

Giới Thiệu Ngôn Ngữ Lập Trình PHP

Tài liệu này trình bày tổng quan về PHP, từ khái niệm cơ bản đến thiết lập môi trường phát triển. Nội dung bao gồm cú pháp cơ bản, cách hoạt động của PHP, và hướng dẫn cài đặt môi trường với Laragon.

Lịch Sử Phát Triển Của PHP

PHP ra đời năm 1994 bởi Rasmus Lerdorf, ban đầu chỉ là một tập script CGI đơn giản để theo dõi lượt truy cập trang web cá nhân, với tên gọi "Personal Home Page Tools" và sau đó là "PHP/FI".

Năm 1997, Zeev Suraski và Andi Gutmans viết lại cú pháp và phát hành PHP 3.0, đánh dấu bước ngoặt quan trọng. PHP 4.0 ra mắt năm 2000 với nền tảng Zend Engine, cải thiện hiệu suất và bổ sung nhiều tính năng mới.

PHP 5.0 (2004) mang đến cải tiến mạnh về lập trình hướng đối tượng. Năm 2015, PHP 7.0 tăng hiệu suất gấp đôi và giảm tiêu thụ bộ nhớ. Hiện tại, PHP đã phát triển đến phiên bản 8.x với nhiều cải tiến về tốc độ và tính năng.

Sau hơn 25 năm, PHP đã trở thành một trong những ngôn ngữ phổ biến nhất cho phát triển web, chiếm khoảng 79% thị phần trang web toàn cầu. Facebook, Wikipedia, WordPress và hàng triệu trang web khác đều sử dụng PHP, minh chứng cho sự ổn định và khả năng mở rộng của ngôn ngữ này.

1994: Ra đời

Rasmus Lerdorf tạo PHP như một tập script CGI đơn giản

1

2

1997: PHP 3.0

Suraski và Gutmans viết lại cú pháp, đổi tên thành "PHP: Hypertext Preprocessor"

2000: PHP 4.0

3

Ra mắt với Zend Engine, cải thiện hiệu suất

4

2004: PHP 5.0

Cải tiến mạnh về lập trình hướng đối tượng

2015: PHP 7.0

5

Tăng hiệu suất gấp đôi, giảm tiêu thụ bộ nhớ

6

2020: PHP 8.0

Thêm JIT compiler và union types

Định Nghĩa và Đặc Điểm Của PHP

PHP (PHP: Hypertext Preprocessor) là ngôn ngữ lập trình kịch bản mã nguồn mở được thiết kế cho phát triển web và có thể nhúng vào HTML. Khác với JavaScript (xử lý phía client), PHP được xử lý trên máy chủ (server-side), tạo ra HTML và gửi đến trình duyệt người dùng.

1 Ngôn Ngữ Kịch Bản Máy Chủ

PHP xử lý hoàn toàn trên máy chủ web. Khi truy cập trang PHP, máy chủ thực thi mã, tạo HTML và gửi kết quả về trình duyệt. Người dùng không thấy mã nguồn gốc, chỉ nhận kết quả đã xử lý.

2 Tính Linh Hoạt

PHP có cú pháp dễ học, đặc biệt với người đã biết C, Java hoặc JavaScript. Cho phép lập trình theo nhiều phong cách: cấu trúc, hướng đối tượng, hoặc kết hợp cả hai.

3 Tích Hợp Với Cơ Sở Dữ Liệu

PHP tích hợp tốt với nhiều hệ quản trị cơ sở dữ liệu như MySQL, PostgreSQL, MongoDB, Oracle và SQL Server, giúp xây dựng ứng dụng web động đơn giản và hiệu quả.

4 Cộng Đồng Lớn Mạnh

Với hơn 25 năm phát triển, PHP có cộng đồng rộng lớn cùng nhiều tài liệu, thư viện và framework như Laravel, Symfony, CodeIgniter giúp đẩy nhanh quá trình phát triển.

