

Bài 5. PHP và MySQL

Giảng viên: Trần Thị Thu Phương

Email: tttphuong2@hnmu.edu.vn

Mobile: 0966 224784

Nội dung

1. Kết nối CSDL
2. Các thao tác với CSDL
3. Tham khảo tại [PHP MySQL Connect to database \(w3schools.com\)](#)

MySQL Database

MySQL Connect

MySQL Create DB

MySQL Create Table

MySQL Insert Data

MySQL Get Last ID

MySQL Insert Multiple

MySQL Prepared

MySQL Select Data

MySQL Where

MySQL Order By

MySQL Delete Data

MySQL Update Data

MySQL Limit Data

MySQL: Kết nối CSDL

Kết nối MySQL

- Tiện ích mở rộng MySQLi (chữ "i" là viết tắt của cải tiến-improved)
- PDO (Đối tượng dữ liệu PHP-PHP Data Objects)

Nên sử dụng MySQLi hay PDO

- PDO sẽ hoạt động trên 12 hệ thống cơ sở dữ liệu khác nhau, trong khi MySQLi sẽ chỉ hoạt động với cơ sở dữ liệu MySQL.
- Vì vậy, nếu phải chuyển dự án của mình sang sử dụng cơ sở dữ liệu khác, PDO sẽ giúp quá trình này trở nên dễ dàng, chỉ phải thay đổi chuỗi kết nối và một vài truy vấn. Với MySQLi, sẽ cần phải viết lại toàn bộ mã - bao gồm cả truy vấn.
- Cả hai đều hướng đối tượng, nhưng MySQLi cũng cung cấp API thủ tục.

1. 1 Mở kết nối

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

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

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

Hướng đối tượng MySQLi

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

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

// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}
echo "Connected successfully";
?>
```

Hướng thủ tục MySQLi

1.2 Mở kết nối: PDO

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

try {
    $conn = new PDO("mysql:host=$servername;dbname=myDB", $username, $password);
    // set the PDO error mode to exception
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    echo "Connected successfully";
} catch(PDOException $e) {
    echo "Connection failed: " . $e->getMessage();
}
?>
```

2.Đóng kết nối

- Hướng đối tượng MySQLi: `$conn->close();`
- Thủ tục MySQLi: `mysqli_close($conn);`
- PDO: `$conn = null;`

3. MySQL: Tạo CSDL

3.1 Tạo CSDL: MySQLi

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

// Create connection
$conn = new mysqli($servername, $username, $password);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

// Create database
$sql = "CREATE DATABASE myDB";
if ($conn->query($sql) === TRUE) {
    echo "Database created successfully";
} else {
    echo "Error creating database: " . $conn->error;
}

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

Hướng đối tượng MySQLi

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

// Create connection
$conn = mysqli_connect($servername, $username, $password);
// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}

// Create database
$sql = "CREATE DATABASE myDB";
if (mysqli_query($conn, $sql)) {
    echo "Database created successfully";
} else {
    echo "Error creating database: " . mysqli_error($conn);
}

mysqli_close($conn);
?>
```

Thủ tục MySQLi

3.2 Tạo CSDL: PDO

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

try {
    $conn = new PDO("mysql:host=$servername", $username, $password);
    // set the PDO error mode to exception
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    $sql = "CREATE DATABASE myDBPDO";
    // use exec() because no results are returned
    $conn->exec($sql);
    echo "Database created successfully<br>";
} catch(PDOException $e) {
    echo $sql . "<br>" . $e->getMessage();
}

$conn = null;
?>
```

- Một lợi ích tuyệt vời của PDO là nó có lớp ngoại lệ để xử lý bất kỳ sự cố nào có thể xảy ra trong các truy vấn cơ sở dữ liệu của chúng ta. Nếu một ngoại lệ được đưa ra trong khối try{ }, tập lệnh sẽ ngừng thực thi và chuyển trực tiếp đến khối catch(){ } đầu tiên

4.MySQL: Tạo bảng

Tạo một bảng có tên "MyGuests", với năm cột: "id", "firstname", "lastname", "email" và "reg_date"

Tạo bảng

```
CREATE TABLE MyGuests (  
  id INT(6) UNSIGNED AUTO_INCREMENT PRIMARY KEY,  
  firstname VARCHAR(30) NOT NULL,  
  lastname VARCHAR(30) NOT NULL,  
  email VARCHAR(50),  
  reg_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
  ON UPDATE CURRENT_TIMESTAMP  
)
```

Lưu ý

- Các kiểu dữ liệu
- NOT NULL - Mỗi hàng phải chứa một giá trị cho cột đó, giá trị null không được phép
- Giá trị MẶC ĐỊNH - Đặt giá trị mặc định được thêm vào khi không có giá trị nào khác được chuyển
- UNSIGNED - Được sử dụng cho các loại số, giới hạn dữ liệu được lưu trữ ở số dương và số không
- TĂNG TỰ ĐỘNG - MySQL tự động tăng giá trị của trường lên 1 mỗi lần thêm bản ghi mới
- PRIMARY KEY - Được sử dụng để xác định duy nhất các hàng trong bảng. Cột có cài đặt PRIMARY KEY thường là số ID và thường được sử dụng với AUTO_INCREMENT

