

# STREAMLIT

## Czym jest Streamlit?

Streamlit to otwarty framework do tworzenia aplikacji internetowych opartych na danych w języku Python. Umożliwia szybkie tworzenie interaktywnych aplikacji internetowych, które wykorzystują analizę danych, uczenie maszynowe i wizualizację danych. Streamlit umożliwia programistom tworzenie aplikacji internetowych za pomocą prostego interfejsu opartego na skryptach Pythona, bez potrzeby znajomości języków front-endowych, takich jak HTML, CSS czy JavaScript.

Główne cechy Streamlit to:

- **Prostota użycia:** Streamlit jest zaprojektowany w taki sposób, aby umożliwić programistom tworzenie aplikacji internetowych za pomocą prostych i czytelnych skryptów Pythona.
- **Dynamiczna aktualizacja:** Aplikacje Streamlit automatycznie aktualizują się w czasie rzeczywistym w odpowiedzi na zmiany w kodzie źródłowym, co pozwala na szybkie eksperymentowanie i iterowanie.
- **Bogata biblioteka komponentów:** Streamlit zawiera bogatą bibliotekę gotowych komponentów, takich jak przyciski, pola wyboru, wykresy czy wykresy interaktywne, które można łatwo włączyć do aplikacji.
- **Wsparcie dla popularnych bibliotek Pythona:** Streamlit jest zintegrowany z wieloma popularnymi bibliotekami Pythona, takimi jak pandas, matplotlib, scikit-learn czy TensorFlow, co ułatwia analizę danych i uczenie maszynowe.

Dzięki tym cechom Streamlit jest popularnym narzędziem do szybkiego tworzenia prototypów, budowy dashboardów danych, udostępniania analiz i modeli maszynowych oraz tworzenia interaktywnych wizualizacji danych.

## Jak użyć Streamlit

### Instalacja Streamlit

#### macOS i Linux:

Użyj komendy w terminalu:

```
pip install streamlit
```

Test streamlit:

```
streamlit hello
```

## Windows:

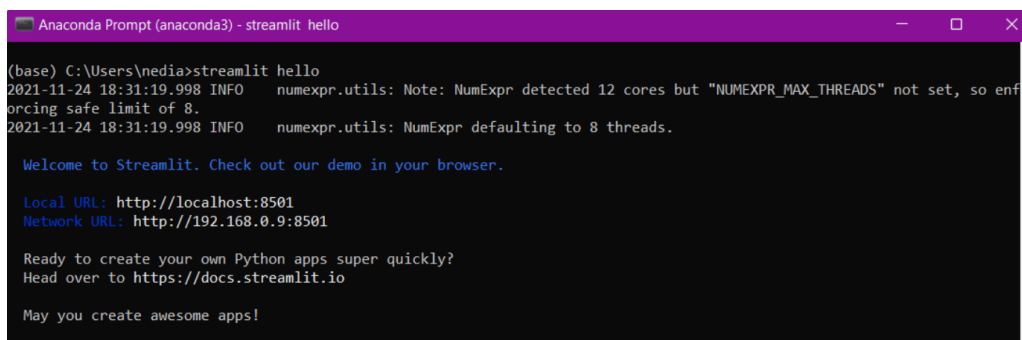
1. Zainstaluj Anacondę i stwórz swoje środowisko
2. Otwórz terminal i wpisz:

```
pip install streamlit
```

