

PHP Development Associate

Prüfungscode IFW-PHP-200-01 · Fassung 1.0 · Stand 18.09.2026

Serverseitige Webentwicklung mit PHP 8: Sprachkern, Objektorientierung, Fehlerbehandlung sowie sichere Formular- und Datenbankzugriffe.

Die Prüfung im Überblick

Bearbeitungszeit
1 Std. 30 Min.

Fragen
70

Bestehensgrenze
70 %

Fragetypen
Einfach- und Mehrfachauswahl

Prüfungssprache
Deutsch

Zertifikat gültig
Unbegrenzt

Durchführung
Online, zeitlich überwacht

Preis
49,00 €

Themenblöcke

Block	Thema	Gewichtung	Fragen
1	Sprachkern, Typen und Funktionen	27 %	19
2	Objektorientierte Programmierung	27 %	19
3	Fehlerbehandlung und Typsicherheit	13 %	9
4	Formulare, Zustand und HTTP	13 %	9
5	Sicherheit und Datenbankzugriff	20 %	14
Gesamt		100 %	70

Die Gewichtung gibt an, welchen Anteil ein Block an den Prüfungsfragen hat. Die Fragen werden beim Start zufällig und anteilig je Block aus dem Fragenpool gezogen.

Sprachkern, Typen und Funktionen

Werte, Operatoren und Funktionen von PHP 8 – besonders dort, wo sich PHP anders verhält als erwartet.

- 1.1** Skalare Typen und Typumwandlung beurteilen
Zahl und String im Vergleich, lose und strikte Gleichheit, Ausgabe mit `echo`
- 1.2** Operatoren und ihre Priorität anwenden
Addition vor Verkettung seit PHP 8, `??`, `?->`, `match` mit striktem Vergleich
- 1.3** Arrays und Array-Funktionen einsetzen
`array_filter()`, `array_keys()`, `implode()`, Erhalt der Schlüssel
- 1.4** Schleifen und Referenzen richtig verwenden
`foreach` mit `&$wert` und dessen Nachwirkung
- 1.5** Funktionen, Standardwerte und Closures schreiben
Named Arguments, Rückgabetypen, Arrow Functions und ihre Bindung an den Kontext

Objektorientierte Programmierung

Klassen und Objekte in PHP 8: Sichtbarkeiten, Vererbung, Interfaces, Traits und moderne Sprachmittel.

- 2.1** Klassen, Eigenschaften und Konstruktoren definieren
Constructor Promotion, `readonly`, `__toString()`
- 2.2** Sichtbarkeiten unterscheiden
`public`, `protected`, `private` und der Zugriff aus Kindklassen
- 2.3** Vererbung, abstrakte Klassen und Interfaces einsetzen
`extends`, `implements`, abstrakte Methoden, mehrere Interfaces
- 2.4** Traits und Late Static Binding nutzen
`use` im Klassenrumpf, `static::` gegenüber `self::`
- 2.5** Statische Eigenschaften und Enums verwenden
geteilter Zustand je Klasse, `from()` und `tryFrom()`

Fehlerbehandlung und Typsicherheit

Exceptions, Fehlerklassen und strikte Typprüfung – und was dabei in welcher Reihenfolge ausgeführt wird.

- 3.1** Exceptions auslösen und behandeln
`try`, `catch`, `finally` und deren Reihenfolge bei einer Rückgabe
- 3.2** Fehlerhierarchie von PHP 8 kennen
`Throwable`, `Exception`, `Error`, `TypeError`, `UnhandledMatchError`

- 3.3** Strikte Typen einsetzen
`declare(strict_types=1)` und seine Wirkung je Datei
- 3.4** Dateien einbinden
`require` gegenüber `include` bei fehlender Datei

BLOCK 4

13 % · 9 Fragen

Formulare, Zustand und HTTP

Eingaben verarbeiten und Zustand über mehrere Anfragen hinweg halten.

- 4.1** Superglobals verwenden
`$_GET`, `$_POST`, `$_SERVER`, `$_COOKIE`
- 4.2** Anfragen und Methoden auswerten
`$_SERVER['REQUEST_METHOD']` statt Prüfung auf gefüllte Felder
- 4.3** Sessions einsetzen
serverseitige Speicherung, Session-ID, Erneuern nach der Anmeldung
- 4.4** Cookies setzen und lesen
`setcookie()` wirkt erst bei der nächsten Anfrage

BLOCK 5

20 % · 14 Fragen

Sicherheit und Datenbankzugriff

Die üblichen Angriffe auf eine Webanwendung und der sichere Weg zur Datenbank.

- 5.1** SQL-Injection verhindern
Prepared Statements, Platzhalter, Bezeichner über eine Whitelist
- 5.2** Vorbereitete Abfragen mit PDO ausführen
`prepare()`, `execute()` mit benannten Platzhaltern
- 5.3** Ausgaben gegen XSS absichern
`htmlspecialchars()` und der Kontext der Ausgabe
- 5.4** CSRF-Schutz einbauen
Token je Sitzung, SameSite-Cookies
- 5.5** Passwörter sicher speichern
`password_hash()`, `password_verify()`, Kostenfaktor im Hash

Institut für Webentwicklung · unabhängiges Lernangebot ohne Verbindung zu den genannten Technologieanbietern. Maßgeblich ist die Fassung auf der Zertifizierungsseite.