4.1 Tạo bảng MySQLi

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

// 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 MyGuests (
id INT(6) UNSIGNED AUTO_INCREMENT PRIMARY KEY,
firstname VARCHAR(30) NOT NULL,
lastname VARCHAR(30) NOT NULL,
email VARCHAR(50),
reg_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
)";

if ($conn->query($sql) === TRUE) {
    echo "Table MyGuests created successfully";
} else {
    echo "Error creating table: " . $conn->error;
}
```

**Hướng đối tượng
MySQLi**

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

// Create connection
$conn = mysqli_connect($servername, $username, $password, $dbname);
// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}

// sql to create table
$sql = "CREATE TABLE MyGuests (
id INT(6) UNSIGNED AUTO_INCREMENT PRIMARY KEY,
firstname VARCHAR(30) NOT NULL,
lastname VARCHAR(30) NOT NULL,
email VARCHAR(50),
reg_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
)";

if (mysqli_query($conn, $sql)) {
    echo "Table MyGuests created successfully";
} else {
    echo "Error creating table: " . mysqli_error($conn);
}

mysqli_close($conn);
?>
```

**Hướng thủ tục
MySQLi**

4.2 (PDO)

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

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 to create table
    $sql = "CREATE TABLE MyGuests (
    id INT(6) UNSIGNED AUTO_INCREMENT PRIMARY KEY,
    firstname VARCHAR(30) NOT NULL,
    lastname VARCHAR(30) NOT NULL,
    email VARCHAR(50),
    reg_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
    )";

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

$conn = null;
?>
```


5. MySQL: Chèn dữ liệu vào bảng

Chèn dữ liệu vào bảng có tên "MyGuests", với năm cột: "id", "firstname", "lastname", "email" và "reg_date"

Cú pháp

```
INSERT INTO table_name (column1, column2, column3,...)
VALUES (value1, value2, value3,...)
```

- Truy vấn SQL phải được trích dẫn bằng PHP
- Các giá trị chuỗi bên trong truy vấn SQL phải được trích dẫn
- Các giá trị số không được trích dẫn
- Từ NULL không được trích dẫn
- Nếu một cột là AUTO_INCREMENT (như cột "id") hoặc TIMESTAMP có cập nhật mặc định là current_timestamp (như cột "reg_date"), thì không cần chỉ định trong truy vấn SQL; MySQL sẽ tự động thêm giá trị.

5.1 Chèn dữ liệu bằng MySQLi

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

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

$sql = "INSERT INTO MyGuests (firstname, lastname, email)
VALUES ('John', 'Doe', 'john@example.com')";

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

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

**Hướng đối tượng
MySQLi**

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

// Create connection
$conn = mysqli_connect($servername, $username, $password, $dbname);
// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}

$sql = "INSERT INTO MyGuests (firstname, lastname, email)
VALUES ('John', 'Doe', 'john@example.com')";

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

mysqli_close($conn);
?>
```

**Hướng thủ tục
MySQLi**

5.2 PDO

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

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 (firstname, lastname, email)
VALUES ('John', 'Doe', 'john@example.com')";
    // 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;
?>
```

6.MySQL: Truy vấn dữ liệu

Truy vấn dữ liệu từ bảng có tên "MyGuests", với năm cột: "id", "firstname", "lastname", "email" và "reg_date"

Cú pháp

- `SELECT column_name(s) FROM table_name`
- `SELECT * FROM table_name`

6.1 MySQLi

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

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

$sql = "SELECT id, firstname, lastname FROM MyGuests";
$result = $conn->query($sql);

if ($result->num_rows > 0) {
    // output data of each row
    while($row = $result->fetch_assoc()) {
        echo "id: " . $row["id"]. " - Name: " . $row["firstname"]. " " . $row["lastname"]. "<br>";
    }
} else {
    echo "0 results";
}
$conn->close();
?>
```

Hướng đối tượng
MySQLi

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

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

$sql = "SELECT id, firstname, lastname FROM MyGuests";
$result = $conn->query($sql);

if ($result->num_rows > 0) {
    echo "<table><tr><th>ID</th><th>Name</th></tr>";
    // output data of each row
    while($row = $result->fetch_assoc()) {
        echo "<tr><td>". $row["id"]. "</td><td>". $row["firstname"]. " " . $row["lastname"]. "</td></tr>";
    }
    echo "</table>";
} else {
    echo "0 results";
}
$conn->close();
?>
```

Hướng thủ tục
MySQLi

6.2 (PDO)

