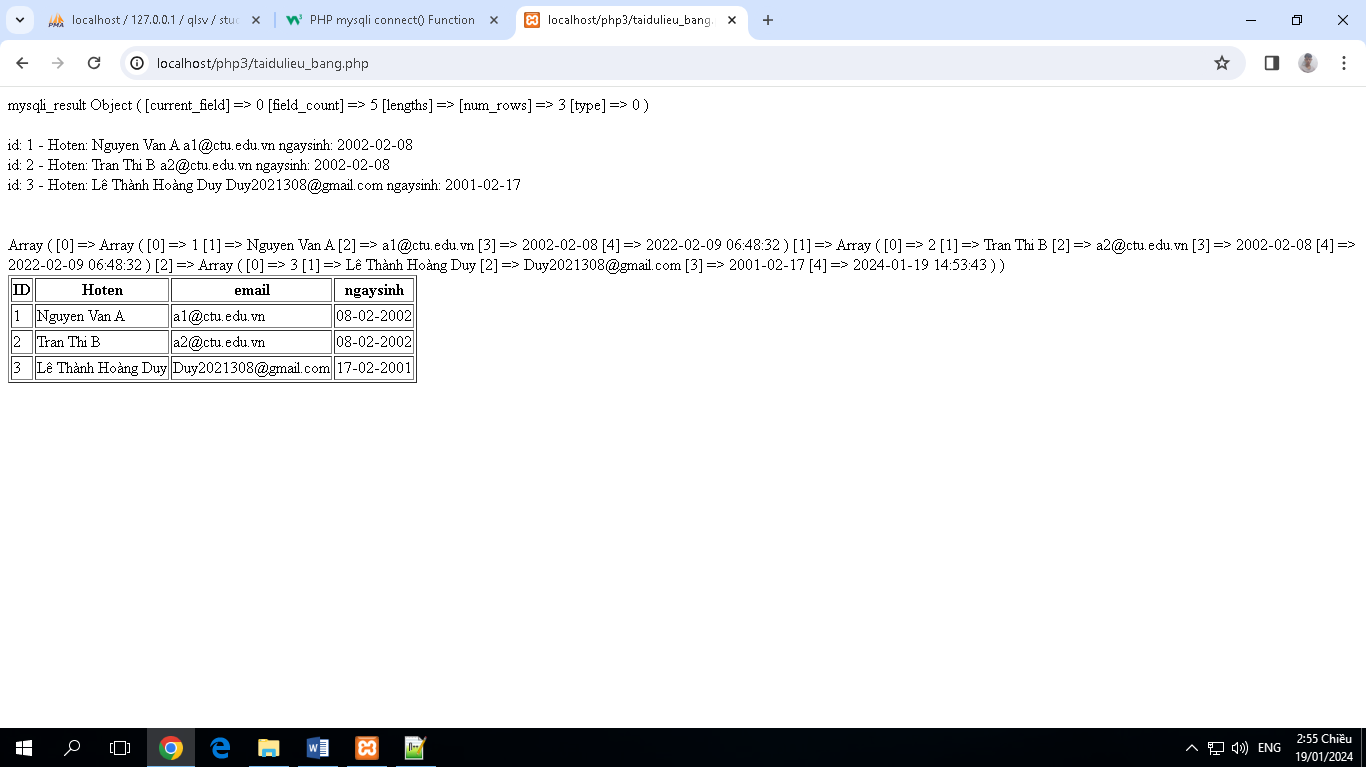


Run tập tin taidulieu\_bang.php

