

# Wprowadzenie i organizacja pracy Korzystanie z dokumentacji języka Python Publikacja na GitHub

lab 1

## 1 Cel zadania

1. Poznanie zasad organizacji pracy w kursie.
2. Rozwinięcie umiejętności efektywnego samodzielnego korzystania z dokumentacji języka Python.
3. Konfiguracja środowiska programistycznego oraz repozytorium na GitHub.
4. Przećwiczenie umiejętności programistycznych w Pythonie.

## 2 Zakres zadania

1. Organizacja środowiska pracy.
2. Wyszukiwanie informacji w dokumentacji Pythona.
3. Stworzenie wieloelementowego programu Python.
4. Publikacja wyników na GitHub i złożenie zadania na Moodle.

## 3 Instrukcje – krok po kroku

### 3.1 Utworzenie repozytorium GitHub

1. Jeśli nie masz jeszcze konta na GitHub – utwórz je.

2. Zaloguj się i utwórz nowe repozytorium:
  - Nadaj mu nazwę, np. `python-intro`.
  - Wybierz opcję Public.
  - Dodaj plik `README.md`.
3. Sklonuj repozytorium na komputer (np. `git clone <URL>`).

### 3.2 Organizacja pracy w lokalnym środowisku

1. Utwórz w repozytorium katalog (np. `zadanie_1`) na bieżące zadanie.
2. Skonfiguruj wybrane środowisko (PyCharm, VS Code, Jupyter Notebook itp.).

### 3.3 Korzystanie z dokumentacji języka Python

Wyszukaj w dokumentacji Pythona co najmniej trzy zagadnienia:

- Opis funkcji wbudowanej (np. `zip()`, `enumerate()`, `sorted()`).
- Opis modułu z biblioteki standardowej (np. `math`, `random`, `time`).
- Opis wyjątku (np. `ValueError`, `ZeroDivisionError`).

Podaj krótkie wyjaśnienie każdego zagadnienia oraz link do dokumentacji.

### 3.4 Napisanie programu w Pythonie

1. W folderze `zadanie_1` stwórz plik `zadanie_1.py` lub `zadanie_1.ipynb`.
2. Program powinien wykonywać:
  - Tworzenie dwóch list i łączenie ich funkcją `zip()`.
  - Wykorzystanie jednej funkcji z wybranego modułu Pythona.
  - Obsługę wyjątku `try-except`.
  - Komentarze w kodzie z opisem użytych funkcji i linkami do dokumentacji.

### **3.5 Publikacja kodu na GitHub**

1. Wykonaj commit zmian i dodaj opis (np. „Dodano zadanie 1”).
2. Wypchnij kod do repozytorium GitHub (`git push origin main`).
3. Sprawdź, czy kod jest widoczny w przeglądarce GitHub.

## **4 Kryteria oceny**

Ocena uzależniona jest od kompletności przesłanego zadania. Czas na wykonanie 7 dni.