



TRƯỜNG ĐẠI HỌC VINH  
VIỆN KỸ THUẬT VÀ CÔNG NGHỆ

## CHƯƠNG 4

# PHP VÀ MYSQL

# **Nội dung giảng dạy**

---

4.1. PHP kết nối với MySQL

4.2. Tạo cơ sở dữ liệu

4.3. Tạo bảng

4.4. Truy vấn

4.5. Thêm dữ liệu

4.6. Cập nhật

4.7. Xóa dữ liệu

4.8. Cơ chế prepared câu lệnh SQL



## 4.1. Kết nối PHP với MySQL

- Các cách kết nối:
  - MySQLi (i là chữ viết tắt của improved)
    - Chỉ làm việc với cơ sở dữ liệu MySQL
    - Có 2 cách: object-oriented (hướng đối tượng) và Procedural (hướng thủ tục)
  - PDO (PHP Data Objects):
    - Làm việc trên nhiều hệ thống cơ sở dữ liệu khác nhau
- Các bước thao tác trên cơ sở dữ liệu:
  - Bước 1: Tạo kết nối
  - Bước 2: Mở kết nối dữ liệu
  - Bước 3: Tạo lệnh điều khiển truy vấn SQL
  - Bước 4: Thực thi lệnh
  - Bước 5: Đóng kết nối
  - Bước 6: In kết quả



## 4.1. Kết nối PHP với MySQL

- Kết nối MySQL bằng MySQLi Object-Oriented:

```
<?php
$conn = new mysqli('localhost', 'root', '');
if ($conn->connect_error) {
    die('Ket noi that bai: ' . $conn->connect_error);
}
echo 'Ket noi thanh cong';
?>
```

- Kết nối MySQL bằng MySQLi Procedural:

```
<?php
$conn = mysqli_connect('localhost', 'root', '', 'quanlyhoc_db');
if (!$conn) {
    die('Ket noi that bai: ' . mysqli_connect_error());
}
echo 'Ket noi thanh cong';
?>
```



## 4.1. Kết nối PHP với MySQL

- Kết nối MySQL bằng PDO:

```
<?php
try{
    $conn=new PDO('mysql:host=localhost;dbname=quanlyhoc_db','root','')
    );
    $conn->setAttribute(PDO::ATTR_ERRMODE,PDO::ERRMODE_EXCEPTION);
    echo 'Ket noi thanh cong';
}
catch(PDOException $e){
    echo 'Ket noi that bai: ' . $e->getMessage();
}
?>
```



## 4.1. Kết nối PHP với MySQL

- Ngắt kết nối:
  - MySQLi Object-Oriented:
    - `$conn->close()`
  - MySQLi Procedural:
    - `mysqli_close($conn)`
  - PDO:
    - `$conn = null`

## 4.2. Tạo cơ sở dữ liệu

- MySQLi Object-oriented:
  - Chọn cơ sở dữ liệu:
    - \$conn->select\_db(ten\_csdl)

```
<?php
$conn = new mysqli('localhost', 'root', '');
if ($conn->connect_error) {
    die("Kết nối thất bại: " . $conn->connect_error);
}
$sql = "CREATE DATABASE test";
if ($conn->query($sql)) {
    echo "Tạo database thành công";
} else {
    echo "Tạo database thất bại: " . $conn->error;
}
$conn->close();
?>
```



## 4.2. Tạo cơ sở dữ liệu

- MySQLi Procedural:
  - Chọn cơ sở dữ liệu:
    - `mysql_select_db(db_name, connection)`

```
<?php
$conn = mysqli_connect('localhost', 'root', '');
if (!$conn) {
    die('Ket noi that bai: ' . mysqli_connect_error());
}
$sql = 'create database test';
if (mysqli_query($conn, $sql)) {
    echo 'Tao DB thanh cong';
} else {
    echo 'Tao DB that bai' . mysqli_error($conn);
}
mysqli_close($conn);
?>
```



## 4.2. Tạo cơ sở dữ liệu

- PDO:
  - Chọn cơ sở dữ liệu:
    - \$conn->exec("use database\_name")

```
<?php
try {
    $conn = new PDO('mysql:host=localhost', 'root', '');
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    $sql = 'create database test4';
    $conn->exec($sql);
    echo 'Tao DB thanh cong';
} catch (PDOException $e) {
    echo 'Loi: ' . $e->getMessage();
}
$conn = null;
?>
```

## 4.3. Tạo bảng

- MySQLi Object-oriented:

```
<?php
$conn = new mysqli('localhost', 'root', '', 'test');
if ($conn->connect_error) {
    die('ket noi that bai: ' . $conn->connect_error);
}
$sql = 'create table News(
id int auto_increment primary key,
title nvarchar(30) not null,
content text,
add_date timestamp)
';
if ($conn->query($sql)) {
    echo 'Tao bang thanh cong';
} else
    echo 'Tao bang khong thanh cong: ' . $conn->error;
$conn->close();
?>
```