Ví Dụ Cú Pháp PHP Cơ Bản:

Ví dụ trên minh họa cách PHP được nhúng vào HTML qua thẻ `<?php ... ?>`. Mã PHP được xử lý trên máy chủ, và trình duyệt chỉ nhận kết quả HTML đã tạo ra - đặc điểm quan trọng giúp PHP trở thành lựa chọn hàng đầu cho phát triển web động.

```
<!DOCTYPE html>
<html>
<head>
  <title>Ví dụ PHP</title>
</head>
<body>
  <h1>Xin chào thế giới PHP!</h1>

  <?php
    // Khai báo biến
    $ten = "Nguyễn Văn A";
    $tuoi = 20;

    // In ra thông tin
    echo "<p>Tên tôi là: " . $ten . "</p>";
    echo "<p>Tôi " . $tuoi . " tuổi.</p>";

    // Cấu trúc điều kiện
    if ($tuoi >= 18) {
      echo "<p>Tôi đã đủ tuổi trưởng thành.</p>";
    } else {
      echo "<p>Tôi chưa đủ tuổi trưởng thành.</p>";
    }
  ?>
</body>
</html>
```

Ưu Và Nhược Điểm Của PHP

Ưu Điểm

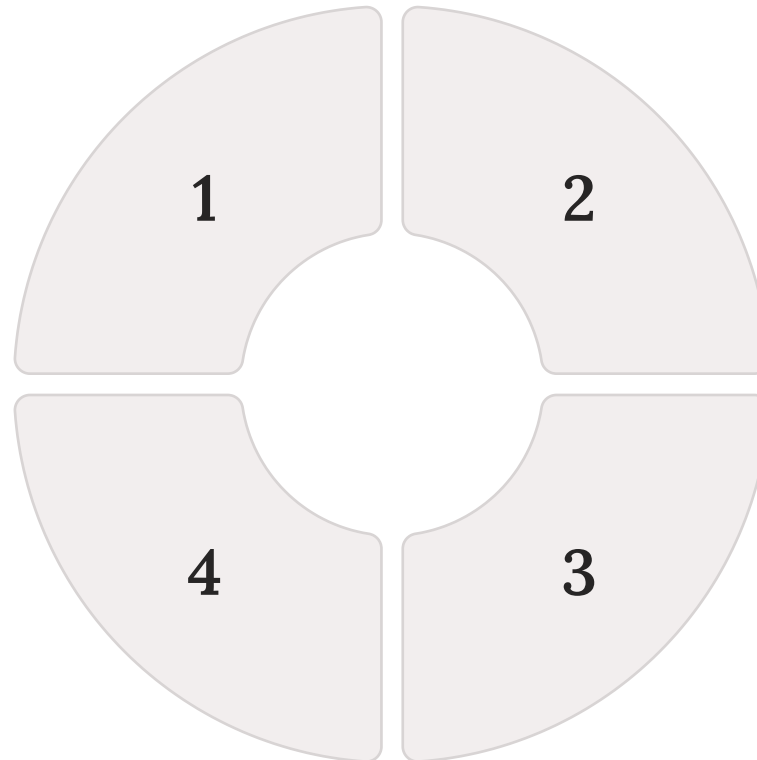
- Dễ học và sử dụng cho người mới
- Cú pháp đơn giản, linh hoạt
- Tài liệu phong phú, cộng đồng lớn
- Tích hợp tốt với nhiều CSDL
- Đa nền tảng (cross-platform)
- Nhiều framework mạnh (Laravel, Symfony)
- Chi phí thấp, dễ triển khai
- Được hỗ trợ bởi hầu hết nhà cung cấp hosting

Nhược Điểm

- Hiệu suất thấp hơn Go, Rust
- Rủi ro bảo mật nếu cấu hình sai
- Thiếu nhất quán trong một số hàm
- Quản lý lỗi chưa tối ưu
- Khó bảo trì nếu không theo chuẩn
- Không được thiết kế cho ứng dụng quy mô lớn

Máy Chủ Web
Tương thích với Apache, Nginx,
IIS và chạy trên mọi hệ điều
hành

CMS
Nền tảng cho WordPress,
Joomla, Drupal, Magento



Cơ Sở Dữ Liệu

Hỗ trợ đa dạng CSDL: MySQL,
PostgreSQL, MongoDB, Oracle,
SQLite

Framework

Laravel, Symfony, CodeIgniter,
CakePHP, Yii giúp phát triển
nhANH chóng

Dù có nhược điểm, PHP vẫn phổ biến nhờ dễ tiếp cận và cộng đồng lớn. Các phiên bản PHP 7.x và 8.x đã cải thiện hiệu suất đáng kể. Với framework hiện đại, PHP có thể xây dựng ứng dụng web quy mô lớn, bảo mật và mở rộng tốt.

