

TÀI LIỆU THỰC HÀNH PHÁT TRIỂN PHẦN MỀM MÃ NGUỒN MỞ

GV: DƯƠNG QUỐC NAM



BÀI 2: TẠO CƠ SỞ DỮ LIỆU VÀ XÂY DỰNG CHỨC NĂNG CRUD VỚI PHP + MYSQL



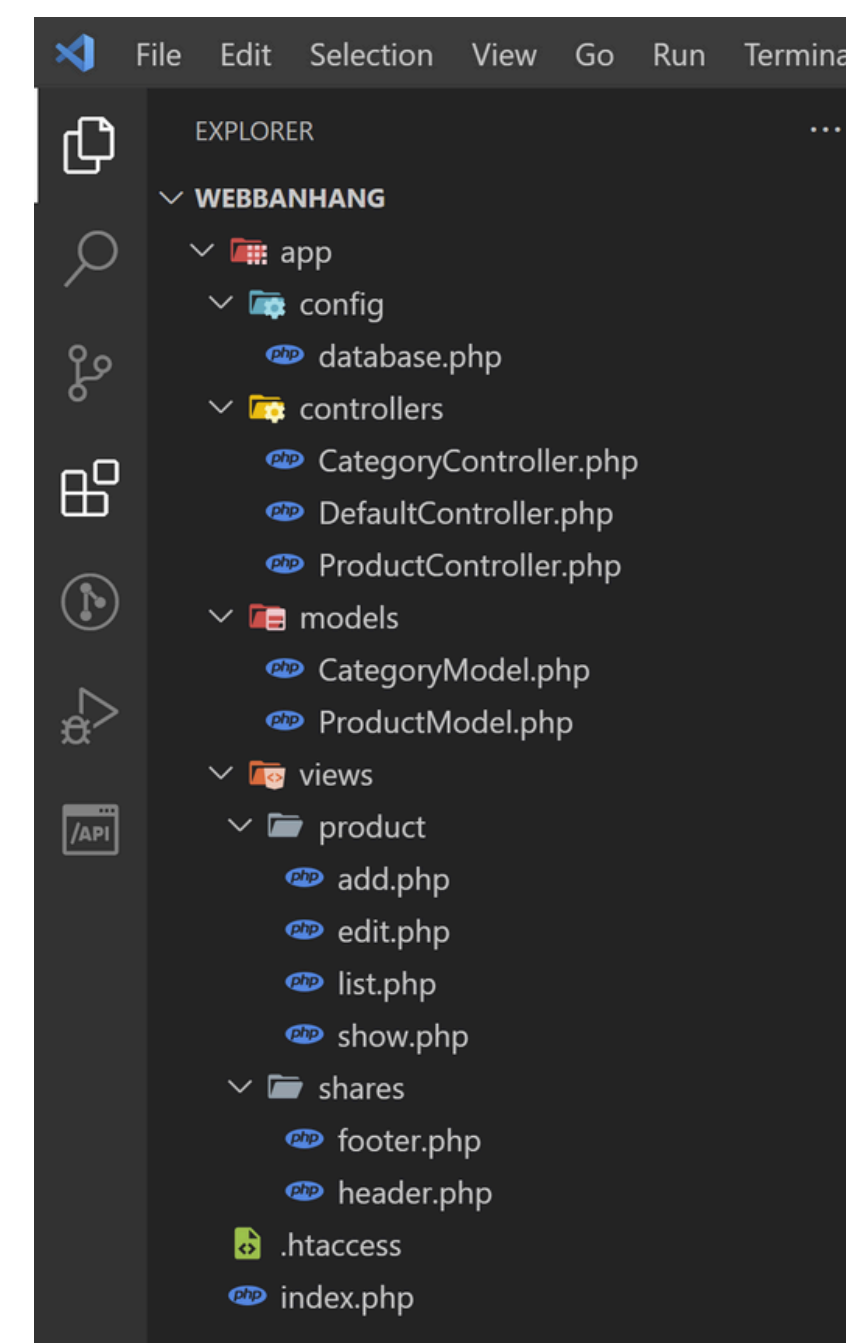
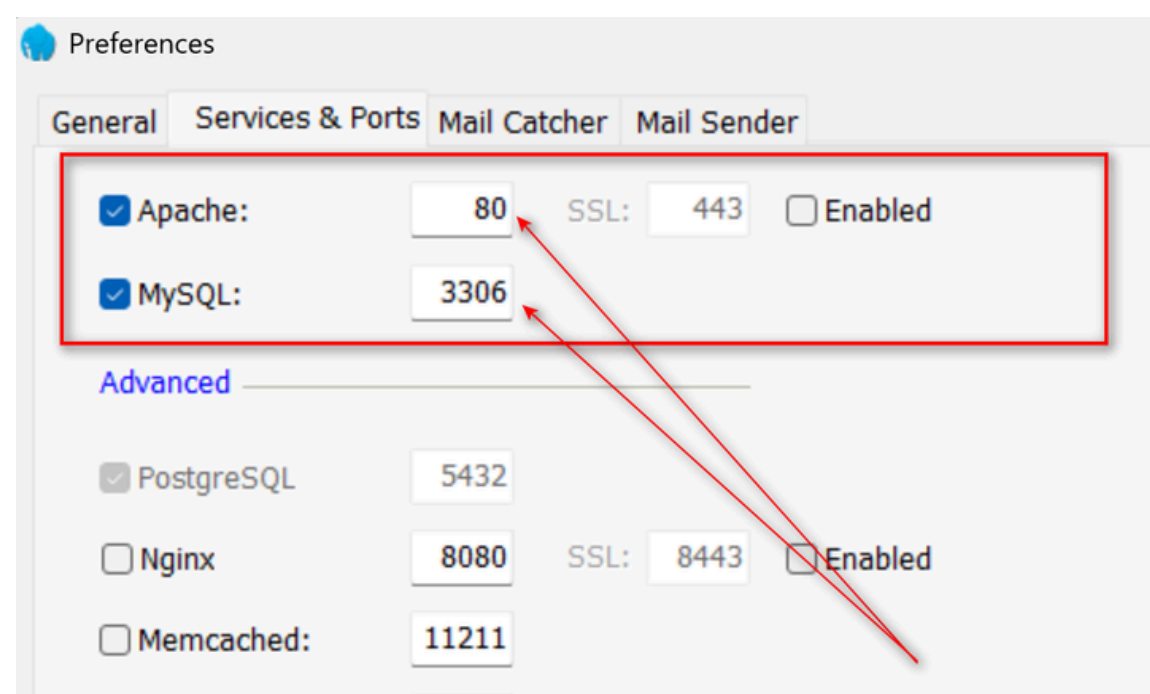
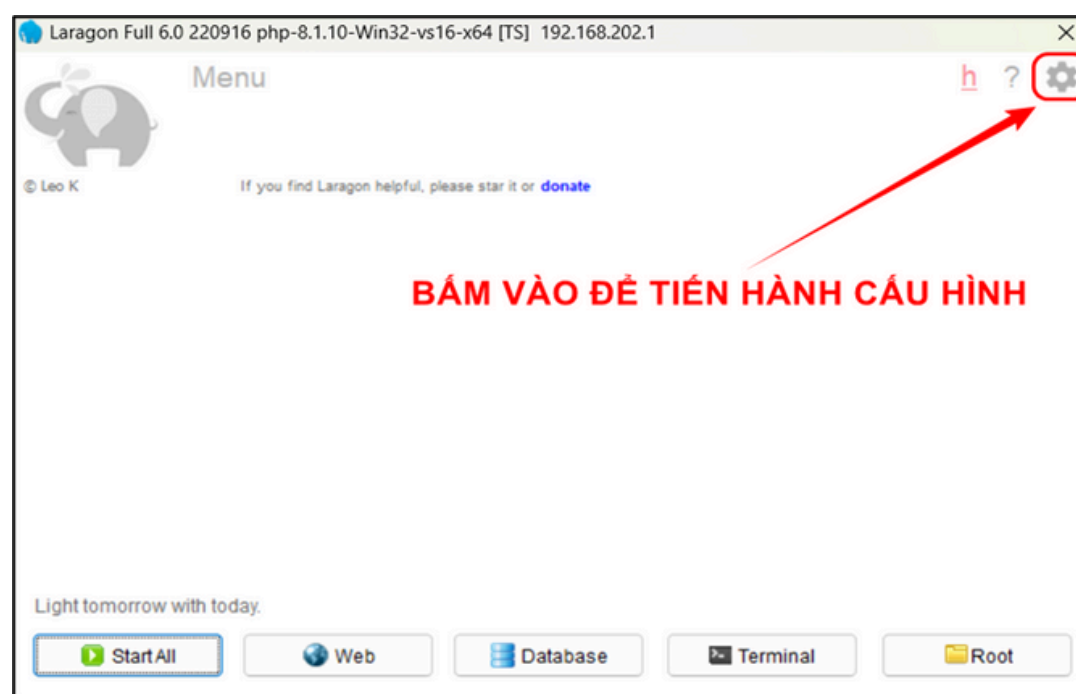
MỤC TIÊU CỦA BÀI THỰC HÀNH

