

Kiến thức PHP





File PHP là gì?

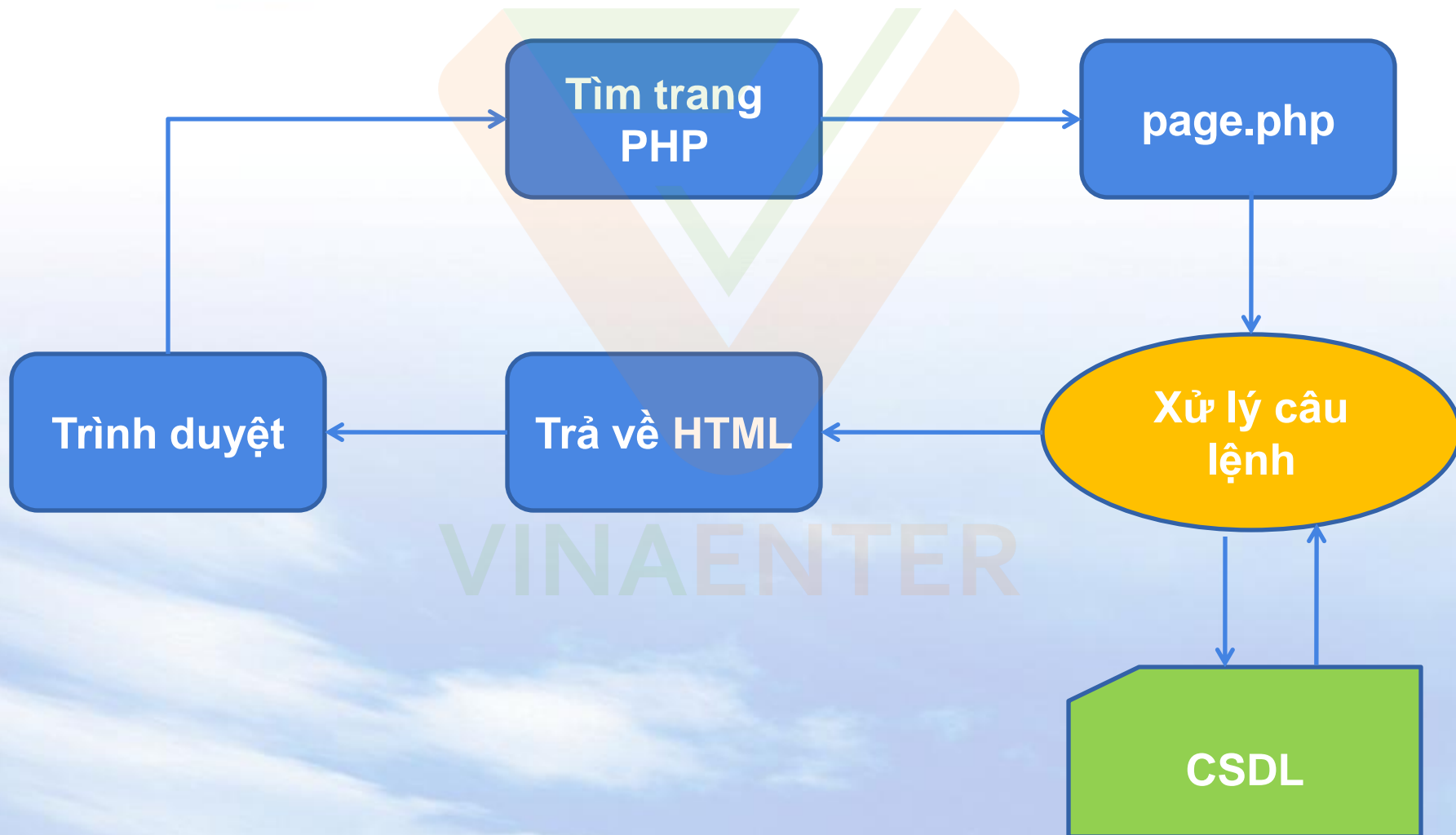
- File PHP có thể bao gồm text, HTML, CSS, JavaScript, và mã code PHP
- Mã code PHP được thực thi tại server, và trả về kết quả trên trình duyệt bằng mã nguồn HTML
- File PHP có đuôi mở rộng là ".php"



