



Program kursu

AI Python Engineer

kurs weekendowy

104 h zajęć z trenerem na żywo

40 h pracy własnej

70% / 30% praktyki / teorii

Postaw na
praktykę

Kurs AI Python Engineer

Buduj aplikacje z LLM-ami — od backendu po wdrożenie.

Kurs AI Python Engineer jest dla osób, które chcą nauczyć się tworzyć nowoczesne aplikacje backendowe w Pythonie – od podstaw, ale w standardzie zbliżonym do pracy w zespole. Program prowadzi przez budowę API, pracę z bazą danych i wdrożenie, a na końcu pokazuje, jak sensownie dodawać funkcje AI (LLM/RAG) do aplikacji.

Dla kogo jest ten kurs?

- ✓ chcesz nauczyć się programować i zrozumieć, jak powstaje backend
- ✓ chcesz tworzyć API i aplikacje webowe (nie tylko skrypty do szuflady)
- ✓ interesuje Cię automatyzacja i praca z danymi (pliki, API, baza)
- ✓ myślisz o przebranżowieniu do IT albo chcesz wejść w rolę backend developera
- ✓ chcesz korzystać z AI jak inżynier: szybko, ale bez halucynacji, z weryfikacją i kontrolą kosztów

Co będziesz potrafić po kursie?

- ✓ Budujesz aplikacje webowe w Pythonie i Django z REST API
- ✓ Integrujesz modele LLM w aplikacjach (function calling, guardrails, MCP)
- ✓ Projektujesz pipeline'y RAG z ewaluacją jakości i wdrożeniem
- ✓ Obsługujesz streaming, zadania w tle i bezpieczne uwierzytelnianie

Liczba godzin i forma zajęć

Zajęcia odbywają się w soboty i niedziele, w godzinach 08:00 – 12:00 lub 11:30 – 14:30.

15 h

Prework

szacowany czas na przygotowanie do kursu

104 h

Zajęcia z trenerem

zdalnie na żywo, zakończone certyfikatem

25 h

Praca własna

szacowany czas na pracę indywidualną między zajęciami

Co zbudujesz na kursie — projekt końcowy

Zbudujesz spójny projekt typu AI-enabled backend — "Support Desk / Knowledge Base API", aplikację webową (backend + panel), która:

- ✓ posiada użytkowników, role i uprawnienia
- ✓ przechowuje dane w PostgreSQL + migracje Django
- ✓ wystawia REST API (Django REST Framework) + dokumentację OpenAPI
- ✓ obsługuje zadania w tle i streaming odpowiedzi
- ✓ pozwala uploadować pliki / teksty do bazy wiedzy
- ✓ ma moduł AI: chat z function calling, RAG na własnych dokumentach użytkownika (vector store), ewaluację jakości odpowiedzi + podstawowy monitoring kosztów
- ✓ jest opakowana w Docker + ma CI + jest wdrożona na środowisko testowe



Nagrania z zajęć

Nagrania wideo z zajęć dostępne przez 6 miesięcy po zakończeniu kursu.



Narzędzia i technologie

W praktyce opanujesz: Python, SQL, PostgreSQL, GitHub, Git, Django, Docker, Pytest, ChatGPT, Copilot, Perplexity, Cursor, Gemini. Zakres opiera się na realnych case studies.



Certyfikat

Kurs kończysz zaliczeniem potwierdzającym umiejętności oraz certyfikatem ukończenia.



AI w kursie

Uczysz się korzystać z AI jak nowoczesny specjalista — do przyspieszania pracy, debugowania i analizy — zawsze z weryfikacją wyniku przez człowieka.

AI Python Engineer

14 modułów · 104 h



MODUŁ 0 · PREWORK

Pework (przygotowanie przed kursem)

Krótki etap przygotowawczy, dzięki któremu wchodzisz na pierwsze zajęcia bez stresu technicznego. Ustawiasz narzędzia, sprawdzasz czy wszystko działa i robisz małą rozgrzewkę z podstaw.

