Python Developer (kod: K-PYTHON)

Opis i cel kursu

Python jest jednym z najbardziej uniwersalnych języków programowania. Może być stosowany do obliczeń, tworzenia gier komputerowych, automatyzacji codziennych czynności i jako silnik aplikacji internetowych. Ta wszechstronność oraz łatwość nauki powodują, że jest on często wybierany nawet przez osoby nie będące programistami do uproszczenia ich codziennej pracy. Dla kursu dostępne są dwie opcje kontynuacji nauki: w kierunku tworzenia aplikacji i systemów serwerowych (backend) lub w kierunku analizy danych, uczenia maszynowego i sztucznej inteligencji (AI/analiza danych). Nasz kurs przeznaczony jest dla osób nigdy nie mających styczności z programowaniem, a Pythona z założenia może nauczyć się każdy!

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

Najbliższe terminy
2026-11-26 (Online (English))
2026-12-17 (Zdalnie)
2026-12-17 (Warszawa)
2026-12-19 (Zdalnie)
2026-12-19 (Warszawa)
2027-03-06 (Online (English))

Program

1. **Wprowadzenie i pierwsze kroki**
   - Czym są „programy” i „programowanie”?
   - Python na tle innych języków
   - Zastosowania i możliwości Pythona: historia i współczesność
   - Instalacja i konfiguracja środowiska pracy
     - Interpreter języka Python
     - Zintegrowane środowiska programistyczne (IDE): PyCharm lub VS Code
   - Pierwsze programy i przegląd dostępnych sposobów pracy z Pythonem
   - Styl kodowania i konwencje (PEP8)
2. **Podstawowe elementy języka Python**
   - Interakcja z użytkownikiem w trybie tekstowym: `print` i `input`
   - Zmienne i podstawowe typy danych
     - Wartości liczbowe, napisowe, logiczne
     - Operatory, porównania, spójniki
   - Instrukcja warunkowa `if`
   - Dopasowanie wartości `match`
   - Pętle `while` i `for`
3. **Struktury danych i ich zastosowania**
   - Listy i krotki
   - Wybieranie elementów i zakresów (*slicing*)
   - Zbiory i słowniki
   - Wyrażenia typu *comprehension*
   - Sortowanie danych
4. **Programowanie proceduralne**
   - Podstawy definiowania funkcji
   - Tryby przekazywania argumentów
   - Importowanie definicji z innych plików
   - Organizacja kodu w projekcie: moduły i pakiety
   - Dokumentacja i adnotacje
5. **Programowanie obiektowe**
   - Podstawy definiowania klas
     - Czym są: obiekt, klasa, atrybut, metoda
   - Zasady inicjalizacji obiektów i klas
   - Metody specjalne (*magic methods*)
   - Metody statyczne i klasowe
   - Dziedziczenie
   - Wprowadzenie do dobrych praktyk OOP: abstrakcja, hermetyzacja (*encapsulation*), spójność (*cohesion*)

6. **Obsługa wyjątków**
   - Zgłaszanie i przechwytywanie wyjątków
   - Definiowanie własnych wyjątków
7. **Biblioteka standardowa języka Python**
   - Wyrażenia regularne (moduł `re`)
   - Obsługa daty i czasu (moduły `time` i `datetime`)
   - Dodatkowe typy kolekcji (moduł `collections`)
   - Operacje na plikach
     - Otwieranie i zamykanie, konstrukcja `with`
     - Odczyt i zapis zawartości pliku
     - Przeglądanie katalogów i dostęp do informacji o systemie (moduł `os`)
   - Graficzny interfejs użytkownika (GUI) (podstawy modułu `tkinter`)
8. **Dodatkowe narzędzia i biblioteki**
   - Wirtualne środowiska (venv) i instalator pakietów `pip`
   - Testy jednostkowe na przykładzie `unittest` lub `pytest`
   - Debugowanie kodu w IDE
   - Pobieranie danych z sieci i dostęp do zdalnych usług
     - Wysyłanie zapytań HTTP (moduł `requests`)
     - Usługi typu REST (tzw. “Web API”) i format JSON z perspektywy Pythona
9. **Sztuczna inteligencja w pracy programisty**
   - AI podczas tworzenia oprogramowania, czyli podstawy *vibe coding*
     - Asystent AI w IDE – podpowiedzi, autouzupełnianie, generowanie kodu, refaktoryzacja
     - Przegląd dostępnych usług i wtyczek
     - Rozwiązywanie całych zadań za pomocą modeli językowych (Claude / ChatGPT / Gemini)
     - Automatyzacja powtarzalnych czynności
   - AI w kontekście testowania i kontroli jakości
   - Zdalny dostęp do usług generatywnej sztucznej inteligencji na przykładzie wybranej usługi (ChatGPT, Gemini lub analogicznej)
   - Jak dbać o poprawność i bezpieczeństwo?
10. **Wstęp do dalszych zastosowań języka Python**
    - Aplikacje webowe (framework Django)
    - Środowiska typu Jupyter / Colab i interaktywna praca z Pythonem
    - Obliczenia, analiza i wizualizacja danych (biblioteki `pandas`, `numpy` i `matplotlib`)

Przeznaczenie i wymagania

Od uczestników wymagana jest znajomość ogólnej obsługi komputera (kopiowanie plików, korzystanie z edytora tekstu) i umiejętność sprawnego korzystania z klawiatury.

Programowanie, szczególnie na początku, jest trudne, a jego nauka jest porównywalna z nauką obcego języka lub gry na instrumencie muzycznym. Dlatego warunkiem skutecznego uczestnictwa w kursie jest zarezerwowanie odpowiedniej ilości czasu w domu na ćwiczenia.

Certyfikaty

Uczestnicy szkolenia otrzymują imienne certyfikaty sygnowane przez ALX.

Zapytaj o szczegóły

tel. 22 63 64 164
akademia@alx.pl

Najbliższe terminy

2026-11-26 (Online (English))
2026-12-17 (Zdalnie)
2026-12-17 (Warszawa)
2026-12-19 (Zdalnie)
2026-12-19 (Warszawa)
2027-03-06 (Online (English))