Tại sao bạn nên dùng PHP

- Mã nguồn mở và miễn phí
- Phổ biến
- Chuyên viết cho môi trường web
- Có thể lập trình với dạng OOP
- Nhiều hỗ trợ và tài liệu hướng dẫn

PHP hoạt động thế nào





Tạo trang PHP đầu tiên

Tạo trang PHP đầu tiên lưu trong thư mục:

htdocs / hocphp / page.php

với nội dung: "**Đây là trang PHP đầu tiên của tôi**"
và chạy trên localhost.

```
<?php
echo "Đây là trang PHP đầu tiên của tôi";
?>
```

Variables: Biến số

The logo for VinaENTER, featuring the text "VinaENTER" in orange and blue, with a white arrow pointing right, all within a blue rectangular frame.

- Trong PHP biến được ký hiệu bởi dấu "\$" đi sau là các ký tự bao gồm các chữ cái thường (a - z), in hoa (A - Z), các chữ số (0 - 9) và dấu "gạch dưới" (_) .
- Biến không được viết cách
- Biến không được bắt đầu bằng chữ số



Một số cách đặt tên biến đúng

- \$bien
- \$bienSo
- \$BienSo
- \$bien_so
- \$bien2





Sử dụng biến

- Biến

```
<?php
$bien = 25251325;
echo $bien;

$bien = "VinaTAB EDU";
echo $bien;

$Bien = "Học lập trình ONLINE";
echo $Bien;
?>
```




In biến trong dấu nháy đôi " ... "

- Kết hợp { ... } để in biến trong dấu nháy " ... "

```
<?php
$fullname = "VinaTAB";
$username = "admin";
$password = 123456;

echo "Tên trung tâm là {$fullname} ";
echo "Tên đăng nhập là {$username} và Mật khẩu là: {$password}";
?>
```



In biến trong dấu nháy đơn ' ... '

- Kết hợp nối chuỗi ■ để in biến trong dấu nháy đơn ' ... '

```
<?php
$fullname = "VinaTAB";
$username = "admin";
$password = 123456;

echo 'Tên trung tâm là ' . $fullname . ' ';
echo 'Tên đăng nhập là ' . $username . ' và Mật khẩu là ' . $password;
?>
```



In HTML trong PHP

Tên trung tâm là **VinaTAB**

Tên đăng nhập là **admin** và Mật khẩu là: **123456**



```
<?php
$fullname = "VinaTAB";
$username = "admin";
$password = 123456;

echo "Tên trung tâm là <strong>{$fullname}</strong> ";
echo "<br /><br />";
echo "Tên đăng nhập là <span style='color:red'>{$username}</span>
và Mật khẩu là: <span style='color:blue'>{$password}</span>";
?>
```



PHP kết hợp HTML

Tên trung tâm là **VinaTAB**

Website trung tâm là: <http://vinatab.net>

Tên đăng nhập là **admin** và Mật khẩu là: **123456**



```
<?php
$fullname = "VinaTAB";
$username = "admin";
$password = 123456;

echo "Tên trung tâm là <strong>{$fullname}</strong> ";
?>
<p>Website trung tâm là: <a href="http://vinatab.net" title="">http://vinatab.net</a></p>
<?php
echo "Tên đăng nhập là <span style='color:red'>{$username}</span>
và Mật khẩu là: <span style='color:blue'>{$password}</span>";
?>
```



Chuỗi - string

- Các biến dạng chuỗi(string) được dùng để lưu trữ văn bản hoặc ký tự.
- Các chuỗi hay ký tự cần được đặt trong dấu ngoặc kép "..." hoặc ngoặc đơn '...'

```
<?php
$hoten = "Nguyễn Văn A";
$lop = "KH01";

echo "<p>Họ tên: {$hoten}<br />Học viên lớp: {$lop}</p>";
?>
```