ZAKRES MODUŁU

- ✓ Przygotowanie narzędzi do zajęć: Zoom, Slack i konto GitHub
- ✓ Instalacja Python 3.12, VS Code (z rozszerzeniami) oraz Git i sprawdzenie działania
- ✓ Podstawowe czynności w terminalu: uruchomienie skryptu i sprawdzenie wersji narzędzi
- ✓ Pierwsze kroki z repozytorium: utworzenie repo na GitHub i umieszczenie pliku README
- ✓ Przećwiczenie podstaw Pythona potrzebnych na start



MODUŁ 1

Start i środowisko (Setup + fundamenty Pythona)

Ustawiasz środowisko pracy i wchodzisz w podstawy Pythona. Od pierwszego dnia pracujesz na repozytorium i poznajesz zasady bezpiecznego użycia AI w programowaniu.

ZAKRES MODUŁU

- ✓ Konfiguracja Pythona 3.12, VS Code oraz venv/uv i uruchamianie projektu lokalnie
- ✓ Podstawy terminala oraz najczęstsze problemy z uruchomieniem kodu
- ✓ Czytanie tracebacków i użycie debuggera na prostych przykładach
- ✓ Fundamenty Pythona: typy, instrukcje warunków, pętle i proste funkcje
- ✓ Zasady AI-pair: bezpieczeństwo, redakcja danych, weryfikacja odpowiedzi

2

MODUŁ 2

Python w praktyce

Uczysz się Pythona na praktycznych danych: kolekcje, pliki i obsługa błędów. Budujesz pierwsze małe narzędzie CLI, które później wykorzystasz jako część aplikacji.

ZAKRES MODUŁU

- ✓ Praca na list/dict/set i przekształcanie danych w praktycznych zadaniach
- ✓ Stosowanie comprehensions i prostych agregacji do analizy danych wejściowych
- ✓ Wczytywanie i zapisywanie plików z użyciem pathlib oraz obsługa encoding
- ✓ Projektowanie obsługi wyjątków i czytelnych komunikatów błędów dla użytkownika
- ✓ Budowa mini-CLI: wejście z pliku, przetworzenie, wynik na stdout

3

MODUŁ 3

Organizacja kodu i debugowanie

Porządkujesz kod tak, by dało się go rozwijać: moduły, pakiety i jasne granice odpowiedzialności. Uczysz się typowania, które zwiększa czytelność i ogranicza błędy, oraz praktycznego debugowania.

ZAKRES MODUŁU

- ✓ Podział kodu na moduły i pakiety oraz projektowanie struktury projektu
- ✓ Dobre praktyki importów i organizacji plików w repozytorium kodu
- ✓ Debugowanie problemów z użyciem breakpoints i krokowania
- ✓ Dodawanie type hints do funkcji i danych (list/dict/Optional) oraz odczyt błędów typów
- ✓ Refaktoring kodu z użyciem typów (uproszczenie i mniejsza podatność na błędy)

4

MODUŁ 4

Testy i jakość kodu

Wprowadzasz standard jakości: testy, automatyzacje i pipeline CI. Dzięki temu kolejne elementy projektu budujesz szybciej i bez rozjeżdżania jakości.

ZAKRES MODUŁU

- ✓ Pisanie testów w pytest: happy path, testy błędów i testy regresji
- ✓ Zastosowanie fixtures i parametryzacja w zadaniach z życia
- ✓ Uruchamianie coverage i interpretacja wyników
- ✓ Konfiguracja Ruff/Black oraz pre-commit jako standard repo
- ✓ Budowa CI w GitHub Actions: lint + test + coverage na pull requestach

5

MODUŁ 5

Framework Django: fundamenty web i API

Wchodzisz w świat aplikacji web: startujesz projekt we frameworku Django, uczysz się struktury, routingu i podstaw request/response, a potem budujesz pierwsze API w Django REST Framework z walidacją i testami.

ZAKRES MODUŁU

- ✓ Tworzenie i konfiguracja projektu w Django (settings, apps, manage.py)
- ✓ Routing URL i przepływ request → view → response
- ✓ Budowa pierwszych endpointów API w DRF (serializers + viewsets/routers)
- ✓ Projekt walidacji i spójnego formatu błędów w API
- ✓ Podstawowe testy endpointów (pytest-django / APIClient)

6

MODUŁ 6

SQL i dane w Django

Uczysz się pracy z danymi: rozumiesz SQL, projektujesz model danych i mapujesz go na Django ORM. Dodasz migracje i pierwsze realne dane w projekcie.

