

Python Web Developer

w CODE:ME

program kursu

1. Start, środowisko i pierwszy działający program

- Czym jest Python i gdzie jest używany (web, data, automation).
- Instalacja Python 3.12+, venv, pip.
- Konfiguracja VS Code, struktura projektu.
- Uruchamianie plików .py, hello world. print(), input(), komentarze w kodzie.
- Podstawy czytania błędów (SyntaxError).

2. Typy danych, operatory, formatowanie

- Typy danych: int, float, str, bool. Konwersje typów.
- Operatory matematyczne i logiczne.
- Formatowanie danych (f-stringi).
- Debugowanie kodu w edytorze.

3. Warunki i logika programu

- Instrukcje if / elif / else.
- Operatory logiczne.
- Typ list i typ dict – wprowadzenie.
- Walidacja danych użytkownika.
- Dobre praktyki czytelnej logiki programu.

4. Pętle i struktury danych

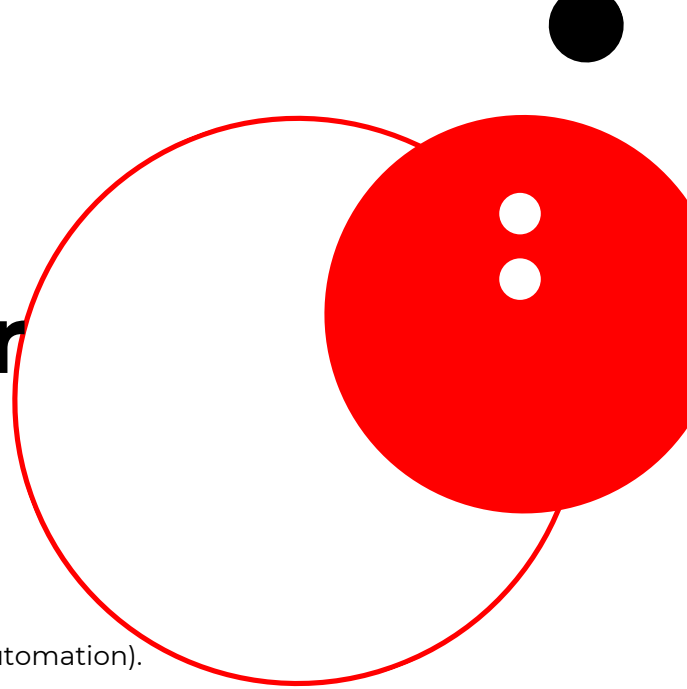
- Pętle for, while.
- Typy list, dict (wspomnienie o set).
- Iterowanie po danych.
- Podstawy pracy na kolekcjach.

5. Funkcje i modułarny kod

- Definiowanie funkcji.
- Parametry i wartości zwracane.
- Importy i podział kodu na pliki.
- Type hints.

6. Pliki, JSON, CSV

- Zapis i odczyt plików. Format JSON i CSV.
- Trwałość danych w aplikacji.



Python Web Developer

w CODE:ME

program kursu

7. Programowanie obiektowe w praktyce

- Klasy i obiekty.
- `__init__`, metody, `@property`. `dataclass`.
- Kompozycja zamiast dziedziczenia i dlaczego?

8. Kontynuacja OOP + Git

- Repozytorium Git
- Commity, branche. `.gitignore`.
- Podstawy code review.

9. HTTP i podstawy web

- Request / response.
- Metody HTTP i statusy odpowiedzi.
- JSON, REST.
- Podstawowe użycie metody GET z biblioteki `requests`.

10. Praca z API w Pythonie

- Biblioteka `requests`, obsługa błędów `try/except`.
- Przetwarzanie i walidacja danych z API.
- Wprowadzenie: backend vs frontend,
- czym jest Django i dlaczego go używamy.