Các hàm thường dùng trong chuỗi

```
<?php
$string1 = "Cố gắng học";
$string2 = " sẽ thành tài";
?>

<p>
    <strong>String Length: </strong><?php echo strlen($string1); ?>
</p>
<p>
    <strong>String thay thế: </strong>
    <?php echo str_replace('sẽ', 'ắt', $string2); ?>
</p>
```

- **strlen(\$chuoi)**: lấy độ dài của chuỗi
- **str_replace(\$tim, \$thaythe, \$chuoi)**: tìm và thay thế



Hằng - constant

- Hằng thường là 1 giá trị không thay đổi và được sử dụng nhiều lần trong 1 file hoặc 1 dự án PHP
- Tên hằng thường viết HOA

```
<?php
define("USERNAME", "admin");
define("PASSWORD", 123456);

echo "Chào, " . USERNAME;
echo "<br /> Password của bạn là: " . PASSWORD;
?>
```



Các hàm thường dùng trong số

\$so = 12189000.478;

- **rand(\$min, \$max):** lấy số ngẫu nhiên từ \$min đến \$max
- **ceil:** Làm tròn đến số nguyên gần nhất và lớn nhất
- **floor:** Làm tròn đến số nguyên gần nhất và nhỏ nhất
- **number_format:** định dạng cách hiển thị giữa các phần nghìn trong 1 số

Các hàm thường dùng trong số

```
<?php
```

```
$so = 12189000.478;
```

```
echo rand(1,100) . "<br />";
```

```
echo ceil($so) . "<br />";
```

```
echo floor($so) . "<br />";
```

```
echo number_format($so, 2, ",", ".") . "<br/>";
```

```
?>
```

56

12189001

12189000

12.189.000,48



Tính toán trong số

```
<?php
$don_gia = 2900000;
$so_luong = 3;
$uu_dai = 0.1;
//tổng học phí 3 học viên
$tongtien = $don_gia * $so_luong;
//tính học phí 3 học viên được ưu đãi
$tienuudai = $tongtien * $uu_dai;
//tổng học phí 3 học viên phải đóng
$thanhtien = $tongtien - $tienuudai;
//in ra
echo "<br />Tổng học phí 3 học viên: {$tongtien}";
echo "<br />Tổng học phí ưu đãi: {$tienuudai}";
echo "<br />Học phí 3 học viên cần đóng: {$thanhtien}";
?>
```

Lệnh điều kiện if

- Dạng 1: khuyết else

```
if (điều kiện) {  
    //khởi lệnh tại đây  
}
```

- Ví dụ

```
$a = 5;  
if ($a > 2) {  
    echo "Số a lớn hơn 2";  
}
```



Lệnh điều kiện if ...else

- Dạng 2

```
if (điều kiện) {  
    //khởi lệnh nếu điều kiện đúng  
} else {  
    //khởi lệnh nếu điều kiện sai  
}
```

- Ví dụ

```
$a = 5;  
if ($a >= 2) {  
    echo "Số a lớn hơn hoặc bằng 2";  
} else {  
    echo "Số a bé hơn 2";  
}
```




Cấu trúc chọn lựa switch

■ Cú pháp

```
switch (biến điều kiện)
{
    case giá trị 1:
        //Khối lệnh1
    break;
    case giá trị 2:
        //Khối lệnh 2
    break;
    ...
    [default:
        //khối lệnh khi không thỏa tất cả các case trên
    ]
}
```



Ví dụ switch...case

■ Ví dụ

```
$thu = rand(2,8);  
switch ($thu)  
{  
    case 2:  
        echo "Thứ 2"; break;  
    case 3:  
        echo "Thứ 3"; break;  
    case 4:  
        echo "Thứ 4"; break;  
    case 5:  
        echo "Thứ 5"; break;  
    case 6:  
        echo "Thứ 6"; break;  
    case 7:  
        echo "Thứ 7"; break;  
    default:  
        echo "chủ nhật";  
}
```