ZAKRES MODUŁU

- ✓ Podstawowe zapytania SQL (SELECT/JOIN/GROUP BY) na potrzeby aplikacji
- ✓ Projektowanie modelu danych i relacji między encjami projektu
- ✓ Definicja modeli w Django (pola, relacje, constraints) i użycie QuerySets
- ✓ Tworzenie i stosowanie migracji w Django oraz utrzymywanie spójnego schematu
- ✓ Testy integracyjne DB (pytest-django) i przygotowanie danych testowych

7

MODUŁ 7

Django REST Framework: CRUD i jakość API

Budujesz stabilne API na Django REST Framework: CRUD, walidacja, błędy i testy. Następnie dokładasz transakcje i podstawy wydajności (N+1 i dobre query).

ZAKRES MODUŁU

- ✓ Budowa CRUD w DRF (serializers + viewsets/routers) w spójnej strukturze projektu
- ✓ Projektowanie walidacji w serializerach i rozdział walidacji wejścia od domeny
- ✓ Utrzymywanie spójnego formatu błędów i statusów HTTP w API
- ✓ Stosowanie transaction.atomic i spójność operacji w DB
- ✓ Rozpoznanie i ograniczenie N+1 (select_related/prefetch_related) oraz testy integracyjne

8

MODUŁ 8

Auth i bezpieczeństwo w Django

Wprowadzasz użytkowników, logowanie i uprawnienia. Uczysz się praktycznych zasad bezpieczeństwa: sekrety, PII, limity i testy uprawnień, tak by aplikacja nie była otwartym API.

ZAKRES MODUŁU

- ✓ Zarządzanie konfiguracją i sekretami (.env) oraz rozumienie settings per env
- ✓ Stosowanie PII hygiene i bezpieczne podejście do logowania oraz błędów
- ✓ Implementacja uwierzytelniania w Django/DRF (np. Session lub JWT jako wariant kursowy)
- ✓ Permissions (role/uprawnienia) i izolacja danych per workspace/użytkownik
- ✓ Testy uprawnień i scenariusze nadużyć (abuse cases) + proste limity

9

MODUŁ 9

REST API i streaming

Dopinasz standardy API jak w pracy: zasoby, kontrakty, wersjonowanie, paginację i filtrowanie oraz testy pod kontrakt. Następnie uruchamiasz streaming odpowiedzi i prosty UI, przygotowując projekt pod czat AI.

ZAKRES MODUŁU

- ✓ Projektowanie REST API wokół zasobów oraz dobór metody i statusy HTTP
- ✓ Wdrożenie paginacji, filtrowanie i sortowanie w DRF
- ✓ Wersjonowanie API (/api/v1/...) i utrzymywanie spójnego formatu błędów
- ✓ Dokumentacja OpenAPI (light) jako kontrakt
- ✓ Streaming odpowiedzi (SSE) i podłączanie prostego UI do odbioru strumienia

10

MODUŁ 10

Zadania w tle

Uczysz się uruchamiać długie procesy poza requestem. Wprowadzasz kolejkę zadań, statusy jobów i odporność na ponowienia.

ZAKRES MODUŁU

- ✓ Konfiguracja kolejki i workera (Redis + job runner) ✓ Idempotency i odporność na ponowione żądania
- ✓ Projektowanie flow: start job → status → wynik ✓ Integracja jobów z realnym procesem w aplikacji (np. ingest plików)
- ✓ Wdrożenie retry/backoff i zastosowanie retry w praktyce

11

MODUŁ 11

LLM w aplikacji

Zaczynasz budować AI-feature'y w aplikacji. Wymuszasz structured output, walidujesz odpowiedzi i uczysz się kontrolować koszty.