## 4.3. Tạo bảng

- MySQLi Procedural:

```
<?php
$conn = mysqli_connect('localhost', 'root', '', 'test');
if (!$conn) {
    die('ket noi that bai: ' . mysqli_connect_error());
}
$sql = 'create table News(
    id int auto_increment primary key,
    title nvarchar(30) not null,
    content text,
    add_date timestamp)
';
if (mysqli_query($conn, $sql)) {
    echo 'Tao bang thanh cong';
} else {
    echo 'Tao bang khong thanh cong: ' . mysqli_error($conn);
}
mysqli_close($conn);
?>
```



## 4.3. Tạo bảng

- PDO:

```
<?php
try {
    $conn = new PDO('mysql:host=localhost;dbname=test', 'root', '');
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    $sql = 'create table News(
        id int auto_increment primary key,
        title nvarchar(30) not null,
        content text,
        add_date timestamp)
        ';
    $conn->exec($sql);
    echo 'Tao bang thanh cong';
} catch (PDOException $e) {
    echo $e->getMessage();
}
$conn = null;
?>
```



## 4.3. Tạo bảng

- MySQLi Procedural:

```
<?php
$conn = mysqli_connect('localhost', 'root', '');
if (!$conn) {
    die('ket noi that bai: ' . mysqli_connect_error());
}
$sql = 'create database test';
if (mysqli_query($conn, $sql)) {
    echo 'Tao DB thanh cong';
    mysqli_select_db($conn, 'test5');
    $sql = 'create table News(id int auto_increment primary key, title nvarchar(30) not null, content text, add_date timestamp)';
    if (mysqli_query($conn, $sql)) {
        echo 'Tao bang thanh cong';
    } else {
        echo 'Tao bang khong thanh cong: ' . mysqli_error($conn);
    }
    mysqli_close($conn);
} else {
    echo 'Tao DB that bai: ' . mysqli_error($conn);
}
mysqli_close($conn);
?>
```

## 4.4. Truy vấn

- MySQLi Object-oriented:
  - Dùng vòng lặp while

```
<?php
$conn = new mysqli('localhost', 'root', '', 'test');
if ($conn->connect_error) {
    die('ket noi that bai: ' . $conn->connect_error);
}
$sql = 'select * from News';
$result = $conn->query($sql);
if ($result->num_rows > 0) {
    while ($row = $result->fetch_assoc()) {
        echo 'Title: ' . $row['title'] . ', Content: ' . $row['content']
    ] . ', Date: ' . $row['add_date'] . '<br>';
    }
} else {
    echo 'Khong co ban ghi nao';
}
$conn->close();
?>
```

## 4.4. Truy vấn

- MySQLi Object-oriented:
  - Dùng vòng lặp foreach

```
<?php
$conn = new mysqli('localhost', 'root', '', 'test');
if ($conn->connect_error) {
    die('ket noi that bai: ' . $conn->connect_error);
}
$sql = 'select * from News';
$result = $conn->query($sql);
if ($result->num_rows > 0) {
    foreach($result as $row){
        echo 'Title: ' . $row['title'] . ', Content: ' . $row['content']
    ] . ', Date: ' . $row['add_date'] . '<br>';
    }
} else {
    echo 'Khong co ban ghi nao';
}
$conn->close();
?>
```

## 4.4. Truy vấn

- MySQLi Procedural:
  - Dùng vòng lặp while

```
<?php
$conn = mysqli_connect('localhost', 'root', '', 'test');
if (!$conn) {
    die('ket noi that bai: ' . mysqli_connect_error());
}
$sql = 'select * from News';
$result = mysqli_query($conn, $sql);
if (mysqli_num_rows($result) > 0) {
    while ($row = mysqli_fetch_assoc($result)) {
        echo 'Title: ' . $row['title'] . ', Content: ' . $row['content']
    ] . ', Date: ' . $row['add_date'] . '<br>';
    }
} else
    echo 'Khong co ban ghi nao: ' . mysqli_error($conn);
mysqli_close($conn);
?>
```



## 4.4. Truy vấn

- PDO:
  - Dùng vòng lặp foreach

```
<?php
try {
    $conn = new PDO('mysql:host=localhost;dbname=test', 'root', '');
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    $sql = 'select * from News';
    $result = $conn->query($sql);
    if ($result->rowCount() > 0) {
        foreach ($result as $row) {
            echo 'Title: ' . $row['title'] . ', Content: ' . $row['content'] . ', Date: ' . $row['add_date'] . '<br>';
        }
    } else {
        echo 'Khong co ban ghi nao';
    }
} catch (PDOException $e) {
    echo $e->getMessage();
}
$conn = null;
?>
```



## 4.5. Thêm dữ liệu

- MySQLi Object-oriented:

```
<?php
$conn = new mysqli('localhost', 'root', '', 'test');
if ($conn->connect_error) {
    die('ket noi that bai: ' . $conn->connect_error);
}
$sql = 'insert into News(title, content, add_date) values("PL","Phap l
uat", NOW())';
if ($conn->query($sql) == true) {
    echo 'Them du lieu thanh cong';
} else
    echo 'Them du lieu khong thanh cong: ' . $conn->error;
$conn->close();
?>
```

