Engineer

PHP

以前からあるウェブの多くを動かす言語

どのようなものか

PHPは、ウェブサーバーで実行するスクリプト言語です。リクエストが来るとコードが動き、ブラウザーに届くのは生成したHTMLであり、コード自体ではありません。名前は現在、「PHP: Hypertext Preprocessor」という再帰的な略語ですが、元は「Personal Home Page」でした。ファイルは、通常のHTMLの `<?php` と `?>` の間にコードを埋めたものにも、フレームワークが全リクエストを通す純粋なコードにもできます。コマンドラインのスクリプトも動かせますが、設計はウェブページの生成を中心にしています。

始まりは1994年で、Rasmus Lerdorfが、自分のオンライン履歴書への訪問数を数えるCGIバイナリー一式をCで書き、Personal Home Page Toolsと呼びました。1995年6月にソースを公開し、PHP/FIへと育ちました。1997年にAndi GutmansとZeev Suraskiがパーサーを書き直し始め、1998年6月に、再帰的な名前のPHP 3.0として公開しました。最盛期には、インターネット上のウェブサーバーの約10%に入っていました。二人の名前から取った次のZend Engineが、2000年5月のPHP 4.0を支え、2004年7月にはPHP 5が続きました。

今日のサーバーでは、同梱するFastCGI Process Manager、PHP-FPMで動かすのが一般的です。手前のウェブサーバーが、各PHPリクエストをワーカープロセスのプールへ渡します。各プールは、独自のユーザーとグループ、環境、php.ini設定で動かせます。プロセス管理は、固定数を保つ、必要時に起動する、上下限の間で数を変える、という方式を取ります。リリースはブランチで進み、直近の四つは、一年ずつ間隔を空けて公開されました。各ブランチは、バグとセキュリティーの修正を行う二年の通常サポート、重大なセキュリティー修正だけの二年を経て、終了します。2022年12月8日公開のPHP 8.2は2026年12月31日まで、2025年11月20日公開のPHP 8.5は2029年12月31日まで、セキュリティー修正を受けます。

PHPは、演算が必要とすれば型を変換し、特に緩やかな比較演算子 `==` で、それが現れます。両方が数値文字列、または片方が数値でもう片方が数値文字列なら、数値として比べます。PHP 8.0.0より前は、数値と比較する文字列を、どれも先に数値へ変えたため、 `0 == "a"` は真でした。厳密な演算子 `===` は、値と型の両方を比べ、変換しません。マニュアル自身も、多くの場合はこちらを選ぶよう勧めます。

```
<?php
var_dump(0 == "a");      // PHP 7: true, PHP 8: false
var_dump("1" == "01");   // 両方でtrue：二つの数値文字列
var_dump("10" == "1e1"); // 両方でtrue：10と1e1は等しい
var_dump(100 == "1e2");  // 両方でtrue
var_dump(0 === "a");     // 両方でfalse：===は型を変換しない
```

概念を判断に生かす

「PHPとは？」を実際の判断に生かしましょう。その概念を、開発または購入するシステムに結び付けて考えるお手伝いをします。 データの保存先、処理のルール、他のシステムとの接続方法など、利用者の目に見えない部分をつくります。

サービスを見る お問い合わせ →

用語の背後にある前提を検証する

解説とリンク先の実績を参考に、プロジェクトについての質問を整理してください。ツールを選ぶ前に、前提を一緒に検討できます。 ETL/ELTパイプライン6本を設計・運用 — 6 3種類のDBエンジンにまたがるETLパイプライン

ETL/ELTパイプライン6本を設計・運用 お問い合わせ →

なぜ重要なのか

PHPが重要なのは、主に、既に書かれたコードのためです。多くの組織が自分で編集するサイトに使うCMS、WordPressとJoomlaはPHPアプリです。WordPressはPHP 8.3以上を推奨し、Joomla 6はPHP 8.3.0以上を必須にします。WordPressはPHP 7.4でも動きますが、要件ページは、その版はサポートを終えたと警告します。社内の誰も一行も書いていなくても、事業が何年も依存する場合があるため、下で動く版は忘れられます。ブランチが終了しても、サイトはページを返し続けます。修正が来なくなるだけで、画面には何も出ません。

更新で、動作が静かに変わることもあります。PHP 8.0は文字列と数値の比較を変え、PHP 7で真だった条件が、PHP 8では、エラーなしに偽になり得ます。PHP 8.4は、 `password_hash()` のbcryptの既定コストを10から12へ上げ、同じハードウェアでも、新しいパスワードハッシュの計算負担が増えました。どちらも文書に記載され、誰かが計画、検査、責任を担う更新で初めて、本番へ届きます。

PHPの版は、横に期限を記した依存関係として扱うよう勧めます。既存のアプリが動くなら、保ち続ける価値があります。サポート中のブランチに置き、セキュリティーサポート終了より十分前に次の更新を予定し、専用のFPMプールとOSユーザーを与えます。新規開発なら、顧客がWordPressやJoomlaで編集するサイトや、引き継ぐ運用チームがPHPをよく知る場合に選び、それ以外は、そのチームの技能で判断します。言語自体が危険なのはまれで、運用責任者のいない版が問題です。