Vòng lặp for

- **for** được sử dụng khi chúng ta biết trước số lần cần lặp, biến đếm chạy trong khoảng giới hạn của vòng lặp, và giá trị lặp

```
for ($biến_đếm=giá trị; điều kiện; giá trị lặp) {  
    //khởi lệnh  
}
```

```
<?php  
$tong = 0;  
for($i = 1; $i <= 10; $i++) {  
    $tong = $tong + $i;  
}  
echo $tong;  
?>
```



Ví dụ Phân trang cách viết 1

[Trang 1](#) [Trang 2](#) [Trang 3](#) [Trang 4](#) [Trang 5](#) [Trang 6](#)



```
<ul class="pagination">
<?php
for($i = 1; $i <= 6; $i++) {
    echo "<li><a href='index.php'>Trang {$i}</a></li>";
}
?>
</ul>

<style>
.pagination{list-style:none;}
.pagination li{float:left; margin-right:10px;}
</style>
```



Ví dụ Phân trang cách viết 2

[Trang 1](#) [Trang 2](#) [Trang 3](#) [Trang 4](#) [Trang 5](#) [Trang 6](#)



```
<ul class="pagination">
<?php
for($i = 1; $i <= 6; $i++) {
?>
    <li><a href="index.php">Trang <?php echo $i; ?></a></li>
<?php
}
?>
</ul>

<style>
.pagination{list-style:none;}
.pagination li{float:left; margin-right:10px;}
</style>
```

Vòng lặp while

VinaENTER

- Thực hiện lặp đi lặp lại một công việc nào đó khi thỏa điều kiện
- **while** được sử dụng khi không xác định được số lần lặp (số lần lặp phụ thuộc vào điều kiện tại thời điểm thực thi)
- Chú ý: **phải có điều kiện để dừng lệnh**

```
<?php  
while (điều kiện) {  
    //khởi lệnh  
}  
?>
```


Vòng lặp while

```
<?php
//Tính tổng các số từ 1 đến 10
$tong = 0; $n = 10; $i = 0;
while ($i <= $n) {
    $i = $i + 1;
    $tong = $tong + $i;
}
echo $tong;
?>
```

Các hàm kiểm tra giá trị của biến

- **isset()**: Kiểm tra tồn tại của biến
- Ví dụ:

```
$a = 5;  
if (isset($a)) {  
    echo "Biến a = {$a}";  
} else {  
    echo "Không tồn tại biến a";  
}
```

Các hàm kiểm tra giá trị của biến

- **unset()**: Hủy giá trị của biến
- Ví dụ:

```
$a = 5;  
unset($a);  
  
if (isset($a)) {  
    echo "Biến a = {$a}";  
} else {  
    echo "Không tồn tại biến a";  
}
```



In biến dùng var_dump()

- **var_dump()**: là một hàm có sẵn trong PHP, dùng để kiểm tra các dữ liệu trong một biến
- Ví dụ:

int(10) bool(true) bool(false)

```
<?php
$biena = 10;
$check1 = true;
$check2 = false;

var_dump($biena);
var_dump($check1);
var_dump($check2);
?>
```

Các toán tử

- Toán tử số học: $+$ $-$ $*$ $/$ $\%$
- Toán tử nối chuỗi: $.$
- Toán tử gán: $+=$ $-=$ $*=$ $/=$
- Toán tử so sánh: $==$ $!=$ $>$ $>=$ $<$ $<=$
- Toán tử luận lý: $!$ $\&\&$ $||$

- Toán tử số học: **+** **-** ***** **/** **%**

```
<?php
```

```
$x = 10; $y = 4;
```

```
echo ($x + $y); // kết quả 14
```

```
echo ($x - $y); // kết quả 6
```

```
echo ($x * $y); // kết quả 40
```

```
echo ($x / $y); // kết quả 2.5
```

```
echo ($x % $y); // kết quả 2
```

```
?>
```