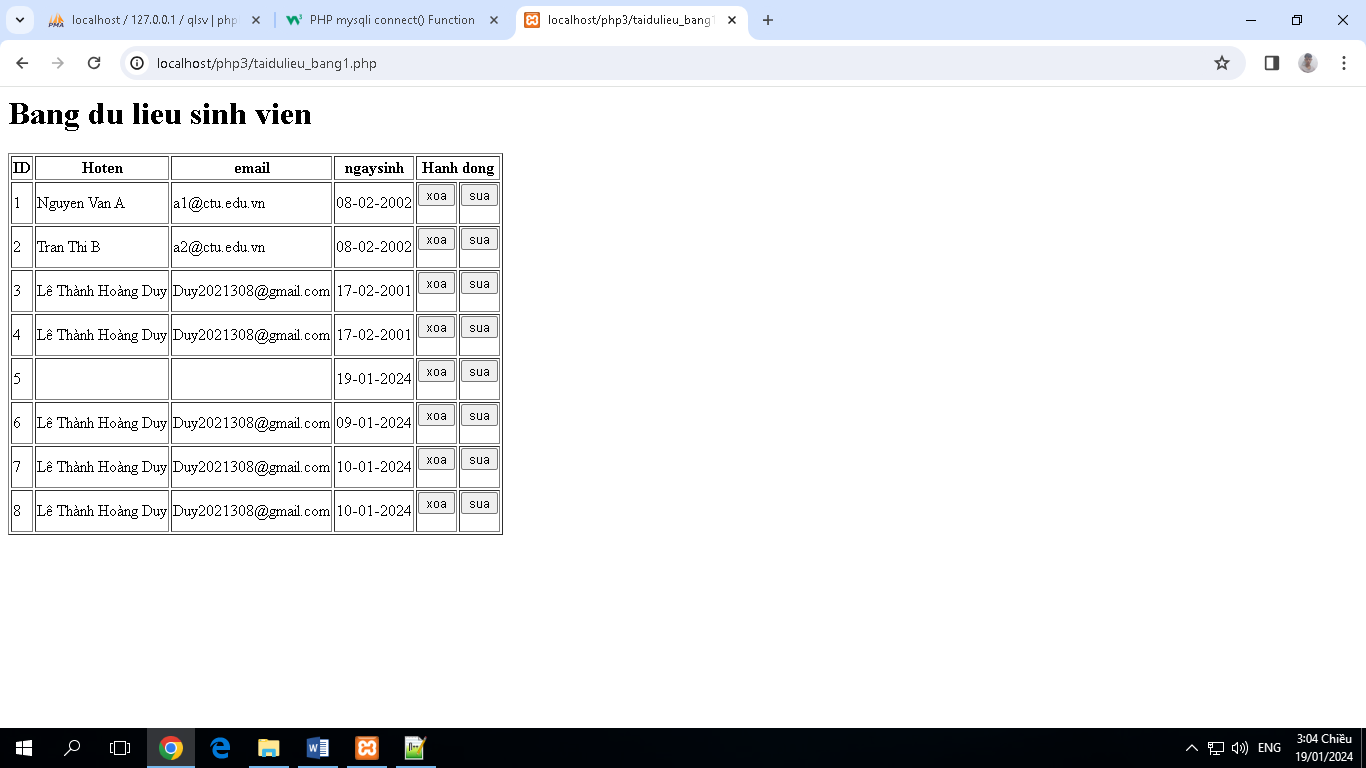


Run form giao diện để nhập: formnhap.php

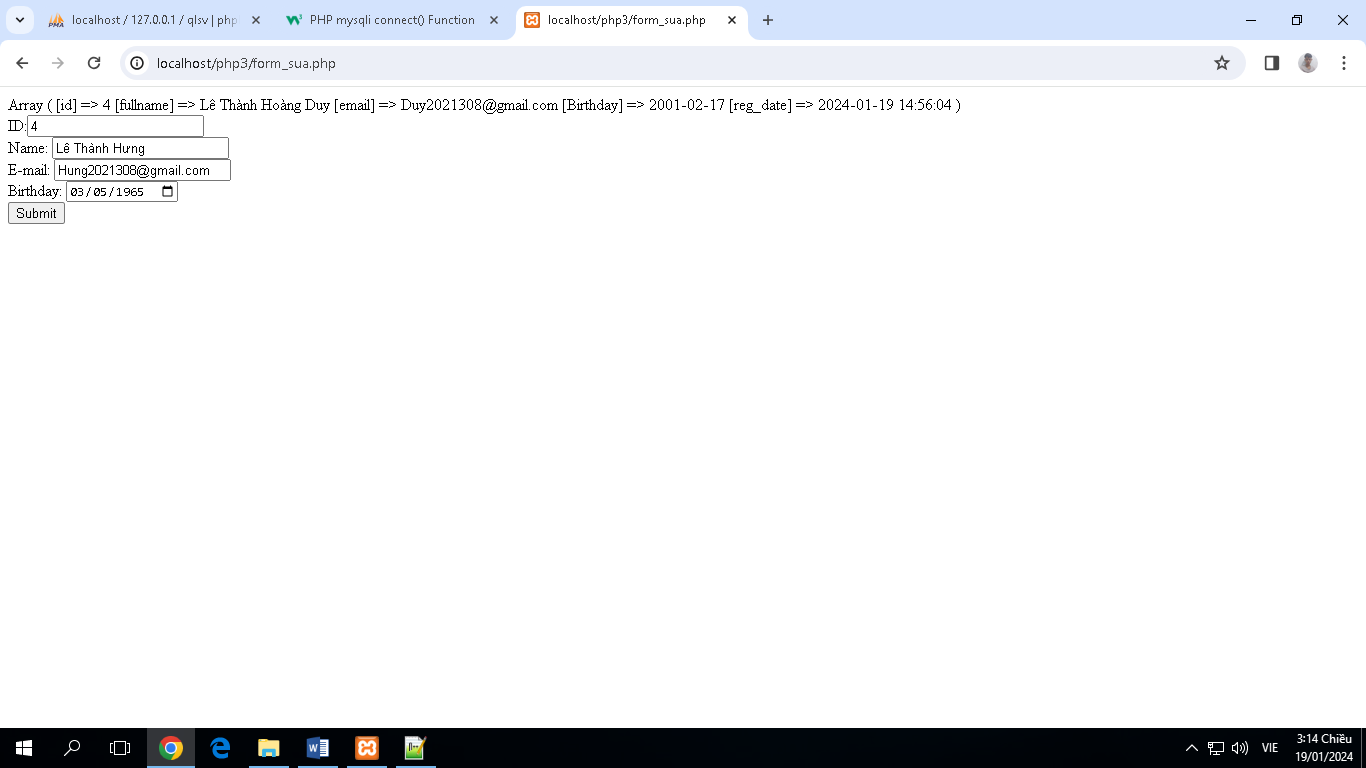