```
<?php
echo "<table style='border: solid 1px black;'>";
echo "<tr><th>Id</th><th>Firstname</th><th>Lastname</th></tr>";

class TableRows extends RecursiveIteratorIterator {
    function __construct($it) {
        parent::__construct($it, self::LEAVES_ONLY);
    }

    function current() {
        return "<td style='width:150px;border:1px solid black;'>" . parent::current(). "</td>";
    }

    function beginChildren() {
        echo "<tr>";
    }

    function endChildren() {
        echo "</tr>" . "\n";
    }
}

$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDBPDO";

try {
    $conn = new PDO("mysql:host=$servername;dbname=$dbname", $username, $password);
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    $stmt = $conn->prepare("SELECT id, firstname, lastname FROM MyGuests");
    $stmt->execute();

    // set the resulting array to associative
    $result = $stmt->setFetchMode(PDO::FETCH_ASSOC);
    foreach(new TableRows(new RecursiveArrayIterator($stmt->fetchAll())) as $k=>$v) {
        echo $v;
    }
} catch(PDOException $e) {
    echo "Error: " . $e->getMessage();
}
$conn = null;
echo "</table>";
?>
```


MySQL: Truy vấn dữ liệu có điều kiện

Truy vấn dữ liệu từ bảng có tên "MyGuests", với năm cột: "id", "firstname", "lastname", "email" và "reg_date" thỏa tên là "Doe"

Cú pháp

- SELECT column_name(s) FROM table_name WHERE column_name operator value

6.1 MySQLi

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

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

$sql = "SELECT id, firstname, lastname FROM MyGuests WHERE lastname='Doe'";
$result = $conn->query($sql);

if ($result->num_rows > 0) {
    // output data of each row
    while($row = $result->fetch_assoc()) {
        echo "id: " . $row["id"]. " - Name: " . $row["firstname"]. " " . $row["lastname"]. "<br>";
    }
} else {
    echo "0 results";
}
$conn->close();
?>
```

**Hướng đối
tượng MySQLi**

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

// Create connection
$conn = mysqli_connect($servername, $username, $password, $dbname);
// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}

$sql = "SELECT id, firstname, lastname FROM MyGuests WHERE lastname='Doe'";
$result = mysqli_query($conn, $sql);

if (mysqli_num_rows($result) > 0) {
    // output data of each row
    while($row = mysqli_fetch_assoc($result)) {
        echo "id: " . $row["id"]. " - Name: " . $row["firstname"]. " " . $row["lastname"]. "<br>";
    }
} else {
    echo "0 results";
}

mysqli_close($conn);
?>
```

**Hướng thủ tục
MySQLi**

6.3 PDO

```
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDBPDO";

try {
    $conn = new PDO("mysql:host=$servername;dbname=$dbname", $username, $password);
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    $stmt = $conn->prepare("SELECT id, firstname, lastname FROM MyGuests WHERE lastname='Doe'");
    $stmt->execute();

    // set the resulting array to associative
    $result = $stmt->setFetchMode(PDO::FETCH_ASSOC);
    foreach(new TableRows(new RecursiveArrayIterator($stmt->fetchAll())) as $k=>$v) {
        echo $v;
    }
}
catch(PDOException $e) {
    echo "Error: " . $e->getMessage();
}
$conn = null;
echo "</table>";
?>
```

7.MySQL: Truy vấn, sắp xếp dữ liệu

Chọn các cột id, firstname và lastname từ bảng MyGuests. Các bản ghi sẽ được sắp xếp theo lastname

Cú pháp

- Mệnh đề ORDER BY được sử dụng để sắp xếp tập kết quả theo thứ tự tăng dần hoặc giảm dần.
- Theo mặc định, mệnh đề ORDER BY sắp xếp các bản ghi theo thứ tự tăng dần. Để sắp xếp các bản ghi theo thứ tự giảm dần, hãy sử dụng từ khóa DESC.
- `SELECT column_name(s) FROM table_name ORDER BY column_name(s) ASC|DESC`

7.1 (MySQLi)

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

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

$sql = "SELECT id, firstname, lastname FROM MyGuests ORDER BY lastname";
$result = $conn->query($sql);

if ($result->num_rows > 0) {
    // output data of each row
    while($row = $result->fetch_assoc()) {
        echo "id: " . $row["id"]. " - Name: " . $row["firstname"]. " " . $row["lastname"]. "<br>";
    }
} else {
    echo "0 results";
}
$conn->close();
?>
```

**Hướng đối
tượng MySQLi**

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

// Create connection
$conn = mysqli_connect($servername, $username, $password, $dbname);
// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}

$sql = "SELECT id, firstname, lastname FROM MyGuests ORDER BY lastname";
$result = mysqli_query($conn, $sql);

if (mysqli_num_rows($result) > 0) {
    // output data of each row
    while($row = mysqli_fetch_assoc($result)) {
        echo "id: " . $row["id"]. " - Name: " . $row["firstname"]. " " . $row["lastname"]. "<br>";
    }
} else {
    echo "0 results";
}

mysqli_close($conn);
?>
```

**Hướng thủ tục
MySQLi**

7.3 PDO

```
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDBPDO";

try {
    $conn = new PDO("mysql:host=$servername;dbname=$dbname", $username, $password);
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    $stmt = $conn->prepare("SELECT id, firstname, lastname FROM MyGuests ORDER BY lastname");
    $stmt->execute();

