

Mów mi Python! – materiały i metody

– czyli **programowanie w języku Python** w ramach projektu "Koduj z klasą" organizowanego przez Centrum Edukacji Obywatelskiej. Szczegóły pod adresem: <http://www.ceo.org.pl/pl/koduj>.

Spis materiałów szkoleniowych

1. Podstawy Pythona

- *Toto Lotek*

Rozbudowany przykład wprowadzający podstawowe elementy języka, jak i programowania: zmienna, pobieranie i wyprowadzanie tekstu, proste typy danych, instrukcja warunkowa *if*, wyrażenie logiczne, pętla *for*, pętla *while*, *break*, *continue*, złożone typy danych, lista, zbiór, tupla, algorytm, poprawność algorytmu, obsługa wyjątków, funkcja, moduł.

- *Python kreśli* (Matplotlib)

Materiał prezentujący tworzenie wykresów oraz operacje matematyczne w Pythonie. Zagadnienia: listy, notacja wycinkowa, wyrażenia listowe, wizualizacja danych.

- *Python w przykładach* – zestaw przykładów prezentujących praktyczne wykorzystanie wprowadzonych zagadnień

2. Gra robotów (*Robot Game*, rlgkit)

Przykład gry planszowej, w której zadaniem gracza-programisty jest tworzenie strategii walki robotów. Na podstawie przykładowych zasad działania robota oraz odpowiadającego im kodu, gracz "buduje" i testuje swojego robota. Zagadnienia: klasa, metoda, biblioteka, wyrażenia listowe, zbiory, listy, tuple, instrukcje warunkowe.

3. Gry w Pythonie (Pygame)

Przykłady multimedialne prezentujące tworzenie i manipulowanie prostymi obiektami graficznymi (Pong, Kółko i krzyżyk) oraz graficzną wizualizację struktur danych (Życie Conwaya).

- *Pong* (wersja strukturalna i obiektowa)
- *Kółko i krzyżyk* (wersja strukturalna i obiektowa)
- *Życie Conwaya* (wersja strukturalna i obiektowa)

4. Bazy danych w Pythonie (SQLite)

Przykłady wykorzystania bazy danych: model bazy, tabela, pole, rekord, klucz podstawowy, klucz obcy, relacje, połączenie z bazą, operacje CRUD (Create, Read, Update, Delete), podstawy języka SQL, kwerenda, system ORM, klasa, obiekt, właściwości.

- Moduł SQL
- Systemy ORM (Peewee i SQLAlchemy)

5. Aplikacje internetowe

Przykłady zastosowania frameworków Flask i Django do tworzenia aplikacji działających w architekturze klient – serwer przy wykorzystaniu protokołu HTTP. Zagadnienia: żądania GET, POST, formularze, renderowanie widoków, szablony, tagi, treści dynamiczne i statyczne, arkusze stylów CSS

- *Quiz* (Flask)
- *ToDo* (Flask, SQLite)
- *Quiz ORM* (Flask)
- *Czat* (Django)

Materiały online

- **Wersja HTML:** <http://python101.readthedocs.org> lub <http://python101.rtfld.org>
- **Wersje źródłowe:** <https://github.com/koduj-z-klasa/python101>
- **Forum Koduj z Klasą:** <http://discourse.kodujzklasa.pl>

Metody realizacji

Cechy języka Python przedstawiane są na przykładach, których realizacja może przyjąć różne formy w zależności od dostępnego czasu. Zasada ogólna jest prosta: im więcej mamy czasu, tym więcej metod aktywizujących (kodowanie, testowanie, ćwiczenia, konsola Pythona, konsola Django itp.); im mniej, tym więcej metod podających (pokaz, wyjaśnienia najważniejszych fragmentów kodu, kopiuj-wklej). W niektórych materiałach (np. Robot Game, gry w Pygame) po skopiowaniu i wklejeniu kodu warto stosować zasadę uruchom-zmodyfikuj-uruchom.

1. Prezentacja, czyli uruchamianie gotowych przykładów wraz z omówieniem najważniejszych fragmentów kodu.
2. Wspólne budowanie programów od podstaw: kodowanie w edytorze, wklejanie bardziej skomplikowanych fragmentów kodu.
3. Ćwiczenia w interpreterze Pythona – niezbędne m. in. podczas wyjaśnianiu elementów języka oraz konstrukcji wykorzystywanych w przykładach.
4. Ćwiczenia i zadania wykonywane samodzielnie przez uczestników.