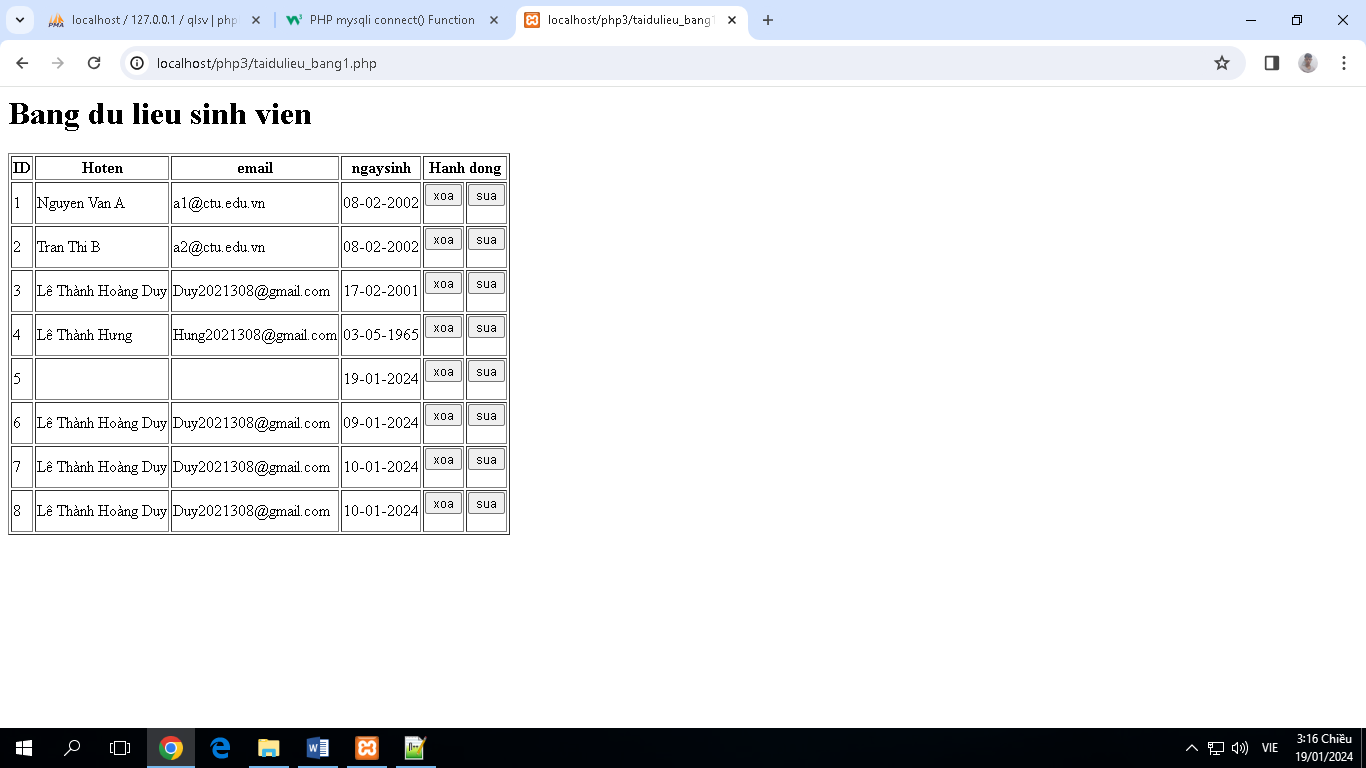
Run tập tin taidulieu\_bang1.php:



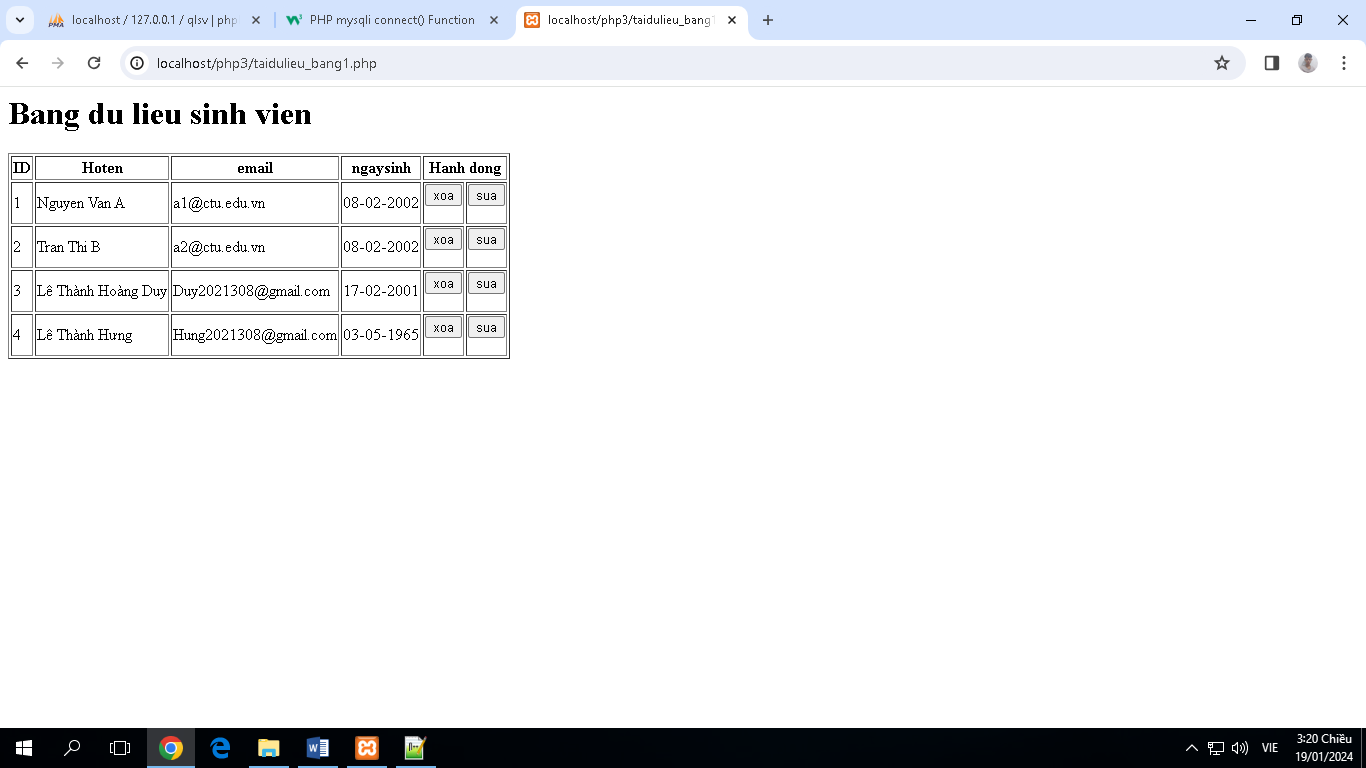
Khi chúng ta chọn sửa dữ liệu, tập tin sau (form\_sua.php) sẽ thực thi:



Khi bấm nút submit sẽ gọi qua tập tin sua.php sau: khi thực hiện sửa xong, sẽ tải trở lại file taidulieu\_bang1.php



khi người dùng bấm vào nút xóa tương ứng với dòng dữ liệu trong bảng:



1. 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 MySQL:***

<?php

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "qlsv";

// Create connection

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

// Check connection

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

***Truy vấn Oracle:***

Connection conn = null;

Statement st = null;

ResultSet rs = null;

try {

String dbURL = "jdbc:oracle:thin:@localhost:1521:XE";

// Tài khoản đăng nhập oracle

String username = "kylh";

String password = "Abc12345";

// Tạo kết nối đến server

conn = DriverManager.getConnection(dbURL, username, password);

if (conn != null) {

System.out.println("Kết nối thành công");

}

***Kết nối với SQL sever:***

function ReadData()

{

try

{

$conn = OpenConnection();

$tsql = "SELECT [CompanyName] FROM SalesLT.Customer";

$getProducts = sqlsrv\_query($conn, $tsql);

if ($getProducts == FALSE)

die(FormatErrors(sqlsrv\_errors()));

$productCount = 0;

while($row = sqlsrv\_fetch\_array($getProducts, SQLSRV\_FETCH\_ASSOC))

{

echo($row['CompanyName']);

echo("<br/>");

$productCount++;

}

sqlsrv\_free\_stmt($getProducts);

sqlsrv\_close($conn);

}

catch(Exception $e)

{

echo("Error!");

}

}

***Kết nối với SQLite:***

câu lệnh thường dùng để tạo và xoá bảng:

private const val SQL\_CREATE\_ENTRIES =

"CREATE TABLE ${FeedEntry.TABLE\_NAME} (" +

"${BaseColumns.\_ID} INTEGER PRIMARY KEY," +

"${FeedEntry.COLUMN\_NAME\_TITLE} TEXT," +

"${FeedEntry.COLUMN\_NAME\_SUBTITLE} TEXT)"

private const val SQL\_DELETE\_ENTRIES = "DROP TABLE IF EXISTS ${FeedEntry.TABLE\_NAME}"

triển khai SQLiteOpenHelper sử dụng một số lệnh nêu trên:

triển khai SQLiteOpenHelper sử dụng một số lệnh nêu trên:class FeedReaderDbHelper(context: Context) : SQLiteOpenHelper(context, DATABASE\_NAME, null, DATABASE\_VERSION) {

override fun onCreate(db: SQLiteDatabase) {

db.execSQL(SQL\_CREATE\_ENTRIES)

}

override fun onUpgrade(db: SQLiteDatabase, oldVersion: Int, newVersion: Int) {

// This database is only a cache for online data, so its upgrade policy is

// to simply to discard the data and start over

db.execSQL(SQL\_DELETE\_ENTRIES)

onCreate(db)

}

override fun onDowngrade(db: SQLiteDatabase, oldVersion: Int, newVersion: Int) {

onUpgrade(db, oldVersion, newVersion)

}

companion object {

// If you change the database schema, you must increment the database version.

const val DATABASE\_VERSION = 1

const val DATABASE\_NAME = "FeedReader.db"

}

}

Truy cập vào cơ sở dữ liệu của bạn, hãy tạo bản sao lớp con của SQLiteOpenHelper:

val dbHelper = FeedReaderDbHelper(context)

1. 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]](#footnote-2) hoặc các tài liệu khác mà bạn tìm được).