Test streamlit:

```
streamlit hello
```

Po wpisaniu tego polecenia w terminalu poniższa strona powinna otworzyć się automatycznie:



```
Anaconda Prompt (anaconda3) - streamlit hello

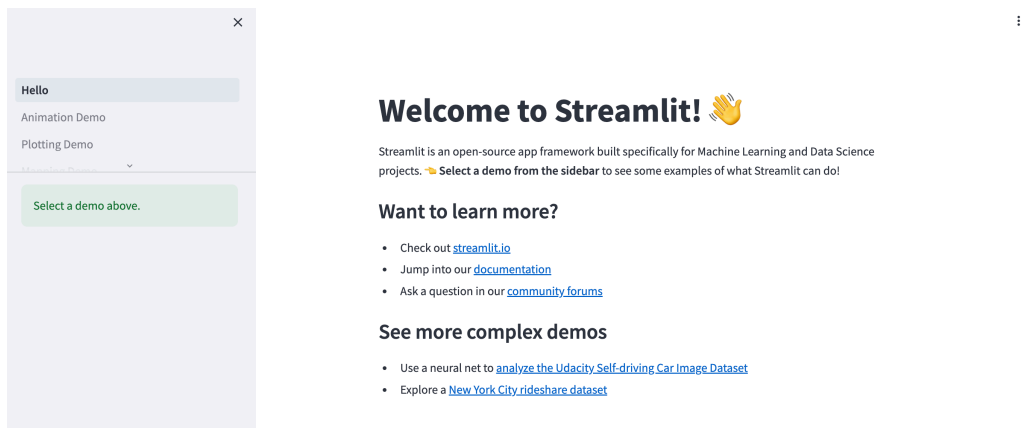
(base) C:\Users\nedia>streamlit hello
2021-11-24 18:31:19.998 INFO     numexpr.utils: Note: NumExpr detected 12 cores but "NUMEXPR_MAX_THREADS" not set, so enforcing safe limit of 8.
2021-11-24 18:31:19.998 INFO     numexpr.utils: NumExpr defaulting to 8 threads.

Welcome to Streamlit. Check out our demo in your browser.

Local URL: http://localhost:8501
Network URL: http://192.168.0.9:8501

Ready to create your own Python apps super quickly?
Head over to https://docs.streamlit.io

May you create awesome apps!
```



## Jak uruchomić kod Streamlit

```
streamlit run file.py
```

## Wyświetlanie tekstu

Na początku zobaczymy, jak dodać tekst do aplikacji Streamlit i jakie są różne polecenia służące do dodawania tekstu.

```
st.write() :
```

Ta funkcja służy do dodawania czegokolwiek do aplikacji internetowej, od sformatowanego ciągu po wykresy w postaci figur Matplotlib, wykresów Altair, figury ploty, ramki danych, modelu Keras i innych.

```
import streamlit as st
st.write("Hello ,your first app")
```

Hello ,your first app

```
st.title() :
```

Ta funkcja umożliwia dodanie tytułu aplikacji.

```
st.header() :
```

Ta funkcja służy do ustawienia nagłówka sekcji.

```
st.markdown() :
```

Możesz wyświetlać tekst w formacie Markdown.

```
st.subheader() :
```

Ta funkcja służy do ustawienia pod-nagłówka sekcji.

```
st.caption() :
```

Służy do dodawania podpisów do różnych elementów w aplikacji.

```
st.code() :
```

Służy do wyświetlania bloku kodu źródłowego.

```
st.latex() :
```

Ta funkcja służy do wyświetlania wyrażeń matematycznych sformatowanych jako LaTeX.

```
import streamlit as st

st.title("This function allows you to add title of the app.")
st.markdown("This function is used to set a markdown of a section.")
st.header("This function is used to set header of section.")
st.subheader("This function is used to set sub-header of a section.")
st.caption("This function is used to write caption.")
st.code("x=abcd")
st.latex("a^2+b^2=c^2")
```

## This function allows you to add the title of the app.

This function is used to set a markdown of a section.

## This function is used to set header of a section.

## This function is used to set sub-header of a section.

This function is used to write caption.

x=abcd

$$a^2 + b^2 = c^2$$

## Wyświetl obraz, plik wideo lub audio za pomocą Streamlit

Nie ma tak prostych funkcji jak funkcje Streamlit do wyświetlania obrazów, filmów i plików audio. Przyjrzyjmy się, jak wyświetlać multimedia za pomocą Streamlit!

```
st.image() :
```

Służy do wyświetlania obrazów.

```
st.audio() :
```

Służy do odtwarzania plików dźwiękowych.

```
st.video() :
```

Służy do odtwarzania plików video.

```
import streamlit as st
st.image("dogs.png")
```

## Widżety

Widżety są najważniejszymi elementami interfejsu użytkownika. Streamlit ma różne widżety, które umożliwiają włączenie interaktywności bezpośrednio do aplikacji za pomocą przycisków, suwaków, wprowadzania tekstu i nie tylko.

