PHP Chạy File EXE với Tham Số và Hiển Thị Kết Quả

# 1. Tạo file Execute

Bạn có thể tạo chương trình bằng bất cứ ngôn ngữ lập trình nào. Dưới đây là hướng dẫn tạo file thực thi với ba ngôn ngữ khác nhau: C++, Python, và C#.

## 1.1. C++

**Tạo file và viết mã nguồn**

1. Mở trình soạn thảo mã nguồn (như Visual Studio Code hoặc bất kỳ trình soạn thảo nào bạn thích).
2. Tạo một file mới và đặt tên là sum.cpp.
3. Viết mã nguồn vào file sum.cpp.

**Biên dịch chương trình**

1. Mở terminal hoặc command prompt.
2. Điều hướng đến thư mục chứa file sum.cpp.
3. Sử dụng trình biên dịch g++ để biên dịch chương trình: g++ sum.cpp -o sum.exe

**Mã nguồn chi tiết**

| #include <iostream>  #include <fstream>  #include <cstdlib>  using namespace std;  int main(int argc, char\* argv[]) {  // Kiểm tra xem có đúng hai đối số được cung cấp không  if (argc != 3) {  cout << "Usage: sum.exe <num1> <num2>" << endl;  return 1;  }  // Chuyển đổi hai đối số thành số nguyên  int num1 = atoi(argv[1]);  int num2 = atoi(argv[2]);  // Kiểm tra chuyển đổi thành công  if (num1 == 0 && argv[1][0] != '0') {  cout << "First argument must be a valid integer." << endl;  return 1;  }  if (num2 == 0 && argv[2][0] != '0') {  cout << "Second argument must be a valid integer." << endl;  return 1;  }  // Tính tổng của hai số  int sum\_result = num1 + num2;  // Ghi kết quả vào file "result.txt"  ofstream outfile("result.txt");  if (!outfile) {  cout << "Failed to open result.txt" << endl;  return 1;  }  outfile << sum\_result;  outfile.close();  return 0;  } |
| --- |

## 1.2. Python

##### **Tạo file và viết mã nguồn**

1. Mở trình soạn thảo mã nguồn.
2. Tạo một file mới và đặt tên là sum.py.
3. Viết mã nguồn vào file sum.py.

##### **Chuyển đổi thành file thực thi**

1. Cài đặt PyInstaller nếu chưa có: pip install pyinstaller
2. Sử dụng PyInstaller để tạo file thực thi: pyinstaller --onefile sum.py
3. File thực thi sẽ được tạo trong thư mục dist với tên sum.exe.

**Mã nguồn chi tiết**

| import sys  if len(sys.argv) != 3:  print("Usage: sum.py <num1> <num2>")  sys.exit(1)  try:  num1 = int(sys.argv[1])  num2 = int(sys.argv[2])  except ValueError:  print("Both arguments must be valid integers.")  sys.exit(1)  sum\_result = num1 + num2  with open("result.txt", "w") as f:  f.write(str(sum\_result)) |
| --- |

## 1.3. C#

##### **Tạo project và viết mã nguồn**

1. Mở Visual Studio.
2. Tạo một project mới:
   * Chọn "Create a new project".
   * Chọn "Console App (.NET Core)" hoặc "Console App (.NET Framework)".
   * Đặt tên cho project và chọn thư mục lưu trữ.
3. Viết mã nguồn vào file Program.cs.

##### **Biên dịch chương trình**

1. Nhấn Ctrl + Shift + B để build project.
2. File thực thi sẽ được tạo trong thư mục bin\Debug\netcoreappX.X\ (với X.X là phiên bản .NET bạn sử dụng).
3. PHP chạy file EXE kèm truyền 2 tham số, đọc file kết quả và echo ra ngoài trình duyệt

**Mã nguồn chi tiết**

| using System;  using System.IO;  class SumProgram  {  static void Main(string[] args)  {  if (args.Length != 2)  {  Console.WriteLine("Usage: sum.exe <num1> <num2>");  return;  }  int num1 = Int32.Parse(args[0]);  int num2 = Int32.Parse(args[1]);  int sum = num1 + num2;  File.WriteAllText("result.txt", sum.ToString());  }  } |
| --- |

# 2. PHP chạy file EXE kèm truyền 2 tham số, đọc file kết quả và echo ra ngoài trình duyệt:

Dưới đây là hướng dẫn chi tiết về cách tạo và sử dụng một trang web đơn giản để tính tổng hai số bằng cách chạy một file thực thi (EXE) từ PHP.

## 2.1. Giao diện trang web

Mã HTML dưới đây tạo một giao diện đơn giản sử dụng Bootstrap, trong đó người dùng có thể nhập hai số và nhận kết quả tính tổng sau khi nhấn nút "Submit".

| <!DOCTYPE html>  <html>  <head>  <title>Form Example with Bootstrap</title>  <link href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css" rel="stylesheet">  <style>  .container-form {  display: flex;  justify-content: center;  align-items: center;  height: 100vh;  padding: 20px;  }  .form-container {  background-color: #f8f9fa;  padding: 20px;  border-radius: 8px;  box-shadow: 0 4px 8px rgba(0,0,0,0.1);  }  .result-container {  margin-top: 20px;  }  </style>  </head>  <body>  <div class="container-form">  <div class="form-container">  <form id="sum-form">  <h2 class="text-center mb-4">Sum Calculator</h2>  <div class="form-group">  <label for="num1">Number 1:</label>  <input type="number" id="num1" name="num1" class="form-control" required>  </div>  <div class="form-group">  <label for="num2">Number 2:</label>  <input type="number" id="num2" name="num2" class="form-control" required>  </div>  <button type="submit" class="btn btn-primary btn-block">Submit</button>  </form>  <div id="result-container" class="result-container"></div>  </div>  </div>  <script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>  <script>  $(document).ready(function() {  $('#sum-form').submit(function(event) {  event.preventDefault(); // Ngăn ngừa submit form mặc định  var num1 = $('#num1').val();  var num2 = $('#num2').val();  // Gửi yêu cầu AJAX đến process.php  $.ajax({  type: 'POST',  url: 'process.php',  data: { num1: num1, num2: num2 },  success: function(response) {  $('#result-container').html(response); // Hiển thị kết quả vào phần tử HTML  },  error: function(xhr, status, error) {  console.error(error);  $('#result-container').html('<div class="alert alert-danger">Failed to execute calculation.</div>');  }  });  });  });  </script>  </body>  </html> |
| --- |