Các chức năng MySQLi cho phép bạn truy cập các máy chủ cơ sở dữ liệu MySQL.

affected\_rows() Trả về số lượng hàng bị ảnh hưởng trong hoạt động MySQL trước đó

autocommit() Bật hoặc tắt tự động sửa đổi cơ sở dữ liệu

begin\_transaction() Bắt đầu một giao dịch

change\_user() Thay đổi người dùng của kết nối cơ sở dữ liệu đã chỉ định

character\_set\_name() Trả về bộ ký tự mặc định cho kết nối cơ sở dữ liệu

close() Đóng kết nối cơ sở dữ liệu đã mở trước đó

commit() Cam kết giao dịch hiện tại

connect() Mở một kết nối mới tới máy chủ MySQL

connect\_errno() Trả về mã lỗi từ lỗi kết nối gần đây nhất

connect\_error() Trả về mô tả lỗi từ lỗi kết nối cuối cùng

data\_seek() Điều chỉnh con trỏ kết quả tới một hàng tùy ý trong tập kết quả

debug() Thực hiện các thao tác gỡ lỗi

dump\_debug\_info() Kết xuất thông tin gỡ lỗi vào nhật ký

errno() Trả về mã lỗi cuối cùng cho lệnh gọi hàm gần đây nhất

error() Trả về mô tả lỗi cuối cùng cho lệnh gọi hàm gần đây nhất

error\_list() Trả về danh sách lỗi cho lệnh gọi hàm gần đây nhất

fetch\_all() Tìm nạp tất cả các hàng kết quả dưới dạng mảng kết hợp, mảng số hoặc cả hai

fetch\_array() Tìm nạp một hàng kết quả dưới dạng kết hợp, mảng số hoặc cả hai

fetch\_assoc() Tìm nạp hàng kết quả dưới dạng mảng kết hợp

fetch\_field() Trả về trường tiếp theo trong tập kết quả, dưới dạng đối tượng

fetch\_field\_direct() Trả về siêu dữ liệu cho một trường trong tập kết quả, dưới dạng đối tượng

fetch\_fields() Trả về một mảng các đối tượng đại diện cho các trường trong tập kết quả

fetch\_lengths() Trả về độ dài của các cột của hàng hiện tại trong tập kết quả

fetch\_object() Trả về hàng hiện tại của tập kết quả, dưới dạng đối tượng

fetch\_row() Tìm nạp một hàng từ tập kết quả và trả về dưới dạng mảng liệt kê

field\_count() Trả về số cột cho truy vấn gần đây nhất

field\_seek() Đặt con trỏ trường thành offset trường đã cho

get\_charset() Trả về một đối tượng bộ ký tự

get\_client\_info() Trả về phiên bản thư viện máy khách MySQL

get\_client\_stats() Trả về số liệu thống kê về máy khách trên mỗi quy trình

get\_client\_version() Trả về phiên bản thư viện máy khách MySQL dưới dạng số nguyên

get\_connection\_stats() Trả về số liệu thống kê về kết nối máy khách

get\_host\_info() Trả về tên máy chủ của máy chủ MySQL và loại kết nối

get\_proto\_info() Trả về phiên bản giao thức MySQL

get\_server\_info() Trả về phiên bản máy chủ MySQL

get\_server\_version() Trả về phiên bản máy chủ MySQL dưới dạng số nguyên

info() Trả về thông tin về truy vấn được thực hiện gần đây nhất

init() Khởi tạo MySQLi và trả về tài nguyên để sử dụng với real\_connect()

insert\_id() Trả về id được tạo tự động từ truy vấn cuối cùng

kill() Yêu cầu máy chủ hủy một luồng MySQL

more\_results() Kiểm tra xem có nhiều kết quả hơn từ nhiều truy vấn không

multi\_query() Thực hiện một hoặc nhiều truy vấn trên cơ sở dữ liệu

next\_result() Chuẩn bị tập kết quả tiếp theo từ multi\_query()

options() Đặt tùy chọn kết nối bổ sung và ảnh hưởng đến hành vi cho kết nối

ping() Ping kết nối máy chủ hoặc cố gắng kết nối lại nếu kết nối bị hỏnghas gone down