Các Ứng Dụng Phổ Biến Của PHP

Hệ Thống Quản Lý Nội Dung (CMS)

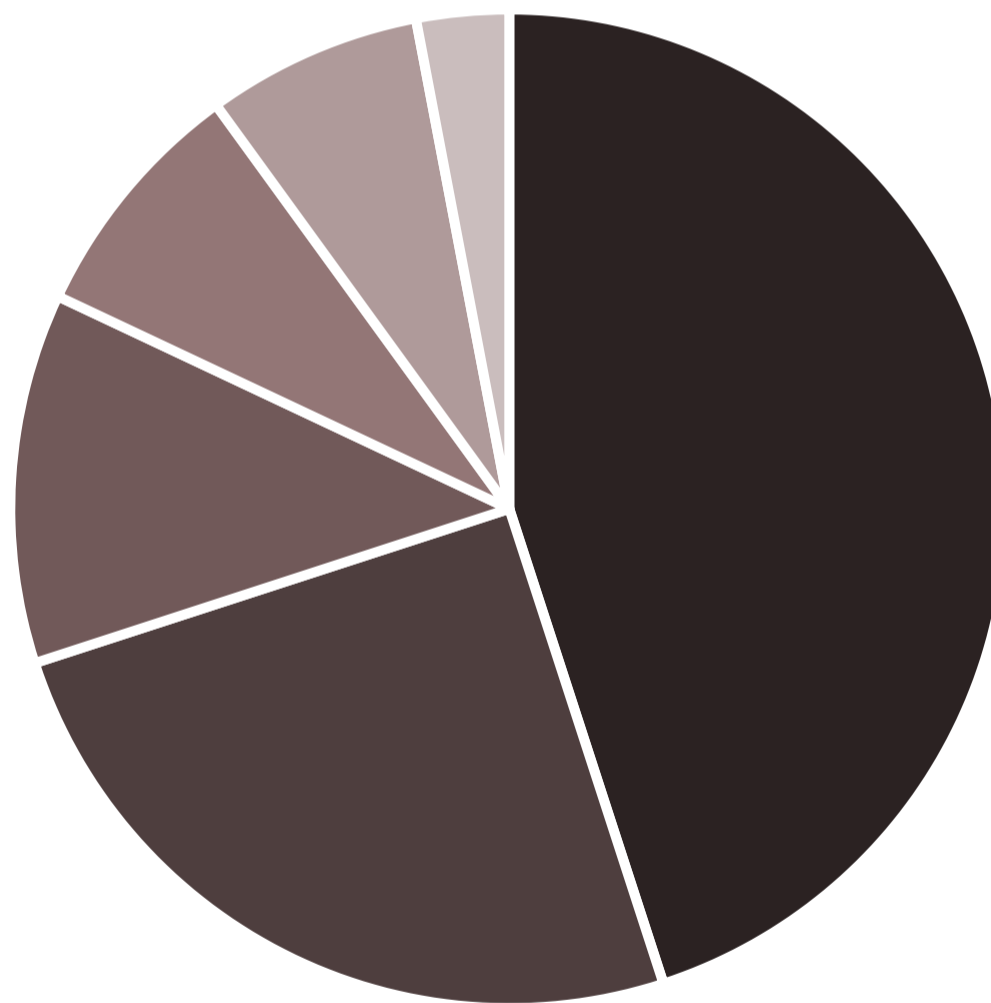
PHP là nền tảng cho các CMS phổ biến như WordPress (chiếm hơn 40% website toàn cầu), Joomla và Drupal. Các hệ thống này giúp người dùng không có kiến thức kỹ thuật vẫn có thể quản lý website dễ dàng.

Thương Mại Điện Tử

Các nền tảng thương mại điện tử như Magento, WooCommerce và PrestaShop đều sử dụng PHP. Chúng cung cấp các tính năng thiết yếu: quản lý sản phẩm, giỏ hàng, thanh toán và vận chuyển.

Mạng Xã Hội

Facebook ban đầu được xây dựng bằng PHP và sau đó phát triển ngôn ngữ Hack dựa trên PHP. Nhiều mạng xã hội khác cũng sử dụng PHP trong hệ thống của họ.



■ CMS ■ Thương mại điện tử ■ Ứng dụng doanh nghiệp ■ Mạng xã hội ■ API & Backend ■ Khác

PHP còn được sử dụng trong các hệ thống CRM, LMS, diễn đàn trực tuyến, cổng thông tin, ứng dụng doanh nghiệp và API RESTful.

