

# Narzędzia w branży IT

## Lab 6, 7 i 8 Projekt końcowy

### 1. Cele ćwiczenia:

- Utrwalenie składni w GiT
- Zapoznanie się ze standardami przechowywania danych w plikach tekstowych
- Utworzenie projektu informatycznego i zarządzanie jego wersją w GiT -
- Zapoznanie się z Github Actions
- Język programowania dowolny, preferowany Python

### 2: Opis projektu:

Program do konwersji danych obsługujący formaty: .xml, .json i .yaml (.yml)

Sposób użycia: `program.exe pathFile1.x pathFile2.y`

gdzie x i y to jeden z formatów .xml, .json i .yaml (.yml).

Powyższe wywołanie programu powinno prawidłowo rozpoznać format, pobrać dane z `pathFile1.x`, a następnie utworzyć nowy plik `pathFile2.y` i przekonwertować dane zgodnie z nowym formatem.

W przypadku tworzenia oprogramowania w pythonie, z projektu należy wygenerować plik .exe przy pomocy `pyinstaller.exe` (dostępny do instalacji przez pip). Aby uniknąć powstania wielu .dll trzeba wykorzystać flagę `--onefile`:

`pyinstaller.exe --onefile project.py`

Dodatkowo można dołączyć do projektu prosty UI (User Interface), sugerowany jest PyQt (dostępny przez pip).

Przy uruchomieniu programu posiadającego własny UI, dodatkowo zostanie uruchomione okno wiersza poleceń. Aby uniknąć takiego zachowania, trzeba wygenerować plik .exe komendą:

`pyinstaller.exe --onefile --noconsole project.py`

### 3: Zadania do wykonania:

- przygotować repozytorium na githubie

Nazwy branchy według zasady „TaskX” gdzie X to numer taska.

- **Task0:** w katalogu projektu utworzyć pusty skrypt (bash, python, batch, powershell, inny dowolny), np. `installResources.ps1`, przy każdej instalacji przez PIP, należy taką samą komendę umieścić w tym skrypcie. Taki skrypt, zawierający listę potrzebnych komponentów pythona będzie niezbędny na etapie konfiguracji automatycznego budowania projektu.

- **Task1:** parsowanie argumentów przekazywanych przy uruchomieniu programu
- **Task2:** wczytywanie **do obiektu** z pliku .json i weryfikacja poprawności składni pliku
- **Task3:** zapis danych **z obiektu** do pliku w formacie i zgodnie ze składnią .json
- **Task4:** wczytywanie **do obiektu** z pliku .yaml i weryfikacja poprawności składni pliku
- **Task5:** zapis danych **z obiektu** do pliku w formacie i zgodnie ze składnią .yaml
- **Task6:** wczytywanie **do obiektu** z pliku .xml i weryfikacja poprawności składni pliku
- **Task7:** zapis danych **z obiektu** do pliku w formacie i zgodnie ze składnią .xml

\*- **Task8:** utworzenie wersji programu z UI (wygląd UI dowolny)

\*- **Task9:** w wersji UI, wczytywanie i zapis do pliku powinien odbywać się asynchronicznie (wielowątkowo)

**!Podczas pisania oprogramowania należy w rozsądny sposób przewidzieć i zabezpieczyć program przed błędnym działaniem, błędnym zachowaniem użytkownika i błędnymi danymi wejściowymi!**

**!Python posiada gotowe biblioteki do obsługi formatów .json, .xml i .yaml, aby ich używać należy doinstalować odpowiednie komponenty do pythona przez pip!**

- utworzenie na githubie „Actions”, automatycznego budowania pliku .exe z projektu
- wykorzystanie [action/upload-artifact@v3](#) do przesłania utworzonego pliku .exe na repozytorium github
- edycja pliku .yaml według specyfikacji ( [.github/workflows/naszaNazwaWorkflow.yaml](#) ):
- ustawienie nazwy workflow (*name:* ),
- uruchomienie workflow (*schedule:* ):
  - a) automatycznie raz w tygodniu (dzień i godzina dowolna),
  - b) automatycznie po pushowaniu na mastera,
  - c) ręcznie przez usera (*workflow\_dispatch:*),
- ustawienie domyślnego serwera windows (najnowszy dostępny),
- domyślnym shellem windowsa jest PowerShell (.ps1), można zostawić i korzystać z komend PowerShella, lub zmienić na inny shell (bash?) – **tutaj mają Państwo dowolność**,
- po ustawieniu serwera, należy go skonfigurować i zainstalować potrzebne komponenty (np. pip) tutaj się przyda uruchomienie skryptu installResources.ps1 jaki Państwo utworzyli i rozwijali w trakcie projektu.

dokumentacja znajduje się pod linkiem:

<https://docs.github.com/en/actions>

**Na zaliczenie zadania, należy najpóźniej na ostatnich zajęciach oddać projekt, wysyłając link do repozytorium.**