Các toán tử

- Toán tử nối chuỗi: ■

```
<?php
```

```
$str1 = "Học lập trình";
```

```
$str2 = "PHP&MySQL";
```

```
echo $str1. " " . $str2;
```

```
?>
```



Các toán tử

- Toán tử gán: $+=$ $-=$ $*=$ $/=$

Toán tử gán	Ý nghĩa
$x = y$	$x = y$
$x += y$	$x = x + y$
$x -= y$	$x = x - y$
$x *= y$	$x = x * y$
$x /= y$	$x = x / y$
$x \% = y$	$x = x \% y$

Ví dụ Toán tử gán

```
<?php
```

```
$y = 150; $y += 50; echo $y;
```

```
$x = 125; $x -= 25; echo $x;
```

```
$y = 10; $y *= 6; echo $y;
```

```
$x = 10; $x /= 4; echo $x;
```

```
$y = 20; $y %= 6; echo $y;
```

```
?>
```



Các toán tử

- Toán tử so sánh: **==** **!=** **>** **>=** **<** **<=**
- Dùng để so sánh 2 giá trị (có kiểu dữ liệu là kiểu chuỗi string hoặc kiểu số)
 - ==** : so sánh hai giá trị. Trả về là TRUE nếu 2 giá trị bằng nhau, ngược lại, khác nhau thì trả về FALSE.
 - !=** : trả về TRUE nếu 2 giá trị khác nhau. Còn trả về FALSE nếu nó giống nhau.
 - >** : ví dụ \$a > \$b, trả về TRUE nếu a lớn hơn b.
 - >=** : ví dụ \$a >= \$b, trả về TRUE nếu a lớn hơn hoặc bằng b.
 - <** : ví dụ \$a < \$b, trả về TRUE nếu a nhỏ hơn b.
 - <=** : ví dụ \$a <= \$b, trả về TRUE nếu a nhỏ hơn hoặc bằng b.



Các toán tử

- Toán tử luận lý: **!** **&&** **||**
- Toán tử **&&** (hay **and**): chỉ đúng khi cả 2 đều đúng
- Toán tử **||** (hay **or**): chỉ sai khi cả 2 đều sai

\$a	\$b	Toán tử	Kết quả
true	true	&&	true
true	false	&&	false
false	true		true
false	false		false
true		!	false

Các toán tử - Kết quả sao?

```
<?php
//cho các biến
$vong1 = 86; $vong2 = 60; $vong3 = 90;
//biểu thức
$kiemtra1 = ($vong1 == $vong3) > true;
$kiemtra2 = ($vong1 >= $vong3) == false;
$kiemtra3 = ($vong2 <= $vong1) && ($vong2 < $vong3);
$kiemtra4 = ($vong1 != $vong3) || $vong2;
$kiemtra5 = (!($vong1 == 90)) || ($vong3 == 90);
$kiemtra6 = (($vong2 > 60) && ($vong2 < 80)) || (!($vong1 == $vong3));
$kiemtra7 = !$kiemtra6;
$kiemtra8 = $kiemtra2 && (!$kiemtra3);
?>
```


Ví dụ HEADER - location

```
<?php ob_start();?>
<html>
  <head>
    <title>Header</title>
  </head>
  <body>
    <?php
      header("location: http://vinatab.net");
      exit();
    ?>
  </body>
</html>
<?php ob_end_flush();?>
```



Dùng HEADER để chuyển hướng

```
<?php
```

```
ob_start();
```

```
// các lệnh PHP
```

```
header ("LOCATION: url_cần_chuyển");
```

```
// các lệnh PHP
```

```
ob_end_flush();
```

```
?>
```

Ví dụ HEADER - location

```
<?php ob_start();?>
<html>
  <head>
    <title>Header</title>
  </head>
  <body>
    <?php
      $role = rand(1,3);
      if ($role == 1){
        header("location: admin.php");
      } elseif ($role == 2){
        header("location: mod.php");
      } else {
        header("location: index.php");
      }
      exit();
    ?>
  </body>
</html>
<?php ob_end_flush();?>
```