- Làm việc với MySQL thông qua Laragon.
- Áp dụng kiến thức về mô hình MVC để thao tác dữ liệu thực.
- Rèn kỹ năng lập trình thực hành với hệ thống CRUD.



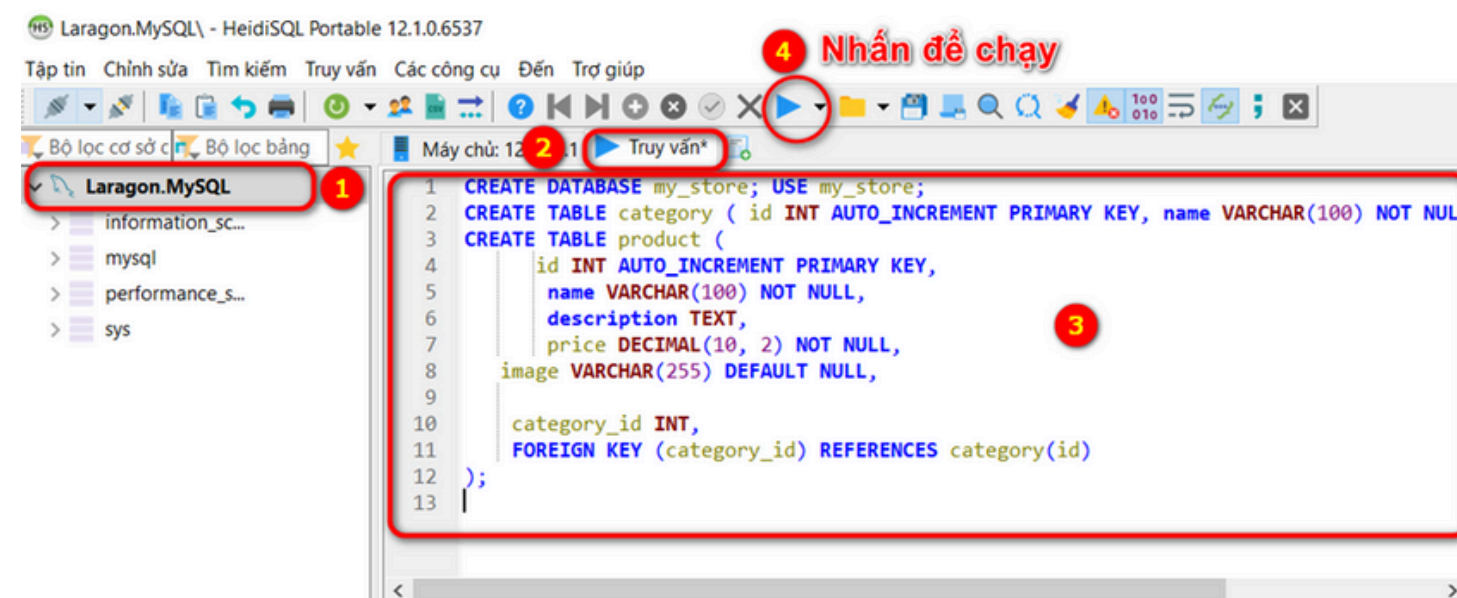
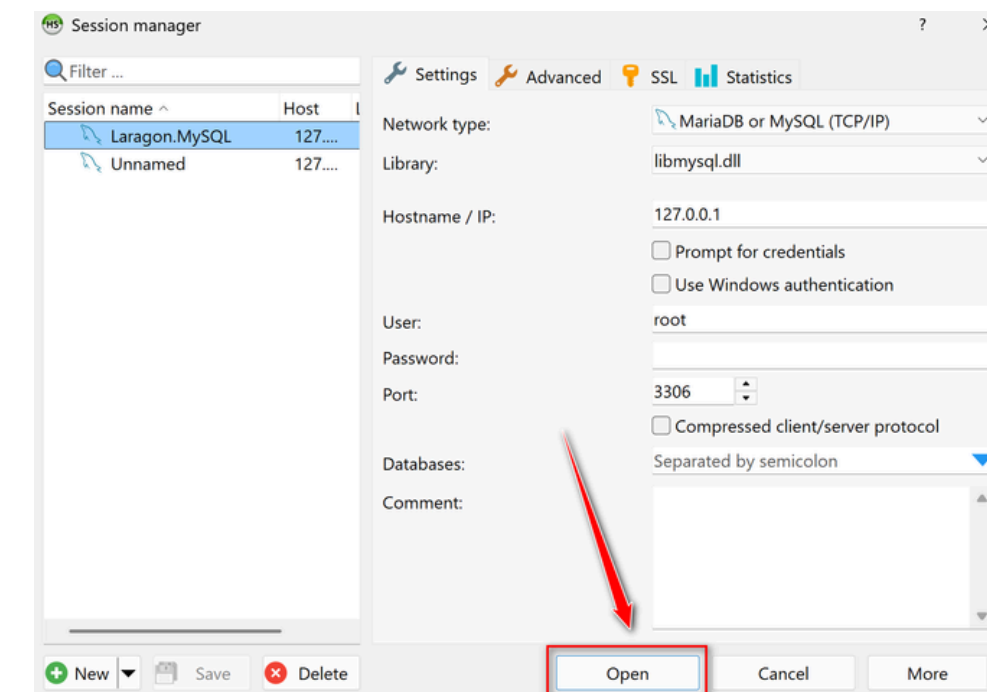
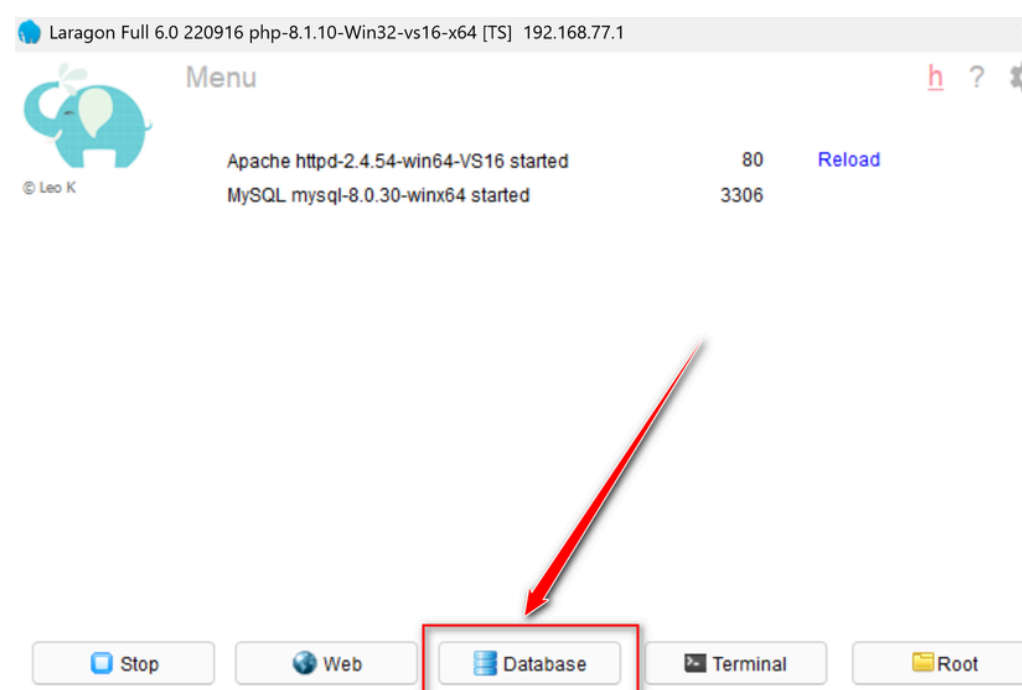
TẠO DỰ ÁN PHP KẾT NỐI CSDL

- Mở thư mục C:\laragon\www → tạo thư mục dự án **webbanhang**
- Mở bằng Visual Studio Code
- Tạo các thư mục models, views, controllers, config



TẠO CƠ SỞ DỮ LIỆU MYSQL

- Tạo CSDL tên **my_store**
- Tạo bảng products gồm:
 - id (int, auto_increment, primary key)
 - name (varchar 100)
 - description (text)
 - price (decimal 10,2)



CẤU HÌNH KẾT NỐI CƠ SỞ DỮ LIỆU TRONG PHP

Giải thích:

1. \$host = "*localhost*" => Máy chủ cơ sở dữ liệu.
2. \$db_name = "*my_store*" => Tên CSDL.
3. \$user = "*root*" => Tài khoản MySQL mặc định.
4. \$pass = "" => Mật khẩu (trống với Laragon).
5. \$conn = new PDO(...) => Tạo đối tượng kết nối.