11. Django: projekt, aplikacje, routing

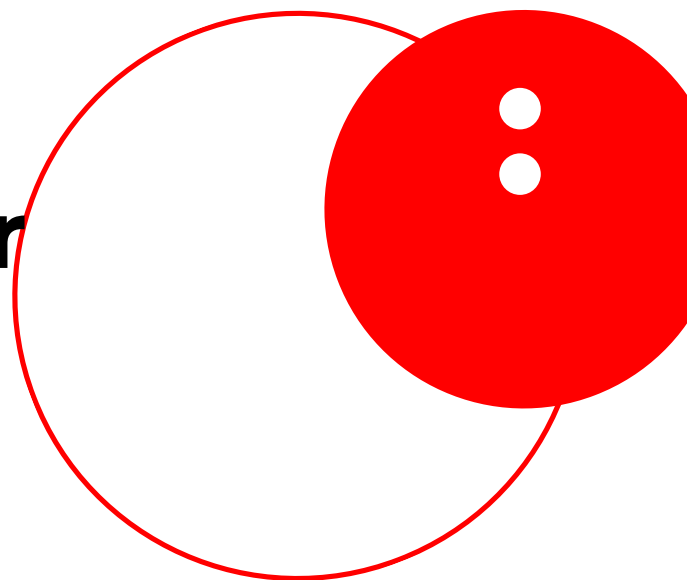
- Struktura projektu Django. `urls`, `views`. `Templates`.
- Uruchamianie serwera developerskiego.

12. Modele i ORM

- Modele Django.
- Migracje.
- Podstawy ORM.
- Relacje między modelami.

13. Django Admin

- Konfiguracja panelu admina.
- Listy, filtry, wyszukiwarka.
- Seed danych.



Python Web Developer

w CODE:ME

program kursu

14. CRUD i widoki

- Tworzenie, edycja i usuwanie danych.
- Routing.
- Komunikaty systemowe.

15. Formularze i walidacja

- Django Forms. ModelForms.
- Walidacja danych.
- Obsługa błędów w formularzach.

16. Autoryzacja użytkowników

- Logowanie i rejestracja. login_required. Dane per użytkownik.

17. Relacje i filtrowanie

- Relacje modeli.
- Filtrowanie danych. Wyszukiwanie. Sortowanie.

18. Testy i jakość kodu

- Debug w VSC.
- Testy widoków i formularzy.
- Test client.
- Podstawy refaktoryzacji.

19. Django REST API

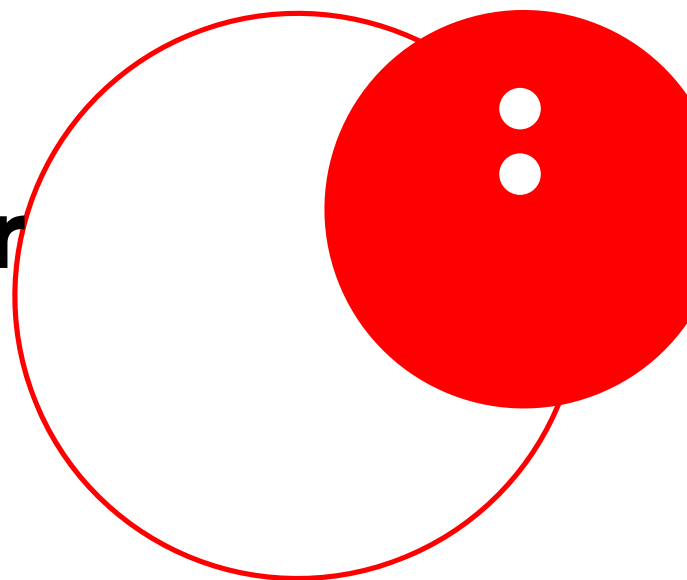
- Serializery. Endpointy API.
- Statusy HTTP.

20. API auth i bezpieczeństwo

- Bearer token / JWT.
- Permissions.
- Autoryzacja vs autentykacja.
- Ochrona danych.

21. Frontend – minimum

- Templates i interakcje bez SPA (HTMX / fetch).
- JavaScript + komunikacja z API (full-stack flow).



Python Web Developer

w CODE:ME

program kursu

22. Praca z bazą danych

- Podstawowa praca z danymi aplikacji, zapytania ORM, optymalizacja podstawowa, porządkowanie modeli i relacji.

23. Konfiguracja produkcyjna

- Zmienne środowiskowe. Static/media. Logging.
- Przygotowanie aplikacji pod deploy.

24. Demo, portfolio i prezentacja projektów końcowych

- Jak prezentować własny projekt.
- Opis projektu do CV i LinkedIn.
- README techniczne.
- Feedback i podsumowanie kursu.

Bądźmy w kontakcie!

kontakt@codeme.pl

tel: +48 724 379 836

codeme.pl