実際の活用例

自社の技術スタックは、ウェブ層でのPHPの置き場を明確にします。ファイルにできるページは静的に生成し、それらのページへのリクエストを処理する際にアプリケーションを実行せずに済むようにしています。このサイトもHugoで構築しています。DjangoとPHPは、アプリケーションを提供する場所で使います。選ぶ基準も記します。問題に合うか、当社が去った後に顧客のチームが運用できるか、デモの見栄えより、どう壊れるかです。既存のスタックが正解なら、そう伝えるという約束は、PHPにもそのまま当てはまります。

2015年から2022年にかけて、創業者はGISソフトウェア開発のチームリーダーとして、三十を超えるGISプロジェクトの開発、デプロイ、稼働後のサポートまでを担当しました。PHPは、JavaScript、Python、SQLと並ぶ、アプリケーション層の四つの言語の一つでした。その下の空間データはすべてPostgreSQLとPostGISにあり、手前にはMapbox GLやLeafletの地図があり、一部はArcGISやHEREのAPIを呼びました。言語が何でも、出荷後も責任を持ち、問題を他へ渡しませんでした。壊れれば自分に電話が来ると知っていると、設計の仕方も変わります。

問題が起きるところ

- **秘密の検証に、緩やかな比較を使います。** `==` で調べるトークンやハッシュは、両方が数値らしく見えると、数値の規則で比べます。`"10"` と `"1e1"` のように異なる文字列も、等しいと見なします。パスワードには `password_verify()` 、ほかの秘密には `hash_equals()` を使います。実行時間から秘密の内容が漏れず、利用者の文字列を第二引数に取ります。それ以外は `===` を使います。
- **リクエストのデータでSQLを組み立てます。** PDOのプリペアドステートメントはパラメーターをクォートし、マニュアルは、これだけを使うアプリはSQLインジェクションから守られると説明します。ただし、エスケープしていない入力から作る部分は例外です。プレースホルダーが表すのは値全体なので、一覧のソート列のような、値でない箇所は、なお文字列に入ります。コード内の固定した一覧で検査します。

- **ハッシュの保存欄が狭過ぎます。** 現在の `PASSWORD_DEFAULT` はbcryptですが、より強いアルゴリズムの追加に合わせ、変える設計だとマニュアルは警告します。60バイトを超えて広げられる列を勧め、255を良い選択とします。bcryptはパスワードを72バイトで切り詰めます。ハッシュ全体を保存し、ログイン成功後に `password_needs_rehash()` を呼び、ログインに合わせて古いハッシュを更新します。
- **ブランチのサポート期限を過ぎます。** アプリが動き続けるため、更新を予定せず、最後には複数ブランチを一気に飛び越すことになります。使う版のセキュリティーサポート終了日を記録し、一ブランチずつ上げ、今の版が切れる前に、次の版でテスト一式を動かします。
- **FPMプールを共有し、制限を緩くします。** 複数のアプリを一つのユーザーで動かすと、一つの欠陥が全アプリのファイルへ届きます。`security.limit_extensions` の既定値は `.php .phar` で、設定を誤ったウェブサーバーが、ほかのファイルをPHPに渡せないよう、 `.php` だけにするよう勧めます。アプリごとに専用のプールとユーザーを与え、サーバーのメモリーで保持できる数に `pm.max_children` を設定します。

参考資料

- PHPマニュアル：PHPとは — プロジェクト自身の短い定義と、HTMLに埋め込む最初の例です。
- PHP：サポート対象のバージョン — サポート方針と、各現行ブランチの通常サポート期間、セキュリティー修正のみの期間です。
- PHPマニュアル：比較演算子 — 緩やかな比較と厳密な比較、PHP 7とPHP 8の出力の対比、型別の比較表です。
- PHPマニュアル：password_hash — アルゴリズム、コスト、保存列の幅、72バイト上限、8.4での既定コスト変更です。
- PHPマニュアル：プリペアドステートメントとストアドプロシージャー — PDOのパラメーター結合と、なおSQLインジェクションが可能な場所です。
- PHPマニュアル：FPM設定 — プールのディレクティブ、プロセス管理の各方式、 `clear_env` 、 `security.limit_extensions` です。

用語集

実際の制約に合わせて選ぶ

業務の流れ、既存のインターフェース、変更を必要とする利用者をお知らせください。検証可能な納品内容と、お客様のチームが保守を続けられる引き継ぎ内容を定められます。

料金を見る お問い合わせ →

用語を実務に結び付ける

「PHPとは？」がお客様の組織にどう当てはまるかをご相談ください。現在のシステムの具体例が一つあれば、話を始められます。

サービスを見る

welcome@engineer.company +45 50 10 61 84 デンマーク、コペンハーゲン

このページを持ち歩く

連絡先カード

心を込めて丁寧に制作： D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

著作権 © 2025 – 現在 · Engineer ApS · すべての権利を留保します。

印刷元： https://engineer.company/ja/glossary/php/