

PHP8 技術者認定試験上級/準上級試験 出題範囲解説と学習のコツ

PHP8 技術者認定試験上級/準上級試験の出題範囲について、少し詳しく解説をしていきます。

この文章が、受験をする人、受験を考えている人への一助になれば幸いです。

1) インストールと設定

出題割合としては 3%程度なのですが、基本的な所にもなりますので大枠は押さえておく
とよいでしょう。

マニュアルの

[インストールにあたっての一般的な注意事項](#)、

[FastCGI Process Manager \(FPM\)](#)、

[実行時設定\(\[php.ini\]\(#\) 関連\)](#)

あたりに通り目を通しておきましょう。

特に `php.ini` の書き方や、「`php.ini` 以外で設定を書ける場所(プログラム内に書ける `ini_set()` を含む)」など、少し突っ込んだ所まで理解をしておく
とよいでしょう。

2) 言語リファレンス

出題割合も 50%近くあり、大変にボリュームのある範囲です。

ここはもう少し詳細にかみ砕いて出題範囲を確認しておきましょう。

まずは基本の構文を確認しておきましょう。

[基本的な構文と式](#)は、ここ自体は試験範囲ではありませんが理解していることが前提になります
ので、しっかりと把握しておきましょう。

[型](#)、

[変数](#)、

[定数](#)、

[演算子](#)、

制御構造

は、いずれも出題範囲になります。一度は一通り目を通して、コードを書いて、理解しておくといよいでしょう。

関数、

クラスとオブジェクト

は重要です。特に「クラスとオブジェクト」については、マニュアルのページ数も多いのですがその分出题も多いので、ここは手を抜かずにしっかりと学習しておくといよいでしょう。

名前空間、

エラー、

例外(exceptions)、

ジェネレータ、

アトリビュート、

リファレンスの説明、

定義済みのアトリビュート

も出題範囲なので、一度は一通り目を通して、コードを書いて、理解しておくといよいでしょう。

なお一部「PHP 8.0 ではまだ追加されていない」ものもあり、それらは出題範囲の対象外です。

定義済みの変数と

定義済みの例外と

定義済みのインターフェイスとクラスは出題範囲になります。

また(PHP マニュアル内の区分では「関数リファレンス」内にありますが)、[Standard PHP Library \(SPL\)](#)も出題範囲になります。記憶する必要はありませんが、様々なクラスがあるので、一度は「目を通したりミニマムコードを書いて動かしてみたり」して、触れておくといよいでしょう。

サポートするプロトコル/ラッパーも出題範囲です。

ここも、一度はコードを書いて、動きや概念を理解しておくといよいでしょう。

また、PHP のマニュアル区分では「付録」になるのですが。

「PHP x.x.x から PHP x.x.x への移行」のページのうち、「PHP x.x で増えたもの」がいくつか、出題範囲になっています。

PHP 7.0 で増えたもの、

PHP 7.1 で増えたもの、
PHP 7.2 で増えたもの、
PHP 7.3 で増えたもの、
PHP 7.4 で増えたもの、
PHP 8.0 で増えたもの

が出題範囲になります。

「どのバージョンで増えたか」の暗記は不要ですが、「こんな事ができるようになった(機能の理解)」はしておきましょう。

「(どのバージョンで増えたか、の暗記は不要なので)最新のバージョンの機能を十全に理解していればよい」とも言えるのですが、(試験からは少し話がそれますが)実務ではどうしても「少し古いバージョンを扱わざるを得ないシーン」もあるかと思うので、マニュアルを一読はしておくとういかに思います。

注意)

列挙型(Enum)と

ファイバーは、いずれも「PHP 8.0 よりは後に追加された機能」であるため、試験範囲対象外なので、出題範囲の対象外です。

3) セキュリティ

出題割合は 13%程度ですが、重要な所なのでしっかりと学習しておきましょう。

マニュアルの「セキュリティ」のカテゴリ以外も含めて、以下が出題範囲となるのでしっかりと学習しておきましょう。

出題範囲の脆弱性と、それに関連する関数等のマニュアルの URL を記載します。

特にここでは「どのようなセキュリティホールがあり得るのか」「それに対して、プログラム側ではどのような対応ができたり、どのような注意が必要なのか」を併せて学んでおくとよいでしょう。

- XSS
- <https://www.php.net/manual/ja/function.htmlspecialchars.php>
- SQL-Injection
- <http://php.net/manual/ja/book.pdo.php>
- CSRF
- <https://www.php.net/manual/ja/function.uniqid.php>
- <https://www.php.net/manual/ja/ref.random.php>

- <https://www.php.net/manual/ja/function.openssl-random-pseudo-bytes.php>
- Cookie
- <https://www.php.net/manual/ja/function.setcookie.php>
- <https://www.php.net/manual/ja/features.cookies.php>
- セッション
- <https://www.php.net/manual/ja/book.session.php>
- ログイン（パスワード認証）
- <https://www.php.net/manual/ja/book.password.php>
- ディレクトリトラバーサル
- <https://www.php.net/manual/ja/function.basename.php>
- クリックジャック
- <https://www.php.net/manual/ja/function.header.php>
- コマンドインジェクション
- <https://www.php.net/manual/ja/function.escapeshellarg.php>
- <https://www.php.net/manual/ja/function.escapeshellcmd.php>
- ファイルアップロード
- <https://www.php.net/manual/ja/features.file-upload.php>
- 安全ではないデシリアライゼーション
- <https://www.php.net/manual/ja/function.serialize.php>
- <https://www.php.net/manual/ja/function.unserialize.php>

