

Quarto - wprowadzenie

Spis treści

1	Wstęp	1
2	Krok pierwszy - metadane dokumentu.	2
3	Edycja kodu	3
3.1	Wykresy	3
3.2	Tabele z danych	4
4	Zaawansowane funkcje	5
4.1	Zapis matematyczny	5
4.2	Odwolania krzyzowe	5
5	Uwagi końcowe, good to know	6

1 Wstęp

Witaj we wprowadzeniu do Quarto! Plik, który właśnie teraz czytasz powstał tylko przy użyciu tego narzędzia.

Czym jest quarto?

To narzędzie open source, które generuje eleganckie prezentacje, pliki PDF, world i inne. Integruje ze sobą markdown i kod działający na jupyterze. Można w ramach jednego dokumentu operować na kilku językach programowania oraz korzystać ze wszystkich funkcji formatowania tekstu jakie oferuje markdown.

Proces instalacji jest prosty:

1. Pobierz CLI do quarto z <<https://quarto.org/>>,
2. Wybierz swoje środowisko pracy (jupyter, vscode r studio lub inne z listy wspieranych),
3. Utwórz plik z rozszerzeniem .qmd,
4. Rozpocznij edycję.

Warto też utworzyć środowisko wirtualne do pracy i śledzić co będzie się działo w terminalu, ponieważ znajdują się tam istotne z naszej perspektywy informacje, jak na przykład adres pod którym będziemy mogli w przeglądarce sprawdzać na żywo zmiany w dokumencie. Lista komend i innych istotnych informacji znajduje się pod linkiem <https://quarto.org/docs/get-started/>.

2 Krok pierwszy - metadane dokumentu.

Aby plik, który będziesz edytować poprawnie się uruchamiał, należy najpierw zainstalowanemu quarto zdefiniować czym tak w zasadzie ma być nasz plik.

Jak wygląda takie podstawowe zdefiniowanie metadanych?

Poniżej znajduje się przykład poprawnego nagłówka YAML:

```
---
title: "Mój Raport"
format: html
execute:
  echo: false
  warning: false
---
```

Omówmy to sobie krok po kroku:

- title - nazwa dokumentu, domyślnie będzie się wyświetlała jako nagłówek w pliku,
- format - opisuje w jakim formacie będzie nasz dokument. Do wyboru są rozszerzenia pdf, html, gfm (github markdown) i docx,
- execute - to najbardziej rozbudowana sekcja. Daje ona instrukcje quarto w jaki sposób ma się obchodzić z kodem.

- echo - decyduje o tym, czy wykonywany kod również będzie pojawiać się w pliku, czy tylko rezultat powstałego po jego wywołaniu,
- warning - ukrywa ostrzelenia wydawane przez pythona (na przykład o przestarzałej wersji jakiejś biblioteki). Polepsza przejrzystość dokumentu.

Inne przydatne deklaracje w permabule to na przykład lang - język dokumentu (zmienia niektóre fragmenty generowanie automatycznie), author - autor, date - dzisiejsza data, theme - uzyty motyw.

Kiedy juz zdefiniujemy początkowo nasz dokument mozemy zacząć spisywać tekst jak gdyby nigdy nic pamiętając oczywiście, ze pod maską naszego dokumentu pracuje markdown. Warto znać jego funkcje, ponieważ pomimo bycia nieco okrojonym względem TeXa, jest wciąż w stanie w stopniu zadowalającym zaspokoić większość potrzeb, które byśmy mieli tworząc raport lub podobny temu dokumentowi skrypt.

3 Edycja kodu

3.1 Wykresy

W Quarto kod umieszczamy w specjalnych blokach. Najważniejsze są opcje zapisu (tzw. “hashpipe” #|), które kontrolują zachowanie bloku:

```
```{python}
#| label: fig-wykres
#| fig-cap: "Wykres funkcji kwadratowej"
#| echo: true

import matplotlib.pyplot as plt
import matplotlib

Wymuszenie generowania wykresu wewnątrz dokumentu
%matplotlib inline

plt.figure(figsize=(8, 4))
plt.plot([1, 2, 3, 4], [1, 4, 9, 16], marker='o', color='royalblue')
plt.title("Przykładowy wykres")
plt.grid(True)

plt.show()
```

```

- **label**: unikalny identyfikator (pozwala na odwołania w tekście),
- **fig-cap**: podpis pod wykresem,
- **echo**: czy pokazywać kod