    // set the resulting array to associative
    $result = $stmt->setFetchMode(PDO::FETCH_ASSOC);
    foreach(new TableRows(new RecursiveArrayIterator($stmt->fetchAll())) as $k=>$v) {
        echo $v;
    }
} catch(PDOException $e) {
    echo "Error: " . $e->getMessage();
}
$conn = null;
echo "</table>";
?>
```


8.MySQL: Xóa dữ liệu

id	firstname	lastname	email	reg_date
1	John	Doe	john@example.com	2014-10-22 14:26:15
2	Mary	Moe	mary@example.com	2014-10-23 10:22:30
3	Julie	Dooley	julie@example.com	2014-10-26 10:48:23

Xóa dòng có id=3

Cú pháp

- DELETE FROM table_name
WHERE some_column = some_value
- Mệnh đề WHERE chỉ định bản ghi hoặc các bản ghi sẽ bị xóa. Nếu bỏ qua mệnh đề WHERE, tất cả các bản ghi sẽ bị xóa!

8.1 (MySQLi)

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

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

// sql to delete a record
$sql = "DELETE FROM MyGuests WHERE id=3";

if ($conn->query($sql) === TRUE) {
    echo "Record deleted successfully";
} else {
    echo "Error deleting record: " . $conn->error;
}

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

**Hướng đối
tượng MySQLi**

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

// Create connection
$conn = mysqli_connect($servername, $username, $password, $dbname);
// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}

// sql to delete a record
$sql = "DELETE FROM MyGuests WHERE id=3";

if (mysqli_query($conn, $sql)) {
    echo "Record deleted successfully";
} else {
    echo "Error deleting record: " . mysqli_error($conn);
}

mysqli_close($conn);
?>
```

**Hướng thủ tục
MySQLi**

8.3 (PDO)

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

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 to delete a record
    $sql = "DELETE FROM MyGuests WHERE id=3";

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

$conn = null;
?>
```

9 MySQL: Cập nhật dữ liệu

id	firstname	lastname	email	reg_date
1	John	Doe	john@example.com	2014-10-22 14:26:15
2	Mary	Moe	mary@example.com	2014-10-23 10:22:30
3	Julie	Dooley	julie@example.com	2014-10-26 10:48:23

Cập nhật có id=2

Cú pháp

- UPDATE table_name

SET column1=value, column2=value2,...

WHERE some_column=some_value

Mệnh đề WHERE chỉ định bản ghi hoặc các bản ghi sẽ được cập nhật. Nếu bỏ qua mệnh đề WHERE, tất cả các bản ghi sẽ được cập nhật!

9.1 (Hướng đối tượng MySQLi)

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

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

$sql = "UPDATE MyGuests SET lastname='Doe' WHERE id=2";

if ($conn->query($sql) === TRUE) {
    echo "Record updated successfully";
} else {
    echo "Error updating record: " . $conn->error;
}

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

**Hướng đối
tượng MySQLi**

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

// Create connection
$conn = mysqli_connect($servername, $username, $password, $dbname);
// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}

$sql = "UPDATE MyGuests SET lastname='Doe' WHERE id=2";

if (mysqli_query($conn, $sql)) {
    echo "Record updated successfully";
} else {
    echo "Error updating record: " . mysqli_error($conn);
}

mysqli_close($conn);
?>
```

**Hướng thủ tục
MySQLi**

9.3 (PDO)

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

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 = "UPDATE MyGuests SET lastname='Doe' WHERE id=2";

    // Prepare statement
    $stmt = $conn->prepare($sql);

    // execute the query
    $stmt->execute();

    // echo a message to say the UPDATE succeeded
    echo $stmt->rowCount() . " records UPDATED successfully";
} catch(PDOException $e) {
    echo $sql . "<br>" . $e->getMessage();
}