Các framework hiện đại như Laravel, Symfony và CodeIgniter đã giúp PHP vượt qua giới hạn ban đầu, trở thành nền tảng cho các ứng dụng web phức tạp. Chúng cung cấp kiến trúc MVC, công cụ ORM, hệ thống xác thực và phân quyền, giúp lập trình viên xây dựng ứng dụng web nhanh chóng và hiệu quả.

Cài Đặt Môi Trường Phát Triển PHP

Để phát triển ứng dụng PHP, bạn cần thiết lập môi trường phát triển phù hợp. Bài này sẽ hướng dẫn cài đặt Laragon - giải pháp tích hợp dành cho Windows, thân thiện với người mới bắt đầu.



Tải Laragon

Truy cập <https://laragon.org> và tải phiên bản phù hợp. Khuyến nghị chọn phiên bản Full để có đầy đủ công cụ cần thiết.



Cài Đặt

Chạy tệp cài đặt và làm theo hướng dẫn. Laragon sẽ tự động cài đặt Apache, MySQL, PHP và các công cụ cần thiết trong khoảng 5-10 phút.



Cấu Hình

Khởi động Laragon và điều chỉnh các cài đặt như phiên bản PHP, cổng web server qua menu Preferences.



Khởi Động Dịch Vụ

Nhấn "Start All" để khởi động Apache và MySQL. Đèn chỉ báo sẽ chuyển từ đỏ sang xanh khi hoạt động thành công.

Cấu Trúc Thư Mục Của Laragon

- **root:** Thư mục gốc chứa tất cả trang web (C:\laragon\www)
- **bin:** Chứa các chương trình thực thi
- **etc:** Chứa tệp cấu hình
- **tmp:** Chứa tệp tạm thời
- **usr:** Chứa công cụ bổ sung

Tạo dự án mới bằng cách tạo thư mục con trong www.

Ví dụ: tạo "myproject" trong www để truy cập qua <http://myproject.test> (Laragon tự cấu hình virtual hosts). Laragon cũng cung cấp HeidiSQL (quản lý CSDL), Terminal và Quick create (tạo nhanh dự án từ template).

Laragon vượt trội hơn các bộ công cụ WAMP thông thường với khả năng chuyển đổi nhanh giữa các phiên bản PHP, tự động quản lý virtual hosts và hỗ trợ nhiều framework, CMS. Với thiết kế hiệu suất cao và tiêu tốn ít tài nguyên, Laragon phù hợp cho cả máy tính cấu hình thấp.

Kiểm Tra Cài Đặt PHP

Để kiểm tra cài đặt PHP, tạo tệp **info.php** trong thư mục www với nội dung:

```
<?php
    phpinfo();
?>
```

Truy cập <http://localhost/info.php>. Nếu thấy trang hiển thị thông tin cấu hình PHP, bạn đã cài đặt thành công.

Phân Tích Tập Cấu Hình PHP.INI

Vị Trí Tập php.ini Trong Laragon

Trong Laragon, tập php.ini nằm tại:

```
C:\laragon\bin\php\php-[version]\php.ini
```

Ví dụ: C:\laragon\bin\php\php-8.1.10-Win32-vs16-x64\php.ini

Để chỉnh sửa: Nhấp chuột phải vào biểu tượng Laragon > PHP > php.ini.

Các Thiết Lập Quan Trọng

- **memory_limit:** Giới hạn bộ nhớ cho script PHP
- **max_execution_time:** Thời gian tối đa script có thể chạy
- **upload_max_filesize:** Kích thước tối đa của tệp tải lên
- **post_max_size:** Kích thước tối đa của dữ liệu POST
- **display_errors:** Hiển thị lỗi (On/Off)
- **error_reporting:** Mức độ báo cáo lỗi
- **date.timezone:** Múi giờ mặc định

Việc tùy chỉnh php.ini cần được thực hiện phù hợp với từng môi trường. Trong môi trường phát triển, nên bật hiển thị lỗi và tăng giới hạn để dễ debug. Với môi trường sản xuất, cần tắt hiển thị lỗi và thiết lập giới hạn hợp lý cho an toàn và hiệu suất.

Sau khi thay đổi php.ini, khởi động lại máy chủ web bằng cách nhấn "Stop All" rồi "Start All", hoặc đơn giản là nhấn "Reload" trên giao diện Laragon.