4) 機能

出題割合は3%程度になりますが、ここも一度は目を通しておくとよいでしょう。

マニュアルの

[PHP](#) をコマンドラインから使用する、

[ガベージコレクション](#)

あたりが範囲になります。。

5) 基本的な関数

出題割合が 30%ほどあり、またマニュアル上でもかなり多くのページがあります。
暗記の必要はありませんが、以下の点に注意しながら学習しておくといよいでしょう。

- 出題範囲は「PHP コアに含まれる関数」が対象になります
- 「関数名」や「引数の一覧」「戻り値」などを暗記する必要はありませんが、マニュアルにある例えば

```
debug_backtrace(int $options = DEBUG_BACKTRACE_PROVIDE_OBJECT, int $limit = 0): array
```

という記述をみて「関数名や、引数と戻り値の型や引数のデフォルト値」は理解できるようにしておきましょう

- 数は多いですが、一度くらいは目を通しておくといよいでしょう
- 何個か、でもよいので、実際にミニマムコードなど書いて挙動の確認をしておくといよいでしょう

出題の範囲は、以下になります。

エラー処理およびログ記録 <http://php.net/manual/ja/book.errorfunc.php>、
PHP オプションと情報 <http://php.net/manual/ja/book.info.php>、
Phar <http://php.net/manual/ja/book.phar.php>、
HASH メッセージダイジェストフレームワーク <http://php.net/manual/ja/book.hash.php>、
OpenSSL <http://php.net/manual/ja/book.openssl.php>、
リフレクション <http://php.net/manual/ja/book.reflection.php>、
日付・時刻クラス <https://www.php.net/manual/ja/book.datetime.php>、
日付・時刻関数 <https://www.php.net/manual/ja/ref.datetime.php>、
ディレクトリ <http://php.net/manual/ja/book.dir.php>、
SPL のディレクトリ <https://www.php.net/manual/ja/class.directoryiterator.php>、
SPL のディレクトリイテレータ <https://www.php.net/manual/ja/class.recursivedirectoryiterator.php>、
ファイル情報 <http://php.net/manual/ja/book.fileinfo.php>、
SPL のファイル情報 <https://www.php.net/manual/ja/spl.files.php>、
メール <http://php.net/manual/ja/book.mail.php>、
Random 関数 <https://www.php.net/manual/ja/ref.random.php>、
出力バッファリング制御 <http://php.net/manual/ja/book.outcontrol.php>、
数学関数 <http://php.net/manual/ja/book.math.php>、
数学関数 <http://php.net/manual/ja/book.math.php>、
システムプログラムの実行 <http://php.net/manual/ja/book.exec.php>、
POSIX <http://php.net/manual/ja/book.posix.php>、

Tokenizer <http://php.net/manual/ja/book.tokenizer.php>、
JavaScript Object Notation <http://php.net/manual/ja/book.json.php>、
その他の関数 <http://php.net/manual/ja/book.misc.php>、
ストリーム <http://php.net/manual/ja/book.stream.php>、
URLs <http://php.net/manual/ja/book.url.php>、
ネットワーク <http://php.net/manual/ja/book.network.php>、
正規表現 (Perl 互換) <http://php.net/manual/ja/book.pcre.php>、
文字列 <http://php.net/manual/ja/book.strings.php>、
配列 <http://php.net/manual/ja/book.array.php>、
クラス/オブジェクトの情報 <http://php.net/manual/ja/book.classobj.php>、
変数操作 <http://php.net/manual/ja/book.var.php>、
文字タイプチェック <http://php.net/manual/ja/book ctype.php>、
関数処理 <http://php.net/manual/ja/book.funchand.php>、
Document Object Model <http://php.net/manual/ja/book.dom.php>、
libxml <http://php.net/manual/ja/book.libxml.php>、
SimpleXML <http://php.net/manual/ja/book.simplexml.php>、
データのフィルタリング <http://php.net/manual/ja/book.filter.php>、

その他

他の出題範囲に混ざっているのですが、PHP マニュアルの [付録](#) の、以下の各ページも一度は読んでおくとよいでしょう。

[PHP のデバッグ](#)、
[Configure オプション](#)、
[php.ini ディレクティブ](#)、
[拡張モジュールの一覧/分類](#)、
[関数エイリアスのリスト](#)、
[予約語の一覧](#)、
[利用できるフィルタのリスト](#)、
[ユーザーレベルでの命名の手引き](#)

最後に、古庄親方から一言。

現在、機構の Web Page 内で PHP8 上級試験のコラムを書いています。

<https://www.phpexam.jp/archives/category/column/oyakata>

日本語の解説部分は比較的少なく、主にコードで説明をしているので。まずは各々の知識を「理解しているか? の確認」ができるかと思います。書かれているコードが「理解している範疇のもの」なのか「知らない知識を含むもの」なのかを確認していただけます。

「理解しているコードの書き方」はそのまま読み流し、「不安がある箇所」について、マニュアルなどで再勉強するとよいでしょう。

また、コラムの日本語部分の説明をもう少し追記して電子書籍にする事も検討しておりますので、どうぞご期待ください。

PHP 技術者認定機構 CTO 古庄道明