PHP Developer II (kod: K-PHP-2)

Opis i cel kursu

Kurs przeznaczony jest dla tych, którzy:

— są już co najmniej średnio zaawansowanymi programistami webowymi,
— chcą podwyższyć swoje kwalifikacje / znaleźć lepszą pracę,
— są ciekawi aktualnych trendów i bieżących, zaawansowanych technologii.

Wszystkie zajęcia prowadzone są przy komputerach i mają charakter warsztatowy, zgodnie z naszym mottem: *teorię też da się pokazać na praktycznym przykładzie.*

Zapytaj o szczegóły
tel. 22 63 64 164
akademia@alx.pl

Program

Kurs dzieli się na dwa moduły, które są także oferowane jako osobne szkolenia Symfony 4 i Doctrine 2 - aplikacje webowe i Nowoczesny JavaScript i TypeScript dla programistów.

Najbliższe terminy
2026-12-15 (Zdalnie)
2026-12-15 (Warszawa)

I. Symfony 4 i Doctrine 2 - aplikacje webowe

1. **PHP w nowoczesnych aplikacjach webowych - wprowadzenie**
   — Obiektowość i MVC
   — Architektura aplikacji webowych
   — Podstawowe praktyki i koncepcje organizacji kodu
2. Symfony 4 - z lotu ptaka
   — Architektura standardowej aplikacji
   — Fundamentalne i akcydentalne elementy (Request, Response, HttpKernel, Controller, Service Container)
   — Modularność, najczęściej używane bundle
   — Instalacja i konfiguracja typowej aplikacji Symfony 4, użycie narzędzia composer oraz konsoli Symfony
   — Implementacja standardów PSR i dobre praktyki związane z formatowaniem kodu
3. Podstawowe składniki aplikacji
   — Silnik szablonów Twig
      — szablony, dziedziczenie szablonów
      — tagi i filtry; tworzenie własnych rozszerzeń Twiga
      — Organizacja szablonów
   — Routing i kontrolery
      — wywoływanie akcji: URL i parametry
      — usługi udostępnione przez standardowy kontroler
      — * obsługa sesji, zwykłej i typu flash;
      — * forwarding i przekierowanie
      — * wyświetlanie szablonów
      — * wyświetlanie odpowiedzi w innych formatach (JSON, XML)
   — obsługa błędów
   — Uruchamianie aplikacji na serwerze wbudowanym i lokalnym
   — Dobre praktyki związane z przechowywaniem plików statycznych (js, css, obrazki)
   — Omówienie różnic trybów działania aplikacji (dev, prod)
   — System zdarzeń
      — Dostępne zdarzenia
      — Listenery i Subscribery
      — Zdarzenia związane z obsługą encji
4. Formularze i walidacja
   — Walidacja danych

- formularz i model formularza; FormBuilder i schemat typowego użycia formularzy
- wbudowane i własne typy pól (e.g. pola tekstowe, guziki, kalendarz, pola wyboru...)
- integracja walidacji i formularzy: obsługa formularza, zgadywanie typu pola;
- integracja Twiga i formularzy: helpery formularza i własne tematy, wyświetlanie formularza automatyczne lub ręczne,
- złożony model: formularze wbudowane, kolekcje formularzy wbudowanych, transformacje modelu;
- dynamicznie modyfikowalny formularz
- zabezpieczenie przeciwko CSRF

5. Uwierzytelnienie i autoryzacja
   - konfiguracja uwierzytelnienia - użytkownicy, grupy, role
   - dostęp do modułów aplikacji: zabezpieczanie ścieżek, metod, kontrolerów
   - dostęp do konkretnych danych
   - podstawowe strategie ACL (Access Control Lists) i Voters
6. Doctrine 2
   - różne modele ORM i ich konsekwencje: Active Record kontra Transparent Persistence;
   - model pracy, wzorce projektowe (unit of work, identity map, repository / dao)
   - konfiguracja i integracja z własnym projektem
   - sposoby generacji encji
   - ręczne tworzenie i poprawianie mappingów:
     - encje, klucze główne (proste i złożone), generatory kluczy głównych
     - pola typu prostego i asocjacje: asoscjacje jedno- i dwukierunkowe, strona nadrzędna i podrzędna
   - podstawy pracy z encjami: dodawanie, usuwanie, modyfikacja danych w bazie
   - kwerendy: Doctrine Query Language, Criteria, repozytoria
   - Wydajność i optymalizacja - leniwe oraz gorliwe wczytywanie relacji, doczytywanie danych kwerendami i relacjami, strategie stronicowania, kaskadowanie operacji
   - Sposoby przygotowania danych testowych (DoctrineFixtures i Faker)
7. Zagadnienia zaawansowane
   - rejestracja i wykorzystywanie serwisów, Service Container; alternatywne techniki: YAML, XML, PHP
   - Expression Language
   - serwisy i ich zależności - dependency injection (constructor, setter, property)
   - serwisy w funkcji kontrolerów