$conn = null;
?>
```


MySQL: Giới hạn dữ liệu

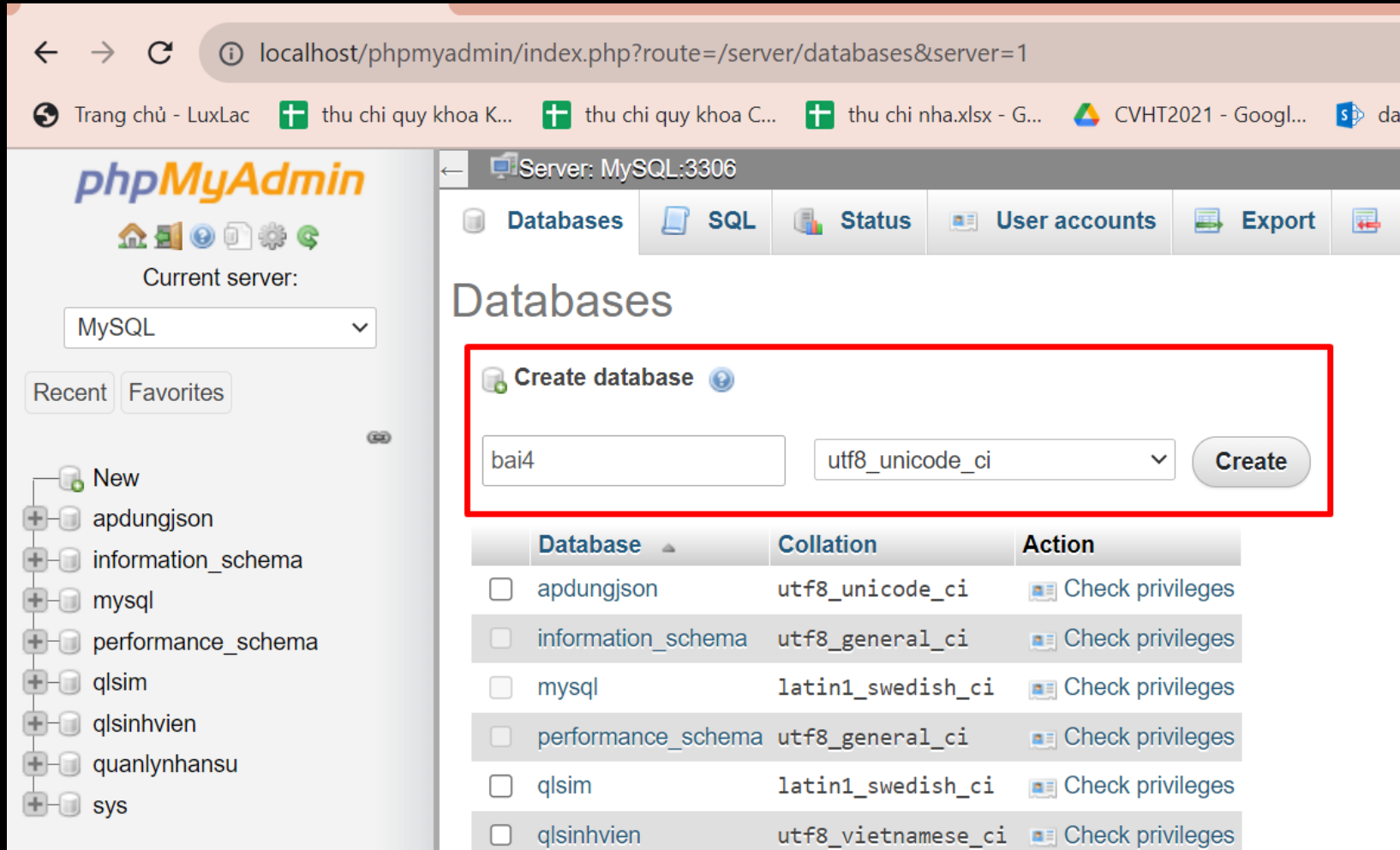
Giới hạn dữ liệu

- `$sql = "SELECT * FROM Orders LIMIT 30";`
- `$sql = "SELECT * FROM Orders LIMIT 10 OFFSET 15";` chỉ trả về 10 bản ghi, bắt đầu từ bản ghi 16 (OFFSET 15)
- Hoặc: `$sql = "SELECT * FROM Orders LIMIT 15, 10";`

Bài tập

- Tự tạo CSDL myDB gồm 1 bảng myGuest, sử dụng các câu lệnh tạo, cập nhật dữ liệu với CSDL vừa tạo bằng file php bằng 2 phương pháp MySQLi và PDO, nội dung cụ thể xem trong assignment.

Hướng dẫn: Tạo CSDL bai4



The screenshot shows the phpMyAdmin web interface. The browser address bar indicates the URL is `localhost/phpmyadmin/index.php?route=/server/databases&server=1`. The interface is for a MySQL server running on port 3306. The 'Databases' tab is selected, and the 'Create database' form is highlighted with a red rectangle. The form contains the database name 'bai4' and the collation 'utf8_unicode_ci'. Below the form is a table listing existing databases and their collations.

Current server: MySQL

Recent Favorites

New

- apdungjson
- information_schema
- mysql
- performance_schema
- qlsim
- qlsinhvien
- quanlynhansu
- sys

Server: MySQL:3306

Databases SQL Status User accounts Export

Databases

Create database ?

bai4 utf8_unicode_ci Create

	Database	Collation	Action
<input type="checkbox"/>	apdungjson	utf8_unicode_ci	Check privileges
<input type="checkbox"/>	information_schema	utf8_general_ci	Check privileges
<input type="checkbox"/>	mysql	latin1_swedish_ci	Check privileges
<input type="checkbox"/>	performance_schema	utf8_general_ci	Check privileges
<input type="checkbox"/>	qlsim	latin1_swedish_ci	Check privileges
<input type="checkbox"/>	qlsinhvien	utf8_vietnamese_ci	Check privileges

Tạo bảng MyGuests

Current server:

MySQL

Recent Favorites

New

apdungjson

bai4

information_schema

mysql

performance_schema

No tables found in database.

Create table

Name: MyGuests Number of columns: 5

Go

MySQL: Tạo bảng

Table name: Add column(s)

Name	Type ?	Length/Values ?	Default ?	Collation	Attributes	Null	Index	A_I	C