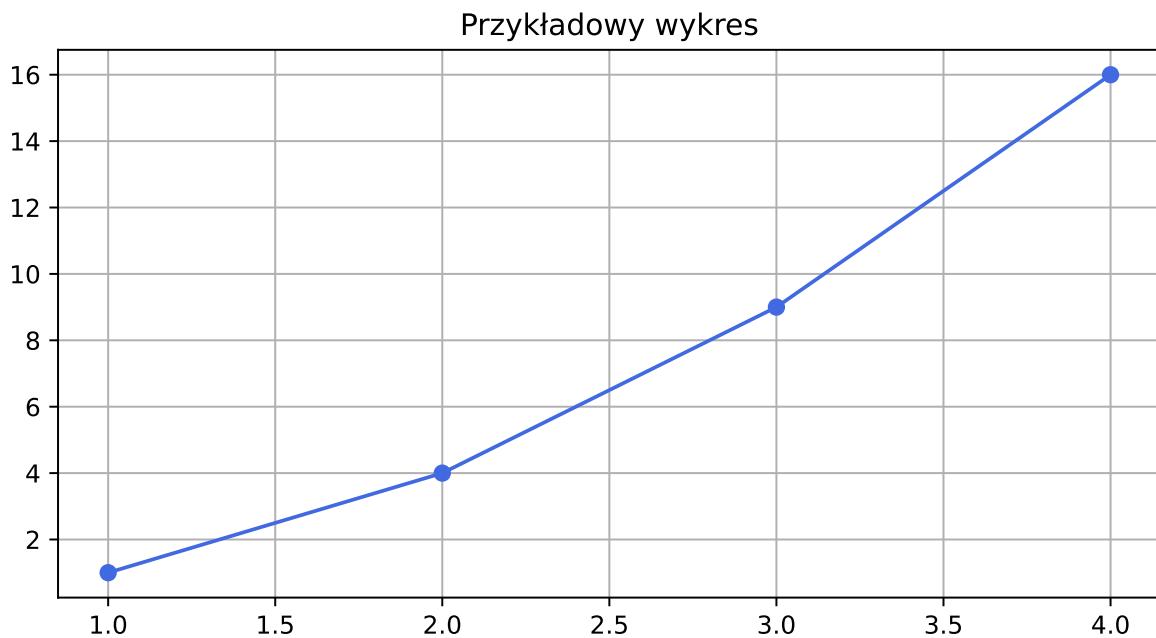
Zobaczmy co dzieje się po wywołaniu takiego kodu

```
import matplotlib.pyplot as plt
import matplotlib

%matplotlib inline

plt.figure(figsize=(8, 4))
plt.plot([1, 2, 3, 4], [1, 4, 9, 16], marker='o', color='royalblue')
plt.title("Przykładowy wykres")
plt.grid(True)

plt.show()
```



Rysunek 1: Wykres funkcji kwadratowej

3.2 Tabele z danymi

Zamiast rysować tabele ręcznie, możemy użyć biblioteki `pandas`. Quarto automatycznie zadba o ich wygląd.

```
import pandas as pd

dane = pd.DataFrame({
    "Metoda": ["Regresja", "Lasy Losowe", "Sieci neuronowe"],
    "Wynik (R2)": [0.85, 0.92, 0.94],
    "Czas (s)": [0.1, 12.5, 340.2]
})

dane
```

Tabela 1: Wybrane dane z ramki Pandas

| | Metoda | Wynik (R2) | Czas (s) |
|---|-----------------|------------|----------|
| 0 | Regresja | 0.85 | 0.1 |
| 1 | Lasy Losowe | 0.92 | 12.5 |
| 2 | Sieci neuronowe | 0.94 | 340.2 |

W ten sposób omówiliśmy dwie struktury kluczowe do tworzenia raportów. Ze znajomością powyższych sekcji jesteście w stanie na spokojnie tworzyć na przykład sprawozdania na fizykę (a jest to wyjątkowo uciążliwe) oraz raporty z analizy danych, co przyda się na zajęciach ze statystyki.

4 Zaawansowane funkcje

4.1 Zapis matematyczny

Skoro posługujemy się markdownem, możemy korzystać z jego wbudowanego modułu obsługi wzorów matematycznych napisanych w składni \LaTeX . Mozesz dodać wzór w tekście korzystając z symbolu ” $\$$ ” lub dodać go w oddzielnej linijce korzystając z podwójnego dolara.

Oto jak zapisać całkę w Quarto:

```
$\int_{a}^{b} f(x) \, dx$  
Dla wzoru w tekście  
  
$$  
\int_{a}^{b} f(x) \, dx  
$$  
Dla oddzielnej linijki
```

Więcej opcji opisanych jest bezpośrednio na stronach poświęconych L^AT_EX, na przykład <https://quickref.me/latex>.

4.2 Odwołania krzyzowe

Jeśli nadasz wykresowi etykietę (np. #| label: fig-wykres), możesz napisać w tekście:

```
"Jak widać na @fig-wykres...".
```

Quarto automatycznie wstawi tam numer rysunku i stworzy hiperłącze.

5 Uwagi końcowe, good to know

Quarto obsługuje interaktywne wykresy w plotly o ile tworzymy nasz raport w html, możemy dzięki temu tworzyć proste i estetyczne dashboardy.

Na różnych komputerach, systemach, a także środowiskach program jest w stanie płatać różne figle i stawiać opór przy renderowaniu pliku lub w ogóle go nie renderować. Nie przejmujcie się, jeśli będziecie musieli uciec się do pomocy asystenta AI przy próbie postawienia tego wszystkiego na nogi.

PS:

Skrypt ten ma przede wszystkim przybliżyć ideę oraz podstawy quarto. Kazdy projekt jest inny i wymaga uzycia różnych funkcji, dlatego nie ma większego sensu w próbie zawarcia wszystkiego na raz w jednym PDF'ie. Podane przeze mnie linki w dokumencie pomagają uczyć się na własną rękę i zaspokajać ciekawość potencjalnego użytkownika. Dziękuję za poświęcony czas, Wesołych Świąt!