Lưu ý:

- File database.php nên tách riêng để dễ quản lý và tái sử dụng.
- Biến **\$conn** nên khai báo global nếu dùng trong class đơn giản.
- Dùng *PDO* thay *mysqli* nếu muốn bảo mật và linh hoạt hơn ở các dự án lớn hơn.

```
1 <?php
2 class Database
3 {
4     private $host = "localhost"; ①
5
6     private $db_name = "my_store"; ②
7
8     private $username = "root"; ③
9
10    private $password = ""; ④
11
12    public $conn;
13
14    public function getConnection()
15    {
16        $this->conn = null;
17
18        try {
19            ⑤ $this->conn = new PDO("mysql:host=" . $this->host . ";dbname=" . $this->db_name, $this->username, $this->password);
20            $this->conn->exec("set names utf8");
21        } catch (PDOException $exception) {
22            echo "Connection error: " . $exception->getMessage();
23        }
24
25        return $this->conn;
26    }
27 }
28
```

LỖI - QUÊN GLOBAL \$CONN TRONG MODEL

Vấn đề gặp phải:

Khi viết hàm truy vấn trong ProductModel, quên khai báo **global \$conn** sẽ khiến kết nối MySQL không hoạt động, mặc dù đã *require database.php*.

Triệu chứng thường gặp:

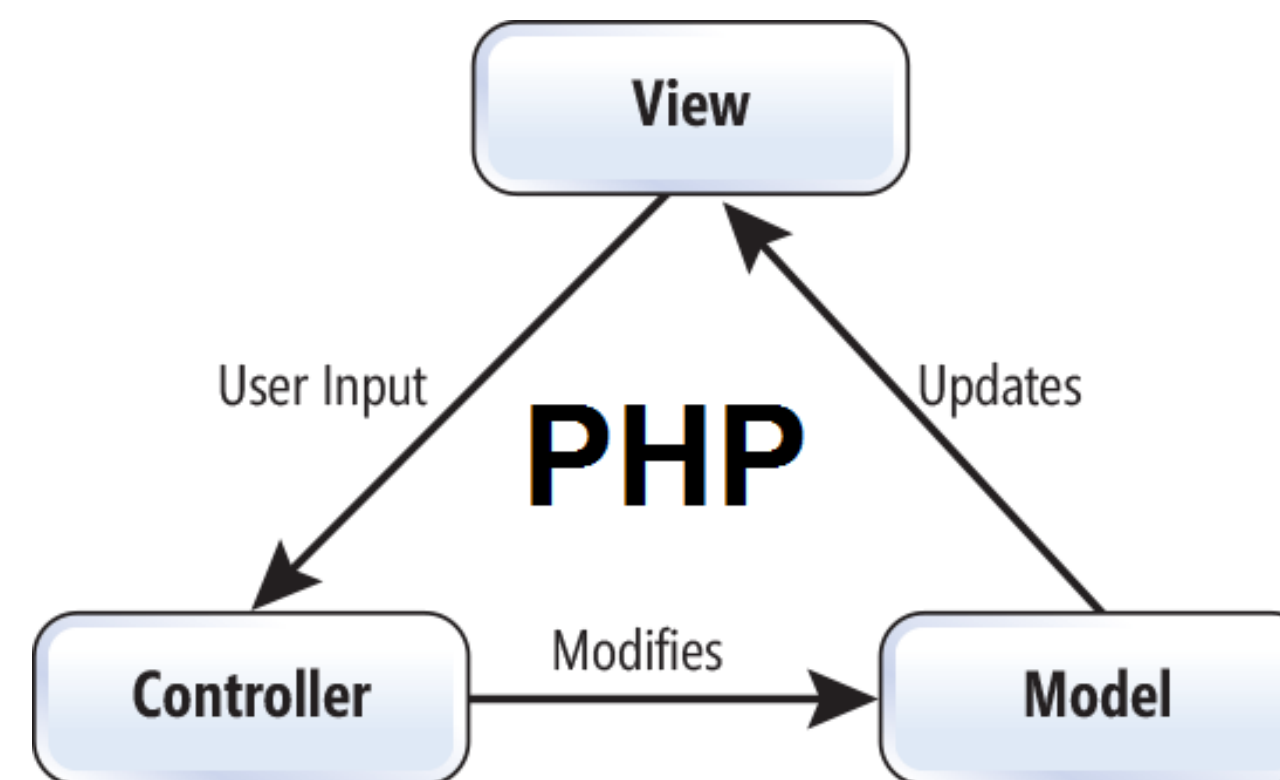
1. Truy vấn không trả dữ liệu.
2. Không báo lỗi rõ ràng (code vẫn chạy nhưng trả về null hoặc false).
3. Gặp lỗi như:
 - *"Call to a member function query() on null"*
 - *"Undefined variable: conn"*

```
1 public function getAll()
2 {
3     global $conn; // Phải khai báo để sử dụng biến $conn toàn cục
4     $sql = "SELECT * FROM products";
5     return $conn->query($sql);
6 }
```

