GUI - Flet

Flet to biblioteka bazująca na Flutterze. Umożliwia tworzenie aplikacji z graficznym interfejsem użytkownika na urządzenia stacjonarne i mobilne w Pythonie. Dostarcza gotowe komponenty umożliwiające budowanie graficznych interfejsów użytkownika.

Ponieważ Flutter został stworzony w Google to interfejs stworzony z jego pomocą ma dość charakterystyczny wygląd wpisujący się w tzw. Material Design.

Dużą zaletą Flet-a jest duża ilość dobrze opisanych domyślnych ustawień - stworzone są one w taki sposób, że edytory kodu "podpowiadają" nam jakie mamy możliwości.

Instalacja Flet-a

```
pip3 install flet[all]
```

Ściągnie to wszystkie zależności i dodatkowe narzędzia używane podczas korzystania z frameworku Flet.

Przez framework rozumiemy szablon z zestawem komponentów wykorzystywanych do budowy aplikacji.

Bazowy schemat aplikacji

Aplikacje w Flet-ie można budować m.in. stosując podejście obiektowe.

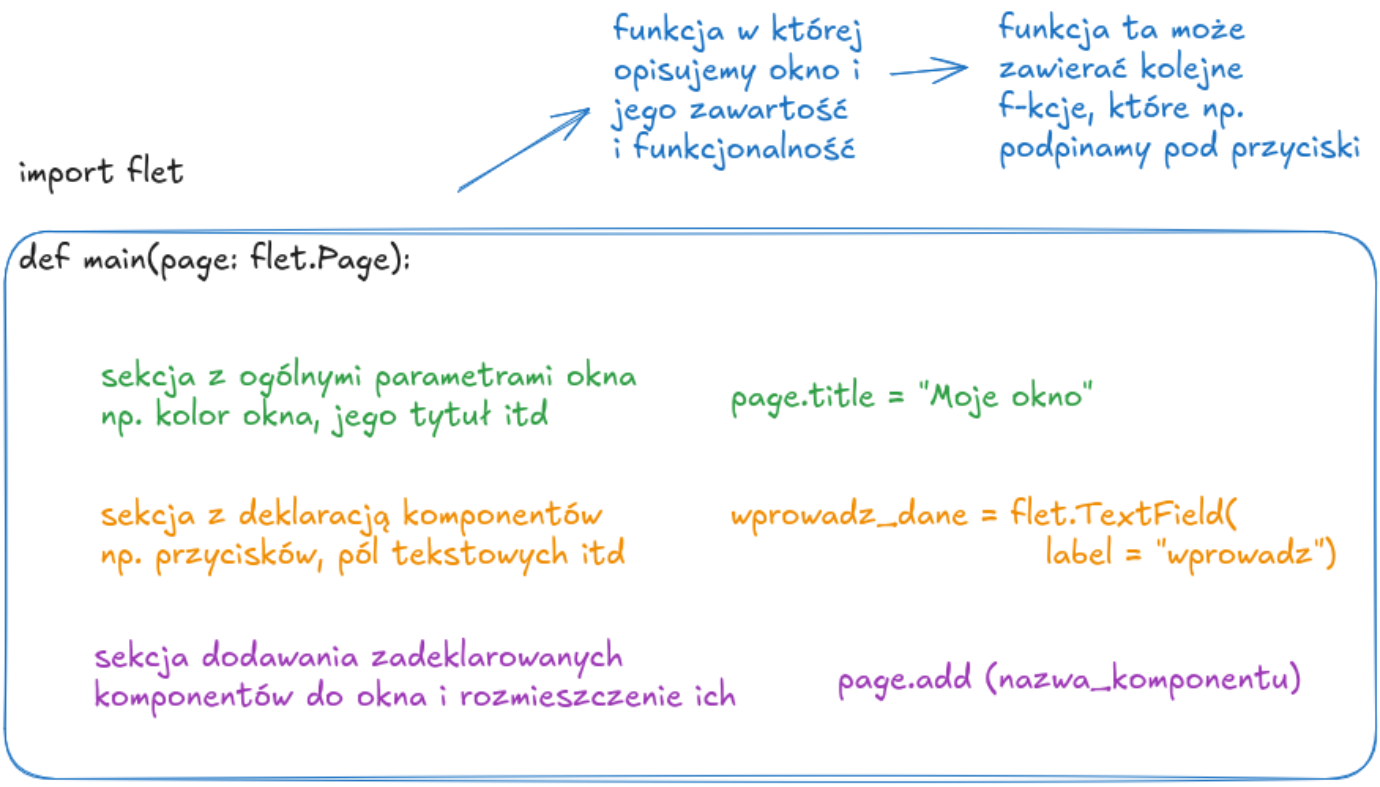
W przypadku początkowych aplikacji można jednak podejść do tego prościej.

