

# Tham Khảo Nhanh PHP

Cú pháp, mảng, OOP, database, file I/O cơ bản

## Cơ Bản

### Hello World

```
<?php
echo "Hello, World!\n";
// PHP code must be inside <?php ... ?> tags
```

### Chạy PHP

```
php script.php          # run a file
php -r 'echo "hi\n";'   # run inline code
php -S localhost:8000   # built-in dev server
```

### Chú Thích

```
// single-line comment
# also single-line
/* multi-line
comment */
```

## Biến & Kiểu Dữ Liệu

### Biến

```
$name = "PHP";           // string
$version = 8.3;         // float
$count = 42;            // int
$active = true;        // bool
$items = null;         // null
```

### Kiểm Tra Kiểu

<b>gettype(\$x)</b>	Trả về kiểu dưới dạng chuỗi
<b>is_string(\$x)</b>	Kiểm tra nếu là chuỗi
<b>is_int(\$x)</b>	Kiểm tra nếu là số nguyên
<b>is_array(\$x)</b>	Kiểm tra nếu là mảng
<b>is_null(\$x)</b>	Kiểm tra nếu là null
<b>isset(\$x)</b>	Kiểm tra nếu đã đặt và không null
<b>empty(\$x)</b>	Kiểm tra nếu rỗng (falsy)

### Ép Kiểu

```
$n = (int) "42";         // 42
$s = (string) 3.14;     // "3.14"
$b = (bool) "";        // false
$a = (array) $obj;     // object to array
```

### Hằng Số

```
define("MAX_SIZE", 100);
const API_VERSION = "v2";
echo MAX_SIZE; // 100
```

## Strings

### String Cơ Bản

```
$name = "World";
echo "Hello, $name!"; // variable interpolation
echo 'Hello, $name!'; // literal (no interpolation)
echo "Value: {$arr['key']}"; // complex expression
```

## Hàm String

<b>strlen(\$s)</b>	Độ dài chuỗi tính bằng byte
<b>mb_strlen(\$s)</b>	Độ dài chuỗi tính theo ký tự (an toàn với multibyte)
<b>strtolower(\$s)</b>	Chuyển thành chữ thường
<b>strtoupper(\$s)</b>	Chuyển thành chữ hoa
<b>trim(\$s)</b>	Xóa khoảng trắng ở hai đầu
<b>str_replace(a, b, \$s)</b>	Thay <b>a</b> bằng <b>b</b> trong <b>\$s</b>
<b>substr(\$s, 0, 5)</b>	Chuỗi con từ vị trí 0, độ dài 5
<b>strpos(\$s, 'find')</b>	Tìm vị trí chuỗi con (false nếu không tìm thấy)
<b>explode(' ', \$s)</b>	Tách chuỗi thành mảng
<b>implode(' ', \$a)</b>	Nối mảng thành chuỗi

### Heredoc & Nowdoc

```
$html = <<<HTML
<p>Hello, $name</p>
HTML;
$raw = <<<'TEXT'
No $interpolation here
TEXT;
```

## Mảng

### Indexed & Associative

```
$nums = [1, 2, 3]; // indexed
$user = ["name" => "Alice", "age" => 30]; // associative
$nums[] = 4; // append
echo $user["name"]; // access
```

### Hàm Mảng

<b>count(\$a)</b>	Số phần tử
<b>array_push(\$a, \$v)</b>	Thêm vào cuối
<b>array_pop(\$a)</b>	Xóa và trả về phần tử cuối
<b>array_merge(\$a, \$b)</b>	Kết hợp hai mảng
<b>in_array(\$v, \$a)</b>	Kiểm tra giá trị tồn tại
<b>array_key_exists(\$k, \$a)</b>	Kiểm tra key tồn tại
<b>array_map(\$fn, \$a)</b>	Áp dụng hàm lên mỗi phần tử
<b>array_filter(\$a, \$fn)</b>	Lọc phần tử theo callback
<b>sort(\$a)</b>	Sắp xếp in-place (đánh lại chỉ số)
<b>array_keys(\$a)</b>	Trả về tất cả keys

### Lặp

```
foreach ($users as $user) { echo $user; }
foreach ($map as $key => $value) {
    echo "$key: $value\n";
}
```

## Hàm

### Hàm Cơ Bản

```
function add(int $a, int $b): int {
    return $a + $b;
}
echo add(3, 5);
```

### Tham Số Mặc Định & Có Tên

```
function greet(string $name, string $greeting = "Hello"): string {
    return "$greeting, $name!";
}
greet("Alice");
greet(greeting: "Hi", name: "Bob"); // named args (PHP 8+)
```

### Arrow Functions

```
$double = fn(int $x): int => $x * 2;
$nums = array_map(fn($n) => $n * 10, [1, 2, 3]);
```

## Closures

```
$factor = 3;
$multiply = function(int $x) use ($factor): int {
    return $x * $factor;
};
echo $multiply(5); // 15
```