## 4.5. Thêm dữ liệu

- Lấy ID bản ghi vừa insert
  - MySQLi Object-oriented:
    - \$conn->insert\_id
  - MySQLi Procedural:
    - mysqli\_insert\_id(\$conn)
  - PDO:
    - \$conn->lastInsertId()

```
.....  
$sql = 'insert into News(title, content, add_date) values("PL","Phap lu  
at", NOW())';  
if (mysqli_query($conn, $sql)) {  
    $last_ID=mysqli_insert_id($conn);  
    echo 'Them du lieu thanh cong, ID = ' . $last_ID;  
} else  
    echo 'Them du lieu khong thanh cong: ' . mysqli_error($conn);  
mysqli_close($conn);
```

## 4.5. Thêm dữ liệu

- Thêm nhiều bản ghi vào bảng
  - MySQLi Object-oriented:
    - `$conn->multi_query($sql)`
  - MySQLi Procedural:
    - `mysqli_multi_query($conn, $sql)`

```
$conn = new mysqli('localhost', 'root', '', 'test');
if ($conn->connect_error) {
    die('ket noi that bai: ' . $conn->connect_error);
}
$sql = 'insert into News(title, content, add_date) values("PL","Phap lu
at", NOW());';
$sql .= 'insert into News(title, content, add_date) values("PL","Phap l
uat", NOW())';
if ($conn->multi_query($sql) == true) {
    echo 'Them du lieu thanh cong';
} else
    echo 'Them du lieu khong thanh cong: ' . $conn->error;
$conn->close();
```



## 4.6. Cập nhật

- MySQLi Object-oriented:

```
<?php
$conn = new mysqli('localhost', 'root', '', 'test');
if ($conn->connect_error) {
    die('ket noi that bai: ' . $conn->connect_error);
}
$sql = 'update News set content = "Phap luat 1" where id = 1';
if ($conn->query($sql) == true) {
    echo 'Cap nhat du lieu thanh cong';
} else
    echo 'Cap nhat du lieu khong thanh cong: ' . $conn->error;
$conn->close();
?>
```

## 4.7. Xóa dữ liệu

- MySQLi Object-oriented:

```
<?php
$conn = new mysqli('localhost', 'root', '', 'test');
if ($conn->connect_error) {
    die('ket noi that bai: ' . $conn->connect_error);
}
$sql = 'delete from News where id = 1';
if ($conn->query($sql) == true) {
    echo 'Xoa du lieu thanh cong';
} else
    echo 'Xoa du lieu khong thanh cong: ' . $conn->error;
$conn->close();
?>
```

## 4.8. Cơ chế prepared câu lệnh SQL

- Mục đích: tránh lỗi SQL Injection
- Cơ chế: truyền tham số bằng biến ẩn danh, sau đó truyền giá trị cho các biến ẩn danh để chạy câu truy vấn  
→ an toàn hơn truyền trực tiếp tham số
- Ví dụ về lỗi SQL Injection:
  - Câu lệnh PHP:

```
$variable = $_POST['user_input'];  
mysql_query("INSERT INTO table('column') VALUES ('$unsafe_variable')");
```

- Truyền giá trị cho \$variable: `value'); DROP TABLE table;--`
- Câu SQL chuyển thành:

```
INSERT INTO table('column') VALUES('value'); DROP TABLE table;--')
```

## 4.8. Cơ chế prepared câu lệnh SQL

- MySQLi Object-oriented:
  - i: integer; d: double; s: string

```
<?php
$conn = new mysqli('localhost', 'root', '', 'test');
if ($conn->connect_error) {
    die('ket noi that bai: ' . $conn->connect_error);
}
$sql = 'insert into News(title, content) values(?, ?)';
$tmp = $conn->prepare($sql);
$tmp->bind_param('ss', $title, $content);
$title = "PL";
$content = "Phap luat";
$tmp->execute();
if ($tmp == true) {
    echo 'Them du lieu thanh cong';
} else
    echo 'Them du lieu khong thanh cong: ' . $conn->error;
$tmp->close();
$conn->close();
?>
```



## 4.8. Cơ chế prepared câu lệnh SQL

- PDO:

```
<?php
try {
    $conn = new PDO('mysql:host=localhost;dbname=test', 'root', '');
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    $sql = 'insert into News(title, content) values(:title, :content)';
    $tmp=$conn->prepare($sql);
    $tmp->bindParam(':title', $title);
    $tmp->bindParam(':content', $content);
    $title = "PL";
    $content = "Phap luat";
    $tmp->execute();
    if ($tmp == true) {
        echo 'Them du lieu thanh cong';
    } else
        echo 'Them du lieu khong thanh cong: ' . $conn->error;
} catch (PDOException $e) {
    echo $e->getMessage();
}
$tmp = null; $conn = null;
?>
```



# Câu hỏi thảo luận

---