

Wizualizacja i animacja drzewa Binary Search Tree

Mikołaj Bieniek

January 2025

Pliki i ich zawartość

1. `bst.py`

- Ten plik zawiera implementację struktury drzewa binarnego (Binary Search Tree).
- Zawiera on metody takie jak wstawiania, wyszukiwania elementu w drzewie, usuwania elementu oraz usuwania całego drzewa.
- Plik ten jest importowany w `mainwindow.py`, aby umożliwić wizualizację operacji w drzewie.

2. `mainwindow.py`

- Główny plik aplikacji, który importuje klasę wygenerowaną z pliku `.ui` oraz importuje logikę z `bst.py`, umożliwiając użytkownikowi wizualizację i interakcję z drzewem binarnym w interfejsie graficznym

3. `requirements.txt`

- Plik ten zawiera listę wymagań odnośnie uruchomienia projektu, takich jak biblioteki Pythona (W naszym przypadku Pyside6)

Wprowadzenie

Framework **Qt** korzysta z biblioteki **PySide6**, jednak najnowsza wersja Pythona (3.13.1) nie jest jeszcze kompatybilna z PySide6. Dlatego zalecane jest użycie starszej wersji Pythona, np. Python **3.12.8**.

Plik `form.ui` – co to jest?

W naszym folderze znajduje się plik `form.ui`. Jest to plik w formacie **XML**, który został wygenerowany przy użyciu aplikacji Qt. Plik `.ui` opisuje wizualny interfejs użytkownika (GUI). Jednak Python nie jest w stanie bezpośrednio korzystać z takich plików, dlatego konieczne jest ich przekształcenie na plik wykonywalny w Pythonie.

Konwersja pliku `.ui` na `.py`

Aby przekształcić plik `.ui` na plik Python (`.py`), używamy narzędzia **pyside6-uic** (User Interface Compiler). Dzięki temu jesteśmy w stanie wygenerować kod w Pythonie, który odpowiada strukturze GUI opisanej w pliku `form.ui`. Ta komenda przekształca nasz plik `form.ui` w plik `py_form.py`

Komenda

```
pyside6-uic form.ui -o py_form.py
```

`form.ui` - nazwa naszego pliku do przekonwertowania

`py_form.py` - nazwa przekonwertowanego pliku

Aktualizacja pliku w projekcie

Jeśli będziemy chcieli zmienić nazwę wygenerowanego pliku `.py`, musimy również zaktualizować odniesienia do tego pliku w innych częściach projektu. Na przykład w pliku `mainwindow.py` importujemy klasę z wygenerowanego pliku:

```
from py_form import Ui_MainWindow
```

Uruchamianie programu

Aby uruchomić program, wystarczy wykonać następującą komendę w terminalu:

```
python3 mainwindow.py
```

Dodatkowe informacje - konwersja pliku

W naszym folderze znajduje się już przekonwertowany plik `py_form.py`.

Oznacza to, że konwersja pliku `.ui` na `.py` nie jest konieczna, o ile nie wprowadzamy zmian w interfejsie użytkownika. Jeśli będziemy chcieli wprowadzić zmiany w naszym interfejsie graficznym, zmian musimy dokonywać w pliku `form.ui` za pomocą aplikacji **Qt**, a następnie zaktualizowany plik przekonwertować z powrotem z `.ui` na `.py` za pomocą odpowiedniej komendy.