## Classes & Objects

### Định Nghĩa Class

```
class User {
    public function __construct(
        private string $name,
        private int $age = 0,
    ) {}
    public function greet(): string { return "Hi, {$this->name}"; }
}
```

### Kế Thừa & Interfaces

```
interface Printable {
    public function toString(): string;
}
class Admin extends User implements Printable {
    public function toString(): string { return "Admin"; }
}
```

### Phạm Vi Truy Cập

<b>public</b>	Truy cập từ bất kỳ đâu
<b>protected</b>	Truy cập từ class và các lớp con
<b>private</b>	Chỉ truy cập trong class
<b>readonly</b>	Chỉ gán được một lần (PHP 8.1+)
<b>static</b>	Thuộc class, không phải instance
<b>abstract</b>	Phải được implement bởi lớp con

### Traits

```
trait Timestamped {
    public function createdAt(): string {
        return date('Y-m-d H:i:s');
    }
}
class Post { use Timestamped; }
```

## Xử Lý Lỗi

### Try / Catch / Finally

```
try {
    $result = riskyOperation();
} catch (InvalidArgumentException $e) {
    echo "Bad input: " . $e->getMessage();
} catch (Exception $e) {
    echo "Error: " . $e->getMessage();
} finally { cleanup(); }
```

### Custom Exceptions

```
class ApiException extends RuntimeException {
    public function __construct(string $message, private int $statusCode = 500) {
        parent::__construct($message, $statusCode);
    }
}
```

### Null Safety (PHP 8+)

```
$len = $user?->address?->zip; // nullsafe operator
$name = $input ?? "default"; // null coalescing
$data ??= []; // null coalescing assignment
```

# Tham Khảo Nhanh PHP

## File I/O

### Đọc & Ghi Files

```
$content = file_get_contents("data.txt");
file_put_contents("out.txt", $content);
$lines = file("data.txt", FILE_IGNORE_NEW_LINES);
```

### File Handle

```
$f = fopen("log.txt", "a");
fwrite($f, "entry\n");
fclose($f);
```

### Hàm File

<b>file_exists(\$path)</b>	Kiểm tra file tồn tại
<b>is_dir(\$path)</b>	Kiểm tra path là thư mục
<b>mkdir(\$path, 0755, true)</b>	Tạo thư mục đệ quy
<b>unlink(\$path)</b>	Xóa file
<b>glob('*.*txt')</b>	Tim file khớp pattern
<b>realpath(\$path)</b>	Giải quyết đường dẫn tuyệt đối đầy đủ

## Database

### Kết Nối PDO

```
$pdo = new PDO(
    "mysql:host=localhost;dbname=app",
    "user", "password",
    [PDO::ATTR_ERRMODE => PDO::ERRMODE_EXCEPTION]
);
```

### Prepared Statements

```
$stmt = $pdo->prepare("SELECT * FROM users WHERE id = :id");
$stmt->execute(["id" => 42]);
$user = $stmt->fetch(PDO::FETCH_ASSOC);
```

### Insert & Update

```
$stmt = $pdo->prepare("INSERT INTO users (name, email) VALUES
(?, ?)");
$stmt->execute(["Alice", "alice@example.com"]);
$id = $pdo->lastInsertId();
```

### Chế Độ Fetch PDO

<b>fetch()</b>	Lấy một hàng
<b>fetchAll()</b>	Lấy tất cả hàng
<b>FETCH_ASSOC</b>	Trả về dạng mảng kết hợp
<b>FETCH_OBJ</b>	Trả về dạng object ẩn danh
<b>FETCH_CLASS</b>	Trả về dạng instance của class cụ thể

## Hàm Phổ Biến

### JSON

```
$json = json_encode(["name" => "Alice", "age" => 30]);
$data = json_decode($json, true); // true = assoc array
$data = json_decode($json); // object
```

### Ngày & Giờ

```
echo date("Y-m-d H:i:s"); // 2026-03-26 12:00:00
$ts = strtotime("+1 week");
$dt = new DateTime("2026-01-01");
echo $dt->format("D, M j"); // Thu, Jan 1
```

## Toán & Ngẫu Nhiên

<b>abs(\$n)</b>	Giá trị tuyệt đối
<b>round(\$n, 2)</b>	Làm tròn đến 2 chữ số thập phân
<b>ceil(\$n) / floor(\$n)</b>	Làm tròn lên / xuống
<b>min(\$a, \$b) / max(\$a, \$b)</b>	Nhỏ nhất / lớn nhất
<b>random_int(1, 100)</b>	Số nguyên ngẫu nhiên bảo mật
<b>number_format(\$n, 2)</b>	Định dạng với dấu phân cách hàng nghìn

## Biểu Thức Chính Quy

```
preg_match('/^[a-z]+$/', $str, $matches);
preg_match_all('/\d+/', $str, $all);
$result = preg_replace('/\s+/', ' ', $str);
```