Można np. stworzyć funkcję opisującą okno programu, w niej z kolei będziemy deklarować wszystkie komponenty.

Takim komponentem może być przycisk, pole tekstowe itd.

Następnie należy dodać te zadeklarowane elementy do okna aplikacji (poprzez page.add).

Na samym końcu z kolei wywołamy naszą funkcję.




```
import flet


def main(page: flet.Page):
    # Sekcja ogólnych ustawień okna aplikacji

    page.title = "Tytul"  # np. tytulu okna

    # Sekcja deklaracji komponentów
    # - czyli opisujemy np. jakie pola tekstowe,
    # przyciski etc mają się znaleźć w oknie i jak
    # mają one wyglądać.

    tekst = flet.Text("Jakiś tekst.") # np.tekst
    przycisk = flet.FilledButton("wciśnij mnie") # np. przycisk

    # Sekcja umieszczania komponentów w oknie
    # - czyli decydujemy jaki będzie ich układ,
    # w jakiej kolejności się będą wyświetlać,
    # gdzie zostaną umieszone
    page.add(
```

```
        tekst,
        przycisk,
    )


flet.app(main)
```

Czyli importujemy aplikację, tworzymy funkcję odpowiadającą za zbudowanie okna (czyli np. dodanie pól tekstowych, przycisków itd). Następnie funkcję tę wywołujemy za pomocą `flet.app`.

WAŻNE!!! Aplikację zbudowanej z framework-iem Flet lepiej uruchamiać przez `flet run nazwapliku.py`, a nie przez guzik "play" w VSCode czy `python nazwapliku.py`. Uruchomienie poprzez `flet run` powoduje, że nie musicie cały czas zamykać i otwierać okna po zaktualizowaniu aplikacji - zmiany wczytają się automatycznie.

Lista komponentów

Flet udostępnia całą masę flutterowych elementów - przycisków, pól do wpisywania danych, różnej maści menu itd.

Informacje na ich temat można znaleźć tutaj:

https://flet.dev/docs/controls

Najbardziej podstawowe komponenty to m.in.:

- `Text` . Nieinteraktywny tekst wewnątrz okna
- `TextField`. Pole do wprowadzania danych
- `FilledButton`. Przycisk wypełniony kolorem, z zaokrąglonymi bokami.

W praktyce

Przykładowa aplikacja wyświetlająca komunikat powitalny:

```
import flet


def strona_glowna(page: flet.Page):
```

```
    # funkcja wykorzystywana w oknie, podpieta do przycisku
    def przywitaj(_event):
        if imie_pole.value:
            imie = imie_pole.value
            komunikat_tekst.value = f"Witaj {imie}!"
            komunikat_tekst.update()

    # sekcja ustawien okna
    page.title = "Hello!"

    # sekcja komponentów
    imie_pole = flet.TextField(
        label="Wprowadź imię:",
    )
    witaj_przycisk = flet.FilledButton(
        text=" przywitaj ",
        on_click=przywitaj,
        style=flet.ButtonStyle(padding=10),
    )
    komunikat_tekst = flet.Text("")

    # sekcja umieszczenia komponentów w oknie
    page.add(
        imie_pole,
        witaj_przycisk,
        komunikat_tekst,
    )



flet.app(strona_glowna)
```

Warto zwrócić uwage na sposób podpięcia funkcji "przywitaj" do przycisku "witaj_przycisk" - czyli wciśnięcie przycisku spowoduje uruchomienie funkcji.

Wymaga to dodania "wyzwalacza" do przycisku - w tym wypadku "on_click" (gdyż chcemy aby funkcja była uruchamiana po wciśnięciu przycisku, są też inne wyzwalacze np. hover). Ponieważ chcemy żeby to Flet obsłużył uruchomienie funkcji to należy zwrócić uwagę, że po podaniu nazwy funkcji (w powyższym przykładzie `przywitaj`) NIE ma nawiasów okrągłych.

Drugą ciekawą rzeczą jest "tajemniczy" parametr `_event` dodawany do funkcji uruchaminej przez przycisk. Nazwę _event można zmienić, ale należy pamiętać, że musi się on pojawić jako pierwszy

parametr funkcji wyzwalanej przez jakiś komponent graficzny aplikacji (przycisk etc).

Wersja #6
Utworzono 2025-08-08 12:56:53 UTC przez Przemek Jeske
Zaktualizowano 2025-11-02 15:45:22 UTC przez Przemek Jeske