poll() Kết nối thăm dò ý kiến

prepare() Chuẩn bị một câu lệnh SQL để thực thi

query() Thực hiện truy vấn đối với cơ sở dữ liệu

real\_connect() Mở kết nối mới tới máy chủ MySQL

real\_escape\_string() Thoát các ký tự đặc biệt trong chuỗi để sử dụng trong câu lệnh SQL

real\_query() Thực hiện một truy vấn SQL

reap\_async\_query() Trả về kết quả từ truy vấn SQL không đồng bộ

refresh() Làm mới/xóa bảng hoặc bộ nhớ đệm hoặc đặt lại thông tin máy chủ sao chép

rollback() Khôi phục giao dịch hiện tại cho cơ sở dữ liệu

select\_db() Chọn cơ sở dữ liệu mặc định cho các truy vấn cơ sở dữ liệu

set\_charset() Đặt bộ ký tự máy khách mặc định

set\_local\_infile\_default() Hủy đặt trình xử lý do người dùng xác định để tải lệnh infile cục bộ

set\_local\_infile\_handler() Đặt chức năng gọi lại cho lệnh LOAD DATA LOCAL INFILE

sqlstate() Trả về mã lỗi SQLSTATE của lỗi

ssl\_set() Được sử dụng để thiết lập kết nối an toàn bằng SSL

stat() Trả về trạng thái hiện tại của hệ thống

stmt\_init() Khởi tạo một câu lệnh và trả về một đối tượng để sử dụng với stmt\_prepare()

store\_result() Chuyển tập kết quả từ truy vấn cuối cùng

thread\_id() Trả về ID luồng cho kết nối hiện tại

thread\_safe() Trả về xem thư viện máy khách có được biên dịch dưới dạng thread-safe hay không

use\_result() Bắt đầu truy xuất tập kết quả từ truy vấn cuối cùng được thực hiện

warning\_count() Trả về số lượng cảnh báo từ truy vấn cuối cùng trong kết nối

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

Phương thức die() dùng để dừng hệ thống, không tiếp tục dịch nội dung PHP, và trả về thông báo

1. 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.

Ngoài 2 phương thức trên, các phương thức khác trong **mysqli** có chức năng tương tự:

fetch\_array() Tìm nạp một hàng kết quả dưới dạng kết hợp, mảng số hoặc cả hai

fetch\_field() Trả về trường tiếp theo trong tập kết quả, dưới dạng đối tượng

fetch\_field\_direct() Trả về siêu dữ liệu cho một trường trong tập kết quả, dưới dạng đối tượng

fetch\_fields() Trả về một mảng các đối tượng đại diện cho các trường trong tập kết quả

fetch\_lengths() Trả về độ dài của các cột của hàng hiện tại trong tập kết quả

fetch\_object() Trả về hàng hiện tại của tập kết quả, dưới dạng đối tượng

fetch\_row() Tìm nạp một hàng từ tập kết quả và trả về dưới dạng mảng liệt kê

1. 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]](#footnote-3)).
2. 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*)

1. 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
2. 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.
3. 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

# Chú ý:

* Các bạn nộp file word: Đặt tên file: <mssv><hoten><bai3>.docx nộp lên classroom, kèm với các file code đã tạo (**ngoại trừ file word trả lời câu hỏi, các file còn lại các bạn nén vào file zip**).
* Mỗi câu các bạn trả lời bằng hình/trả lời dạng text tùy vào yêu cầu của câu hỏi, và ghi rõ số thứ tự câu hỏi mà bạn trả lời.
* Các bài phát hiện copy từ các bài của bạn khác sẽ không được tính điểm.
* *Chú ý: Các câu trả lời có tham khảo trên Internet phải trích dẫn link*

1. Mysqli: https://www.w3schools.com/php/php\_ref\_mysqli.asp#:~:text=The%20MySQLi%20functions%20allows%20you,13%20or%20newer. [↑](#footnote-ref-1)
2. https://www.w3schools.com/php/php\_ref\_mysqli.asp [↑](#footnote-ref-2)
3. https://www.php.net/manual/en/function.header.php [↑](#footnote-ref-3)