GIỚI THIỆU VỀ MÔ HÌNH MVC

Giới thiệu tổng quát:

- Model: quản lý dữ liệu và xử lý logic.
- View: hiển thị giao diện người dùng.
- Controller: điều phối dữ liệu giữa **Model** và **View**



Link bài viết tham khảo: <https://viblo.asia/p/tim-hieu-mo-hinh-mvc-danh-cho-nguoi-moi-bat-dau-cau-truc-va-vi-du-V3m5WLDyKO7>

XÂY DỰNG LỚP PRODUCT MODEL

```
1 public function create($name, $desc, $price)
2 {
3     global $conn;
4     $sql = "INSERT INTO products(name, description, price)
5           VALUES ('$name', '$desc', $price)";
6     return $conn->query($sql);
7 }
```

```
1 public function update($id, $name, $desc, $price)
2 {
3     global $conn;
4     $sql = "UPDATE products SET name='$name', description='$desc', price=$price WHERE id=$id";
5     return $conn->query($sql);
6 }
```

```
1 public function delete($id)
2 {
3     global $conn;
4     $sql = "DELETE FROM products WHERE id = $id";
5     return $conn->query($sql);
6 }
```

Mục tiêu

- Tạo lớp *ProductModel* để xử lý dữ liệu từ bảng products.
- Thực hiện các thao tác CRUD bằng câu lệnh SQL trong PHP.

Lớp *ProductModel* dùng để làm gì?

- Giao tiếp với bảng products trong MySQL.
- Xử lý dữ liệu sản phẩm: thêm, sửa, xóa, lấy danh sách.
- Đóng vai trò là phần Model trong kiến trúc MVC.

XÂY DỰNG PRODUCT CONTROLLER

```
1 public function add()
2 {
3     if ($_SERVER['REQUEST_METHOD'] == 'POST') {
4         $name = $_POST['name'];
5         $desc = $_POST['description'];
6         $price = $_POST['price'];
7         $this->model->create($name, $desc, $price);
8         header("Location: /webbanhang/Product/list");
9         exit();
10    }
11    include 'app/views/product/add.php';
12 }
```

```
1 public function delete($id)
2 {
3     $this->model->delete($id);
4     header("Location: /webbanhang/Product/list");
5     exit();
6 }
```

```
1 public function edit($id)
2 {
3     if ($_SERVER['REQUEST_METHOD'] == 'POST') {
4         $this->model->update($id, $_POST['name'], $_POST['description'], $_POST['price']);
5         header("Location: /webbanhang/Product/list");
6         exit();
7     }
8     $product = $this->model->getById($id);
9     include 'app/views/product/edit.php';
10 }
```

🎯 Mục tiêu:

- Tiếp nhận yêu cầu từ người dùng thông qua URL.
- Gọi hàm từ ProductModel để xử lý dữ liệu.
- Điều hướng đến các giao diện View tương ứng.

🕒 Chức năng chính của Product Controller:

- Hiển thị danh sách sản phẩm.
- Lấy tất cả sản phẩm từ model → truyền vào view.
- Hiển thị form thêm + xử lý thêm mới sản phẩm.
- Hiển thị form sửa + cập nhật thông tin sản phẩm.
- Xóa sản phẩm theo ID.