II. Nowoczesny JavaScript i TypeScript

1. Środowisko pracy współczesnego JavaScriptu
   - JavaScript w przeglądarce i poza nią: Node.js jako środowisko uruchomieniowe i narzędziowe
   - Menedżer pakietów npm i package.json: zależności, skrypty, wersje pakietów
   - Projekt startowy w Vite: serwer deweloperski i budowanie wersji produkcyjnej
   - Edytor, ESLint i automatyczne formatowanie kodu
2. JavaScript z perspektywy programisty innego języka
   - Zmienne let i const, typy wartości, porównania == i ===, wartości „prawdziwe” i „fałszywe”

Zapytaj o szczegóły
tel. 22 63 64 164
akademia@alx.pl

Najbliższe terminy
2026-12-15 (Zdalnie)
2026-12-15 (Warszawa)

- Obiekty i tablice, destrukturyzacja, operator spread, metody map, filter i reduce
- Funkcje, funkcje strzałkowe, domknięcia i zachowanie this
- Klasy, a pod spodem prototypy — rzut okiem
- Typowe pułapki i jak ich unikać

3. Moduły i organizacja kodu
   - Moduły ES: import i export
   - Korzystanie z bibliotek z npm
   - CommonJS w starszym kodzie — rzut okiem
4. Asynchroniczność
   - Pętla zdarzeń: dlaczego JavaScript nie czeka
   - Promise, async/await i obsługa błędów
   - Pobieranie danych z REST API: fetch, JSON, kody odpowiedzi
   - Kilka operacji naraz: Promise.all i pokrewne
5. TypeScript — po co i jak działa
   - Sprawdzanie typów a uruchamianie kodu: kompilator tsc, narzędzia usuwające typy, uruchamianie TypeScriptu w Node.js
   - Konfiguracja: tsconfig.json i tryb strict
   - Typowanie stopniowe: od JavaScriptu do TypeScriptu
6. System typów TypeScriptu
   - Adnotacje i inferencja typów
   - Typy literalne, unie i zawężanie typów
   - Typy any, unknown, null i undefined
   - Interfejsy i aliasy typów, typowanie funkcji
   - Typowanie strukturalne — czym różni się od Javy i C#
   - Klasy w TypeScripcie: modyfikatory dostępu, implements, klasy abstrakcyjne
7. Generyki i typy zaawansowane w praktyce
   - Funkcje i typy generyczne, ograniczenia typów
   - Typy pomocnicze: Partial, Pick, Omit, Record
   - Operatory keyof i typeof
   - Unie rozłączne do opisu stanów aplikacji
8. TypeScript w projekcie
   - Typowanie danych z API i sprawdzanie ich w czasie działania programu
   - Biblioteki z npm i pliki deklaracji typów (.d.ts)
   - TypeScript w przeglądarce: elementy DOM, zdarzenia, formularze
   - Przeniesienie fragmentu kodu z JavaScriptu do TypeScriptu krok po kroku
9. Praca z AI w projekcie TypeScript
   - Generowanie i refaktoryzacja kodu z asystentem AI w edytorze
   - Typy i kompilator jako pierwsza kontrola kodu wygenerowanego przez AI
   - Gdzie AI się myli: typowe błędy i sposoby ich wyłapania
10. Co dalej: TypeScript w React, Angular i Node.js — rzut okiem

Uwaga: ponieważ w świecie nowoczesnych technologii czas biegnie szybko, zastrzegamy sobie prawo do aktualizacji programu, a w ramach niej - zamianę opisanych wyżej technologii na bliskie odpowiedniki.

Przeznaczenie i wymagania

Szkolenie przeznaczone jest dla wszystkich chętnych, co najmniej średniozaawansowanych programistów WWW. Konieczne: dobra znajomość SQL, dobra znajomość PHP i co najmniej podstawowa programowania obiektowego; podstawowa znajomość administracji dowolnym wybranym systemem bazodanowym.

Zapytaj o szczegóły
tel. 22 63 64 164
akademia@alx.pl

Najbliższe terminy
2026-12-15 (Zdalnie)
2026-12-15 (Warszawa)

Pod względem programu, szkolenie stanowi kontynuację naszego kursu Deweloper Aplikacji WWW.

Certyfikaty

Uczestnicy szkolenia otrzymują imienne certyfikaty sygnowane przez ALX.

Informacje dodatkowe

W trosce o poziom zajęć i możliwość pełnego ich spożytkowania przez wszystkich kursantów, w wypadku wątpliwości co do spełniania wymagań kursu zachęcamy do odbycia rozmowy z wykładowcą.

Zapytaj o szczegóły

tel. 22 63 64 164
akademia@alx.pl

Najbliższe terminy

2026-12-15 (Zdalnie)
2026-12-15 (Warszawa)