`st.checkbox()` :

Ta funkcja zwraca wartość logiczną. Gdy pole jest zaznaczone, zwracana jest wartość True, w przeciwnym razie wartość False.

`st.button()` :

Służy do tworzenia przycisku interaktywnego.

`st.radio()` :

Służy do tworzenia interaktywnych przycisków typu radio.

`st.selectbox()` :

Służy do tworzenia interaktywnego rozwijanego menu w aplikacji. Menu to umożliwia użytkownikowi wybór jednej opcji spośród wielu dostępnych.

`st.multiselect()` :

Służy do tworzenia interaktywnego wielokrotnego wyboru.

`st.select_slider()` :

Ta funkcja służy do wyświetlania widżetu suwaka wyboru.

`st.slider()` :

Służy do tworzenia interaktywnego suwaka numerycznego.

```
st.checkbox('yes')
st.checkbox('no')
st.button('Click')
st.radio('Pick your gender',['Male','Female'])
st.selectbox('Pick your gender',['Male','Female'])
st.multiselect('Choose an animal',['dog','cat','mouse'])
st.select_slider('Pick a mark',['Bad','Good','Excellent'])
st.slider('Pick a number', 0,50)
```

☐ yes☐ noClick

Pick your gender

☒ Male☐ Female

Pick your gender

Male

Choose an animal

Choose an option

Pick a mark

Bad



Bad

Excellent

Pick a number

0



0

50

```
st.number_input():
```

Służy do tworzenia interaktywnego pola wprowadzania liczby.

```
st.text_input():
```

Służy do tworzenia interaktywnego pola wprowadzania tekstu. Pole to umożliwia użytkownikowi wprowadzenie tekstu za pomocą klawiatury.

```
st.date_input():
```

Służy do tworzenia interaktywnego pola wprowadzania daty. Pole to umożliwia użytkownikowi wybór daty za pomocą kalendarza..

```
st.time_input():
```

Służy do tworzenia interaktywnego pola wprowadzania czasu. Pole to umożliwia użytkownikowi wybór czasu za pomocą rozwijanej listy lub poprzez bezpośrednie wprowadzenie godziny i minuty.

```
st.text_area():
```

Służy do tworzenia interaktywnego pola wprowadzania wielolinijkowego tekstu.

```
st.file_uploader():
```

Służy do tworzenia interaktywnego elementu umożliwiającego użytkownikowi przesyłanie plików do aplikacji. Ten element umożliwia użytkownikom przekazywanie plików z lokalnego systemu plików do aplikacji..

```
st.color_picker():
```

Służy do tworzenia interaktywnego narzędzia wybierania kolorów. Pozwala użytkownikowi na wybór koloru za pomocą interaktywnego kolorowego selektora..

```
st.number_input('Pick a number', 0,10)
st.text_input('Email address')
st.date_input('Travelling date')
st.time_input('School time')
st.text_area('Description')
st.file_uploader('Upload a photo')
st.color_picker('Choose your favorite color')
```

Pick a number

0

Email address

Travelling date

2023/11/27

School time

00:19

Description

Upload a photo

Drag and drop file here

Limit 200MB per file

Browse files

Choose your favorite color

## Wyświetlaj postęp i status za pomocą Streamlit

```
st.balloons():
```

Jest interaktywnym narzędziem, które wyświetla animowane balony na ekranie aplikacji, tworząc efekt wizualny przyciągający uwagę użytkownika.

```
st.progress():
```

Służy do tworzenia paska postępu. Pasek ten pozwala wizualnie prezentować postęp w procesie, co jest przydatne szczególnie podczas wykonywania długotrwałych operacji lub obliczeń, aby użytkownik mógł śledzić, jak długo jeszcze trwać będzie proces..

```
st.spinner():
```