<input type="text" value="id"/>	<input type="text" value="INT"/> ▼	<input type="text" value="6"/>	<input type="text" value="None"/> ▼	<input type="text" value=""/> ▼	<input type="text" value="UNSIGNED"/> ▼	<input type="checkbox"/>	<input type="text" value="PRIMARY"/> ▼ PRIMARY	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input type="text" value="firstname"/>	<input type="text" value="VARCHAR"/> ▼	<input type="text" value="30"/>	<input type="text" value="None"/> ▼	<input type="text" value=""/> ▼	<input type="text" value=""/> ▼	<input type="checkbox"/>	<input type="text" value="---"/> ▼	<input type="checkbox"/>	<input type="checkbox"/>
<input type="text" value="lastname"/>	<input type="text" value="VARCHAR"/> ▼	<input type="text" value="30"/>	<input type="text" value="None"/> ▼	<input type="text" value=""/> ▼	<input type="text" value=""/> ▼	<input type="checkbox"/>	<input type="text" value="---"/> ▼	<input type="checkbox"/>	<input type="checkbox"/>
<input type="text" value="email"/>	<input type="text" value="VARCHAR"/> ▼	<input type="text" value="50"/>	<input type="text" value="None"/> ▼	<input type="text" value=""/> ▼	<input type="text" value=""/> ▼	<input type="checkbox"/>	<input type="text" value="---"/> ▼	<input type="checkbox"/>	<input type="checkbox"/>
<input type="text" value="reg_date"/>	<input type="text" value="TIMESTAMP"/> ▼	<input type="text" value=""/>	<input type="text" value="CURRENT_TIME"/> ▼	<input type="text" value=""/> ▼	<input type="text" value="on update CURR"/> ▼	<input type="checkbox"/>	<input type="text" value="---"/> ▼	<input type="checkbox"/>	<input type="checkbox"/>

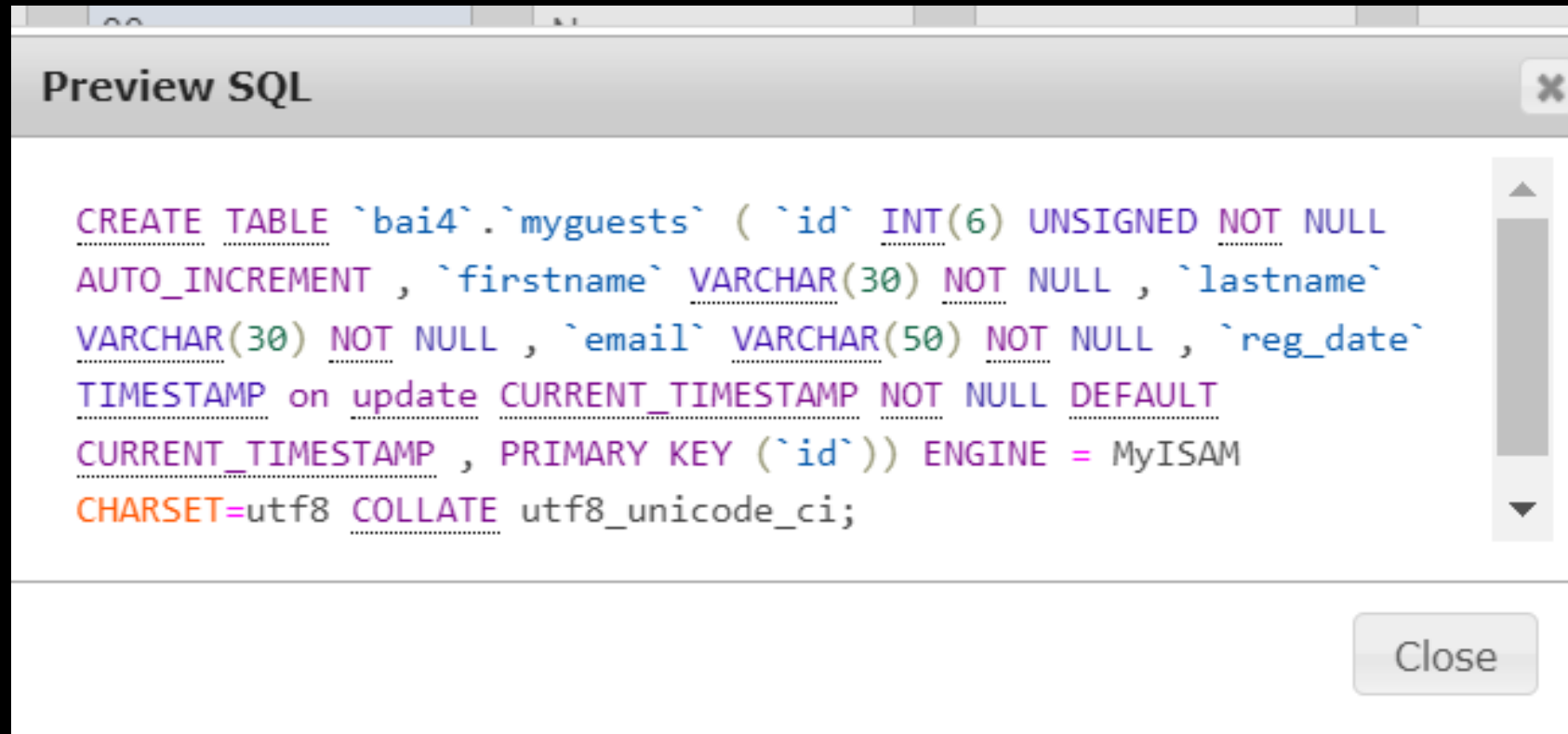
Structure ?

Table comments:

Collation: ▼

Storage Engine: ▼

SQL



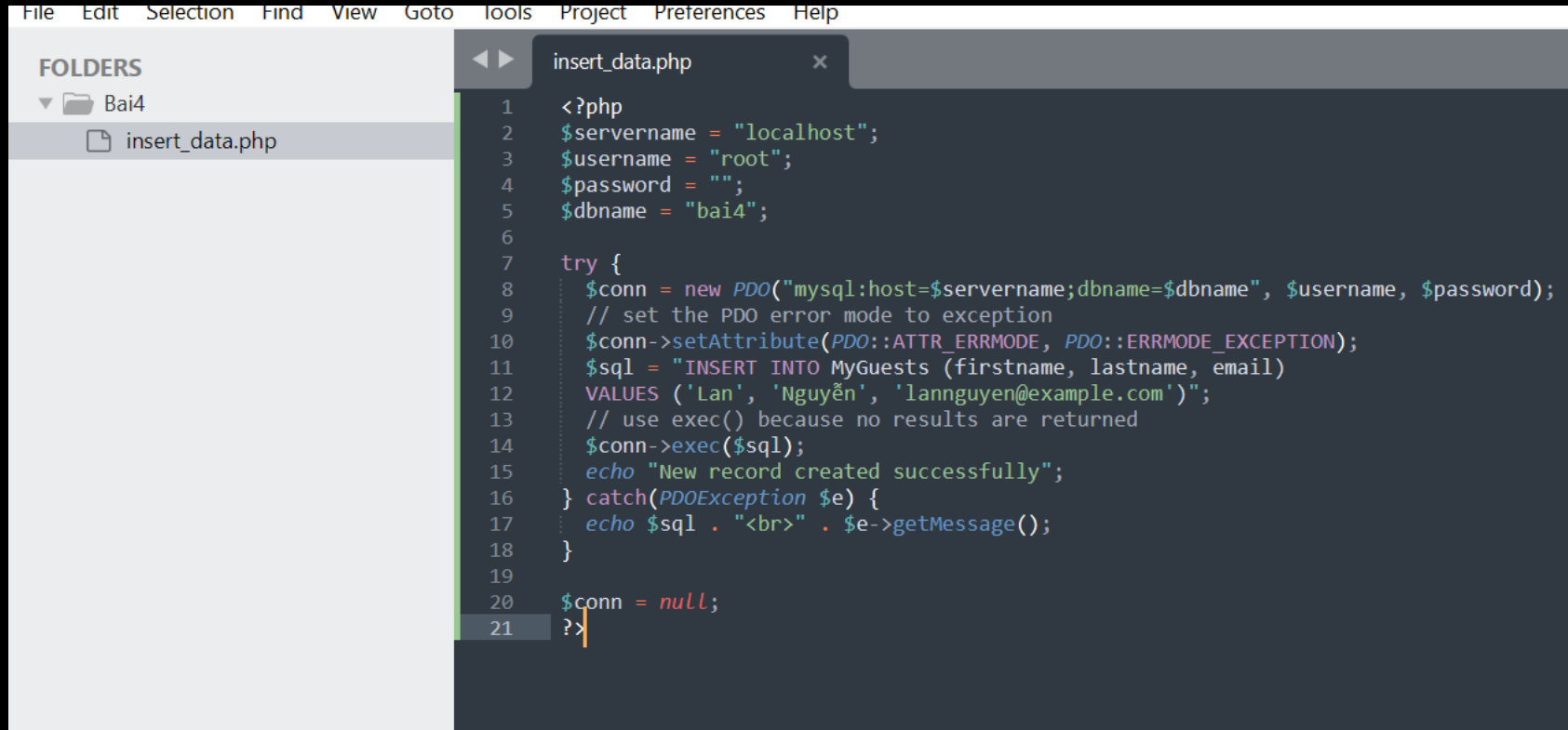
The image shows a 'Preview SQL' dialog box with a close button (X) in the top right corner. The dialog contains a MySQL CREATE TABLE statement for a table named 'myguests' in a database named 'bai4'. The table has four columns: 'id' (INT(6) UNSIGNED NOT NULL AUTO_INCREMENT), 'firstname' (VARCHAR(30) NOT NULL), 'lastname' (VARCHAR(30) NOT NULL), and 'email' (VARCHAR(50) NOT NULL). There is also a 'reg_date' column of type TIMESTAMP with an ON UPDATE CURRENT_TIMESTAMP clause. The table is set to use the MyISAM engine and the utf8 character set with the utf8_unicode_ci collation. A 'Close' button is located at the bottom right of the dialog.