ZAKRES MODUŁU

- ✓ Tokeny, kontekst i koszty w perspektywie backendu ✓ Fallbacki, timeouts i odporność na błędy modelu
- ✓ Projektowanie structured JSON output, który można walidować i testować ✓ Testy bad output: brak pól, złe typy, niepoprawny JSON
- ✓ Walidacja odpowiedzi modelem Pydantic i obsługa niezgodności

12

MODUŁ 12

Function calling, guardrails i MCP

Łączysz LLM z logiką domenową przez function calling, a potem robisz krok production-ish: poznajesz MCP jako standardowe podejście do narzędzi i kontekstu dla agentów. Uczysz się też realnych zagrożeń (prompt injection / tool poisoning) i podstaw obrony.

ZAKRES MODUŁU

- ✓ Implementacja function calling do wywoływania akcji domenowych w Django/DRF
- ✓ Whitelist narzędzi + walidacja argumentów + kontrola uprawnień
- ✓ Guardrails: timeouts, limity, bezpieczne błędy, testy nadużyć
- ✓ MCP (client/server) oraz pojęcia tools/resources/prompts i transporty
- ✓ Budowa mini serwera MCP, który wystawia narzędzia do Twojego projektu (bezpiecznie i przewidywalnie)

13

MODUŁ 13

RAG, ewaluacja i wdrożenie

Finalizujesz projekt: RAG end-to-end z cytowaniami, mini-ewaluacja jakości i wdrożenie. Dodajesz obserwowalność i kontrolę kosztów, czyli production-ish w praktyce.

ZAKRES MODUŁU

- ✓ Budowa RAG: ingest → chunking → embeddings → retrieval + citations
- ✓ Izolacja danych per workspace i bezpieczny upload dokumentów
- ✓ Mini-ewaluacja jakości: golden set + prompt tests + cache
- ✓ Observability light: logi, error tracking, podstawowe metryki
- ✓ Pakowanie i wdrożenie aplikacji: Docker/Compose + CI/CD + runbook + demo

Sfinansuj kurs tak, jak Ci wygodnie

W infoShare Academy oferujemy różne formy sfinansowania kursu — bez względu na to, który pakiet wybierzesz. Sprawdź, która forma jest dla Ciebie najlepsza!

01 Raty PayU 0%

Najszybszy sposób na rozłożenie płatności na niskie, comiesięczne raty. Do wyboru 5, 10 lub 15 rat 0% albo 24, 36, a nawet 50 rat nisko oprocentowanych.

02 Płatność jednorazowa

Płacisz i zaczynasz naukę. Jednorazowa płatność często wiąże się z dużym rabatem — zapytaj o niego nasze konsultantki.

03 Płatność ratalna

Możliwość rozłożenia płatności na wygodne raty wewnątrz Akademii. Skontaktuj się z nami, aby poznać szczegółowy harmonogram płatności.

04 Dofinansowanie z Urzędu Pracy

Warunkiem jest status osoby bezrobotnej oraz gwarancja zatrudnienia od przyszłego pracodawcy. Wniosek składasz osobiście w swoim Urzędzie Pracy.

05 Baza Usług Rozwojowych (BUR)

Nawet do 80% dofinansowania na kursy w infoShare Academy — dla pracowników mikro, małych i średnich firm, osób prowadzących działalność oraz sektora publicznego i NGO.

06 Pożyczki na kształcenie

Nieoprocentowana pożyczka spłacana do 36 miesięcy, finansuje do 100% kosztów (maks. 75 000 zł). Możliwe umorzenie od 25% do 50% kwoty.

07 Krajowy Fundusz Szkoleniowy (KFS)

Inicjatywa ze środków Funduszu Pracy wspierająca pracodawców inwestujących w kształcenie ustawiczne pracowników oraz własne.

Stawiamy na jakość

2015

uczymy IT
od tego roku

9970+

przeszkolonych
osób

330+

trenerów
w naszej kadrze

Łączymy kompetencje techniczne z praktyką biznesową, a każdy kurs prowadzą trenerzy pracujący na co dzień w branży. Dzięki temu uczysz się tego, co realnie przydaje się w pracy.



MASZ PYTANIA?

Sylwia Liedtke

Doradczyni ds. kursów

+48 730 822 825 · sylwia.liedtke@infohareacademy.com