Służy do wyświetlania animowanego wskaźnika "spinner", który wskazuje, że aplikacja jest zajęta lub przetwarza dane..

```
st.balloons()  
st.subheader('Progress...')  
st.progress(10)  
with st.spinner('Wait for it...'):  
    time.sleep(10)
```

```
st.success():
```

Służy do wyświetlania wiadomości informującej o sukcesie lub pozytywnym wyniku operacji.

```
st.error():
```

Służy do wyświetlania wiadomości informującej o błędzie lub niepowodzeniu konkretnej operacji.

```
st.info():
```

Służy do wyświetlania ogólnych informacji lub komunikatów użytkownikowi.

```
st.exception():
```

Służy do wyświetlania szczegółowych informacji o wyjątku (błędzie).

```
st.success("You did it !")  
st.error("Error")  
st.info("It's easy to build a streamlit app")  
st.exception(RuntimeError("RuntimeError exception"))
```

## Wyświetlaj wykresy za pomocą Streamlit

```
st.pyplot():
```

Umożliwia wyświetlanie wykresów z biblioteki Matplotlib w aplikacji.

```
import streamlit as st  
import matplotlib.pyplot as plt  
import numpy as np  
  
rand=np.random.normal(1, 2, size=20)  
fig, ax = plt.subplots()  
ax.hist(rand, bins=15)  
st.pyplot(fig)
```

```
st.line_chart():
```

Służy do wyświetlania wykresu liniowego.



```
import streamlit as st
import pandas as pd
import numpy as np

df= pd.DataFrame(
    np.random.randn(10, 2),
    columns=['x', 'y'])
st.line_chart(df)
```

`st.bar_chart()`:  
Służy do wyświetlania wykresu słupkowego.

```
import streamlit as st
import pandas as pd
import numpy as np

df= pd.DataFrame(
    np.random.randn(10, 2),
    columns=['x', 'y'])
st.bar_chart(df)
```

`st.area_chart()`:  
Służy do wyświetlania wykresu obszarowego w aplikacji. Wykres obszarowy jest podobny do wykresu liniowego, ale obszar pod linią jest wypełniony kolorem, co umożliwia wizualizację trendów oraz różnic między danymi.

```
import streamlit as st
import pandas as pd
import numpy as np

df= pd.DataFrame(
    np.random.randn(10, 2),
    columns=['x', 'y'])
st.area_chart(df)
```

`st.altair_chart()`:  
Służy do wyświetlania wykresów przy użyciu biblioteki Altair w aplikacji. Altair jest biblioteką do wizualizacji danych w języku Python, która umożliwia tworzenie interaktywnych i estetycznych wykresów.

```
import streamlit as st
import numpy as np
import pandas as pd
import altair as alt

df = pd.DataFrame(np.random.randn(500, 3), columns=['x', 'y', 'z'])
c = alt.Chart(df).mark_circle().encode(x='x' , y='y' , size='z',
color='z', tooltip=['x', 'y', 'z'])
```

```
st.altair_chart(c, use_container_width=True)
```

```
st.graphviz_chart():
```

Służy do wyświetlania grafów wygenerowanych przy użyciu biblioteki Graphviz w aplikacji. Graphviz to narzędzie do automatycznego generowania grafów i struktur danych, które jest często używane do wizualizacji relacji między elementami.

```
import streamlit as st
import graphviz as graphviz

st.graphviz_chart('''
    digraph {
        Big_shark -> Tuna
        Tuna -> Mackerel
        Mackerel -> Small_fishes
        Small_fishes -> Shrimp
    }
''')
```

### 1. Prosty quiz:

Utwórz mini-quiz z kilkoma pytaniami. Użytkownik powinien być w stanie odpowiedzieć na pytania i na koniec zobaczyć wynik.

### 2. Konwerter jednostek:

Zbuduj aplikację, która umożliwi konwersję pomiędzy różnymi jednostkami, takimi jak kilogramy na funty, kilometry na mile, stopnie Celsjusza na stopnie Fahrenheita itp.

### 3\*. Lista zakupów:

Utwórz prostą aplikację z listą zakupów. Użytkownik powinien mieć możliwość dodawania i usuwania produktów z listy.