```
CREATE TABLE `bai4`.`myguests` ( `id` INT(6) UNSIGNED NOT NULL  
AUTO_INCREMENT , `firstname` VARCHAR(30) NOT NULL , `lastname`  
VARCHAR(30) NOT NULL , `email` VARCHAR(50) NOT NULL , `reg_date`  
TIMESTAMP on update CURRENT_TIMESTAMP NOT NULL DEFAULT  
CURRENT_TIMESTAMP , PRIMARY KEY (`id`)) ENGINE = MyISAM  
CHARSET=utf8 COLLATE utf8_unicode_ci;
```

Close

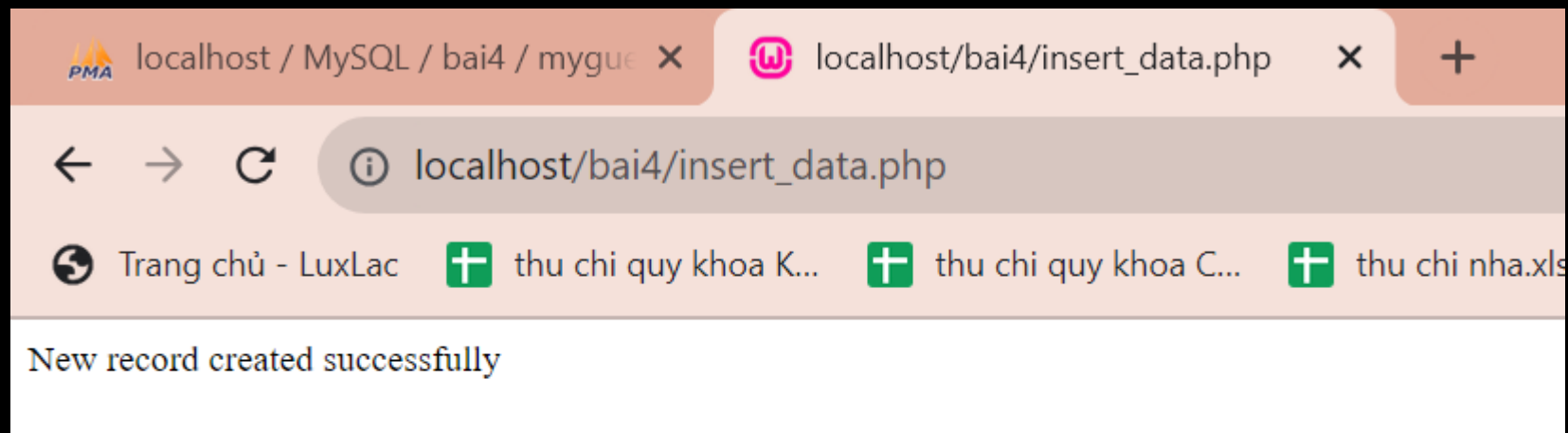
Insert dữ liệu

- Thử lần lượt insert dữ liệu bằng các cách khác nhau



The screenshot shows a code editor with a menu bar (File, Edit, Selection, Find, View, Goto, Tools, Project, Preferences, Help) and a sidebar titled 'FOLDERS' containing a folder 'Bai4' and a file 'insert_data.php'. The main editor area displays the following PHP code:

```
1 <?php
2 $servername = "localhost";
3 $username = "root";
4 $password = "";
5 $dbname = "bai4";
6
7 try {
8     $conn = new PDO("mysql:host=$servername;dbname=$dbname", $username, $password);
9     // set the PDO error mode to exception
10    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
11    $sql = "INSERT INTO MyGuests (firstname, lastname, email)
12    VALUES ('Lan', 'Nguyễn', 'lannguyen@example.com')";
13    // use exec() because no results are returned
14    $conn->exec($sql);
15    echo "New record created successfully";
16 } catch(PDOException $e) {
17     echo $sql . "<br>" . $e->getMessage();
18 }
19
20 $conn = null;
21 ?>
```

✓ Showing rows 0 - 2 (3 total, Query took 0.0067 seconds.)

```
SELECT * FROM `myguests`
```

☐ Profiling [[Edit inline](#)] [[Edit](#)] [[Explain SQL](#)] [[Create PHP code](#)] [[Refresh](#)]

☐ Show all | Number of rows: 25 | Filter rows: Sort by key: None

Options

				id	firstname	lastname	email	reg_date
<input type="checkbox"/>				1	John	Doe	john@example.com	2023-07-31 11:17:31
<input type="checkbox"/>				3	Mary	Jane	Mary@example.com	2023-07-31 14:21:22
<input type="checkbox"/>				4	Lan	Nguyễn...n	lannguyen@example.com	2023-07-31 14:24:37

☐ Check all With selected: Edit Copy Delete Export

☐ Show all | Number of rows: 25 | Filter rows: Sort by key: None