CHỨC NĂNG UPLOAD HÌNH ẢNH SẢN PHẨM

Mục tiêu:

- Biết cách upload hình ảnh từ form HTML lên server bằng PHP.
- Lưu tên file ảnh vào cơ sở dữ liệu để hiển thị cùng sản phẩm.
- Kiểm tra định dạng và kích thước tệp ảnh khi upload.

Xử lý upload trong add() (Controller):

```
1 <?php
2
3 $imageName = '';
4 if (isset($_FILES['image']) && $_FILES['image']['error'] == 0) {
5     $targetDir = "public/uploads/";
6     $imageName = basename($_FILES["image"]["name"]);
7     move_uploaded_file($_FILES["image"]["tmp_name"], $targetDir . $imageName);
8 }
9 $this->model->create($name, $desc, $price, $imageName);
10
```

Form HTML thêm sản phẩm (add.php):

```
1 <form method="POST" enctype="multipart/form-data">
2     <input type="text" name="name">
3     <input type="file" name="image">
4     <button type="submit">Thêm</button>
5 </form>
6
```

Kiểm tra định dạng an toàn

```
1 <?php
2
3 $allowed = ['jpg', 'jpeg', 'png'];
4 $ext = strtolower(pathinfo($imageName, PATHINFO_EXTENSION));
5 if (!in_array($ext, $allowed)) {
6     die("Chỉ cho phép file ảnh JPG, JPEG, PNG!");
7 }
```

RÀNG BUỘC KHI UPLOAD HÌNH ẢNH SẢN PHẨM

🎯 Mục tiêu

- Đảm bảo an toàn và đúng định dạng khi người dùng upload hình ảnh.
- Tránh lỗi hệ thống và ngăn chặn upload mã độc.
- Hạn chế kích thước ảnh quá lớn gây tốn bộ nhớ.

```
1
2 $allowedTypes = ['jpg', 'jpeg', 'png', 'gif'];
3 $maxSize = 2 * 1024 * 1024; // 2MB
4
5 $file = $_FILES['image'];
6 $ext = strtolower(pathinfo($file['name'], PATHINFO_EXTENSION));
7
8 if (!in_array($ext, $allowedTypes)) {
9     die("Chỉ chấp nhận ảnh định dạng JPG, PNG, GIF.");
10 }
11
12 if ($file['size'] > $maxSize) {
13     die("Dung lượng ảnh vượt quá 2MB.");
14 }
```

📋 Các ràng buộc cần kiểm tra cho việc upload hình ảnh:

- Kiểm tra định dạng chỉ cho phép: *jpg, jpeg, png, gif*
- Kiểm tra *dung lượng tối đa 2MB là hợp lý (< 2 * 1024 * 1024 bytes)*
- Kiểm tra trùng tên file *tự động đổi tên nếu trùng (gắn timestamp hoặc random chuỗi)*
- Chống mã độc từ chối file có đuôi lạ như *.php, .exe, .js*

LỖI KHÔNG KIỂM TRA \$_FILES KHI UPLOAD ẢNH

Vấn đề gặp phải:

Khi người dùng không chọn hình ảnh, nếu *không kiểm tra kỹ \$_FILES*, hệ thống sẽ báo lỗi hoặc upload thất bại.

Triệu chứng thường gặp:

1. Xuất hiện lỗi:

- *"Undefined index: image"*
- *"Trying to access array offset on value of type null"*

2. Hình ảnh không được lưu, hoặc gây gián đoạn cả quá trình thêm/sửa sản phẩm.

```
1  $imageName = '';  
2  // Kiểm tra xem người dùng có chọn file không và không có lỗi trong quá trình upload  
3  if (isset($_FILES['image']) && $_FILES['image']['error'] == 0) {  
4      $targetDir = "public/uploads/";  
5      $imageName = basename($_FILES["image"]["name"]);  
6      move_uploaded_file($_FILES["image"]["tmp_name"], $targetDir . $imageName);  
7  }
```


CÁCH HIỂN THỊ TỪNG GIAO DIỆN

- *Hiển thị danh sách sản phẩm* => `http://localhost:<port>/<ten_du_an>/Product/list`
- *Thêm mới sản phẩm* => `http://localhost:<port>/<ten_du_an>/Product/add`
- *Sửa sản phẩm* => `http://localhost:<port>/<ten_du_an>/Product/edit/<ma_san_pham>`
- *Xoá sản phẩm* => `http://localhost:<port>/<ten_du_an>/Product/delete/<ma_san_pham>`