## 2.2. Xử lý yêu cầu trong PHP (process.php)

File process.php sẽ nhận các tham số từ form, gọi file thực thi để tính tổng và trả về kết quả.

1. Tạo một file mới và đặt tên là process.php.
2. Thêm mã PHP để xử lý yêu cầu:
   * Định nghĩa đường dẫn tới file EXE.
   * Truyền hai tham số số nguyên cần tính tổng.
   * Sử dụng hàm exec trong PHP để chạy file EXE với các tham số đã truyền.
   * Đọc kết quả từ file kết quả (ví dụ: result.txt) được tạo bởi chương trình EXE.
   * Hiển thị kết quả ra trình duyệt bằng cách sử dụng PHP.

| <?php  if ($\_SERVER["REQUEST\_METHOD"] == "POST") {  $num1 = isset($\_POST['num1']) ? intval($\_POST['num1']) : 0;  $num2 = isset($\_POST['num2']) ? intval($\_POST['num2']) : 0;  // OPTION 1: Dùng đường dẫn tương đối  $exe\_relative\_path = 'C++\Sum.exe'; // Đường dẫn tương đối tới file thực thi  $result\_relative\_path = 'result.txt';  // OPTION 2: Dùng bằng biến siêu toàn cục trong PHP chứa đường dẫn tới thư mục gốc của tài liệu trên máy chủ web.  $exe\_path = $\_SERVER['DOCUMENT\_ROOT'] . '/Job39/' . $exe\_relative\_path;    // Kiểm tra xem file .exe có tồn tại và có quyền chạy  if (file\_exists($exe\_path) && is\_executable($exe\_path)) {  // Chạy file .exe với 2 tham số  $command = escapeshellcmd("$exe\_path $num1 $num2");  exec($command, $output, $return\_var);  if ($return\_var === 0) {  // Đường dẫn đến file kết quả  $result\_file = $\_SERVER['DOCUMENT\_ROOT'] . '/Job39/' . $result\_relative\_path; // Đảm bảo file kết quả ở nơi bạn mong muốn  // Đọc nội dung file kết quả và echo ra ngoài trình duyệt  if (file\_exists($result\_file)) {  $result = file\_get\_contents($result\_file);  if ($result !== false) {  echo "<div class='alert alert-info'>The sum of " . htmlspecialchars($num1) . " and " . htmlspecialchars($num2) . " is: " . htmlspecialchars(trim($result)) . "</div>";  } else {  echo "<div class='alert alert-danger'>Failed to read result.txt.</div>";  }  } else {  echo "<div class='alert alert-danger'>result.txt not found.</div>";  }  } else {  echo "<div class='alert alert-danger'>Failed to execute sum.exe with return code $return\_var.</div>";  }  } else {  echo "<div class='alert alert-danger'>Executable file not found or permission issue.</div>";  }  }  ?> |
| --- |

Hàm escapeshellcmd() trong PHP là một hàm giúp bảo vệ an toàn khi thực hiện các lệnh shell bằng cách loại bỏ hoặc chuyển đổi các ký tự có thể gây rủi ro hoặc tác động không mong muốn khi thực hiện các lệnh hệ thống.

**Cách thức hoạt động của escapeshellcmd()**

1. **Chức năng:** escapeshellcmd() được sử dụng để chuẩn bị một chuỗi hoặc lệnh shell trước khi truyền vào hàm exec(), system(), hoặc passthru(). Hàm này giúp ngăn ngừa tấn công bằng cách tiêm lệnh (command injection) bằng cách loại bỏ các ký tự đặc biệt hoặc các tham số mà lệnh shell không thể xử lý một cách an toàn.
2. **Công dụng:**
   * Khi bạn truyền một chuỗi hoặc lệnh vào hàm exec(), có thể có rủi ro tiềm ẩn nếu chuỗi đó chứa các ký tự đặc biệt hoặc các ký tự có thể bị hiểu sai thành các tham số cho một lệnh khác.
   * escapeshellcmd() giúp loại bỏ hoặc thay thế các ký tự này bằng cách tạo ra một chuỗi an toàn để truyền vào lệnh hệ thống.
3. **Cách sử dụng:**

| $command = "path/to/your/executable.exe " . escapeshellcmd($num1) . " " . escapeshellcmd($num2);  exec($command, $output, $return\_var); |
| --- |

Trong ví dụ trên, escapeshellcmd() sẽ đảm bảo rằng giá trị của $num1 và $num2 đã được xử lý an toàn trước khi truyền vào hàm exec().

1. **Lưu ý:**

escapeshellcmd() không phải là giải pháp hoàn toàn bảo mật. Nếu dữ liệu đầu vào chưa được kiểm tra kỹ lưỡng, vẫn có thể có rủi ro. Vì vậy, đảm bảo kiểm tra và kiểm soát đầu vào từ người dùng là rất quan trọng.