Cài Đặt Khuyến Nghị Cho Môi Trường Phát Triển

```
; Hiển thị tất cả lỗi
display_errors = On
error_reporting = E_ALL

; Tăng giới hạn bộ nhớ
memory_limit = 256M

; Tăng thời gian thực thi
max_execution_time = 300

; Tăng kích thước tệp tải lên
upload_max_filesize = 64M
post_max_size = 64M

; Thiết lập múi giờ Việt Nam
date.timezone = "Asia/Ho_Chi_Minh"

; Bật opcode để tăng hiệu suất
opcache.enable = 1
opcache.memory_consumption = 128
```

Tạo Dự Án PHP Đầu Tiên Với Laragon

Sau khi cài đặt Laragon, bạn bắt đầu tạo dự án PHP đầu tiên. Laragon cho phép tạo nhanh dự án từ nhiều template hoặc từ đầu.

Tạo Dự Án Từ Đầu

1. Đảm bảo Laragon đang chạy
2. Tạo thư mục mới trong C:\laragon\www (ví dụ: "myproject")
3. Tạo tệp index.php trong thư mục
4. Thêm mã PHP của bạn vào tệp
5. Truy cập dự án qua <http://myproject.test>

Tạo Dự Án Bằng Quick Create

1. Nhấp chuột phải vào biểu tượng Laragon
2. Chọn Quick app > PHP
3. Nhập tên dự án
4. Laragon tự động tạo thư mục với các tệp cần thiết

Laragon cũng hỗ trợ tạo nhanh dự án với các framework như Laravel, Symfony, CodeIgniter, hoặc CMS như WordPress - tiện lợi cho các dự án lớn.

Cấu Trúc Dự Án PHP Cơ Bản

- **index.php**: Điểm khởi đầu ứng dụng
- **css/**: Chứa tệp CSS
- **js/**: Chứa tệp JavaScript
- **images/**: Chứa hình ảnh
- **includes/**: Chứa tệp PHP tái sử dụng
- **config.php**: Tệp cấu hình

Ví Dụ Mã Cho index.php

```
<!DOCTYPE html>
<html lang="vi">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Dự Án PHP Đầu Tiên</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      line-height: 1.6;
      margin: 0;
      padding: 20px;
      background-color: #f5f5f5;
    }
    .container {
      max-width: 800px;
      margin: 0 auto;
      background-color: white;
      padding: 20px;
      border-radius: 5px;
      box-shadow: 0 0 10px rgba(0,0,0,0.1);
    }
    h1 {
      color: #2c3e50;
    }
  </style>
</head>
<body>
  <div class="container">
    <h1>Xin chào từ PHP!</h1>

    <?php
      // Thiết lập múi giờ
      date_default_timezone_set('Asia/Ho_Chi_Minh');

      // Hiển thị ngày giờ hiện tại
      echo "<p>Hôm nay là: " . date("d/m/Y H:i:s") . "</p>";

      // Hiển thị thông tin PHP
      echo "<p>Bạn đang sử dụng PHP phiên bản: " . phpversion() . "</p>";

      // Tạo một mảng đơn giản
      $languages = array("HTML", "CSS", "JavaScript", "PHP", "MySQL");

      // Hiển thị mảng sử dụng vòng lặp
      echo "<h2>Các ngôn ngữ web phổ biến:</h2>";
      echo "<ul>";
      foreach ($languages as $lang) {
        echo "<li>" . $lang . "</li>";
      }
      echo "</ul>";

      // Sử dụng điều kiện
      $hour = (int)date("H");
      echo "<p>";
      if ($hour < 12) {
        echo "Chào buổi sáng!";
      } elseif ($hour < 18) {
        echo "Chào buổi chiều!";
      } else {
        echo "Chào buổi tối!";
      }
      echo "</p>";
    ?>
  </div>
</body>
</html>
```

1 Hiểu Về Môi Trường Phát Triển

Laragon cung cấp môi trường hoàn chỉnh (Apache, MySQL, PHP) với virtual hosts tự động, giúp bạn tập trung vào phát triển ứng dụng.

2 Học Cú Pháp PHP Cơ Bản

Nắm vững cú pháp PHP: biến, mảng, điều kiện, vòng lặp, hàm. Thực hành với ví dụ đơn giản trước khi học khái niệm phức tạp.

3 Xử Lý Lỗi Và Debug

Đọc hiểu thông báo lỗi PHP. Sử dụng `var_dump()`, `print_r()` để kiểm tra biến. Cấu hình hiển thị lỗi trong môi trường phát triển.

4 Mở Rộng Kiến Thức

Sau khi nắm vững cơ bản, tiếp tục với chủ đề nâng cao: xử lý biểu mẫu, kết nối cơ sở dữ liệu, xác thực người dùng và thư viện phổ biến.