

Aplicativos WEB com



Streamlit

Live de Python



1. O Streamlit

Instalação e conceitos básicos

2. Nosso primeiro APP

Um removedor de fundos de fotos

3. Extensões

Plugins externos ao streamlit

4. Deploy

Colocando um APP em produção



picpay.me/dunossauro



apoia.se/livedepython



pix.dunossauro@gmail.com



Ajude o projeto <3



A2n, Ademir Peixoto, Adilson Herculano, Adriana Cavalcanti, Adriano Ferraz, Alexandre Harano, Alexandre Lima, Alexandre Souza, Alexandre Takahashi, Alexandre Villares, Alex Lima, Allan Almeida, Alynne Ferreira, Alysso Oliveira, Ana Carneiro, Andre Azevedo, André Rafael, Aquiles Coutinho, Arnaldo Turque, Aurelio Costa, Bruno Batista, Bruno Divino, Bruno Freitas, Bruno Guizi, Bruno Lopes, Bruno Ramos, Caio Felix, Caio Nascimento, Carina Pereira, Christiano Moraes, Clara Battesini, Dandara Sousa, Daniel Freitas, Daniel Haas, Daniel Santos, Danilo Segura, David Couto, David Kwast, Delton Porfiro, Denis Quirino, Diego Farias, Diego Guimarães, Dilenon Delfino, Dino Aguilar, Diogo Paschoal, Douglas Bastos, Douglas Zickuhr, Eduardo Tolmasquim, Elton Silva, Emanuel Betcel, Emerson Rafael, Eneas Teles, Erick Ritir, Érico Andrei, Eugenio Mazzini, Euripedes Borges, Everton Silva, Fabiano Tomita, Fabio Barros, Fábio Barros, Fabio Castro, Fábio Thomaz, Fabricio, Fabricio Araujo, Felipe Rodrigues, Fernanda Prado, Fernando Florêncio, Firehouse, Flávio Meira, Flavkaze, Gabriel Barbosa, Gabriel Mizuno, Gabriel Nascimento, Gabriel Simonetto, Geandreson Costa, Guilherme Felitti, Guilherme Gall, Guilherme Ostrock, Guilherme Piccioni, Guilherme Silva, Gustavo Suto, Harold Gautschi, Heitor Fernandes, Henrique Junqueira, Hugo Cosme, Igor Taconi, Ismael Ventura, Italo Silva, Jairo Jesus, Jairo Lenfers, Janael Pinheiro, João Paulo, Joelson Sartori, Johnny Tardin, Jônatas Silva, José Barbosa, José Gomes, Joseíto Júnior, Jose Mazolini, José Pedro, Juan Gutierrez, Juliana Machado, Julio Franco, Júlio Gazeta, Júlio Pereira, Julio Silva, Kaio Peixoto, Kaneson Alves, Leandro Miranda, Leonardo Mello, Leonardo Nazareth, Leon Solon, L. Perciliano, Luancomputacao Roger, Luã Vacaro, Lucas Adorno, Lucas Carderelli, Lucas Mello, Lucas Mendes, Lucas Nascimento, Lucas Schneider, Lucas Simon, Lucas Valino, Luciano Ratamero, Luciano Silva, Luciano Teixeira, Luiz Junior, Luiz Lima, Luiz Paula, Maicon Pantoja, Maiquel Leonel, Marcelino Pinheiro, Márcio Martignoni, Marcio Moises, Marco Mello, Marcos Gomes, Marco Yamada, Maria Clara, Maria Gabriela, Marina Passos, Matheus Cortezi, Matheus Oliveira, Matheus Silva, Matheus Vian, Mauricio Fagundes, Mauricio Nunes, Mirian Batista, Mlevi Lsantos, Murilo Andrade, Murilocunha, Murilo Viana, Nando Sangenetto, Natan Cervinski, Nathan Branco, Nicolas Teodosio, Osvaldo Neto, Otávio Carneiro, Patricia Minamizawa, Patrick Felipe, Paulo D., Paulo Tadei, Pedro Henrique, Pedro Pereira, Pedro Silva, Peterson Santos, P Muniz, Priscila Santos, Rafael Lopes, Rafael Romão, Ramayana Menezes, Regis Santos, Regis Tomkiel, Rene Bastos, Ricardo Silva, Ricarte Jr, Riverfount, Robson, Robson Maciel, Rodrigo Alves, Rodrigo Cardoso, Rodrigo Freire, Rodrigo Messias, Rodrigo Quiles, Rodrigo Ribeiro, Rodrigo Vaccari, Rodrigo Vieira, Rogério Lima, Rogério Nogueira, Rogério Sousa, Ronaldo Silva, Ronaldo Silveira, Rui Jr, Samanta Cicilia, Sebastião Tolentino, Talita Rossari, Tay Turnner, Thaynara Pinto, Thi, Thiago Araujo, Thiago Borges, Thiago Curvelo, Thiago Moraes, Thiago Salgado, Thiago Souza, Tiago Minuzzi, Tiago Souza, Tony Dias, Tony Santos, Tyrone Damasceno, Uadson Emile, Valcilon Silva, Valdir Tegon, Vcwild, Vinícius Bastos, Vinicius Stein, Vitor Luz, Vladimir Lemos, Walter Reis, Wesley Mendes, Willian Lopes, Wilson Duarte, Wilson Neto, Wilson Rocha, Xico Silvério, Yury Barros



Obrigado você



Uma introdução

Stre
amlit

O Streamlit



“Uma maneira mais rápida de criar e compartilhar aplicativos de dados.”

Streamlit transforma scripts em python para aplicativos web. Tudo usando Python. Sem nenhuma necessidade de entender de front-end ou back-end.

- Licença Apache 2.0
- Primeira release (0.1) em Março de 2018
- Atualmente na versão 1.15.1 (data da live)
- Focado em aplicativos para ciência de dados e machine learning
- API extremamente simples
- Integração com bibliotecas de mercado
 - pandas, numpy, matplotlib, bokeh, plottly, etc.
- Uma gama de plugins externos

<https://dunossauro-test-streamlit-app-ly9jox.streamlit.app/>



Um demo vale mais que mil palavras



```
pip install streamlit
```



Instalação



streamlit hello



Teste!

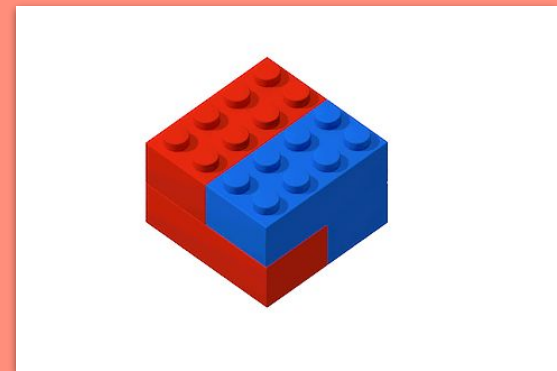


Os Widgets



O streamlit conta com diversos tipos de widgets. Eles são dispostos em 7 categorias:

- Texto
- Exibição de dados
- Gráficos
- Inputs
- Midia
- Status / Progresso
- Layouts



Um exemplo básico



```
1  # exemplo_00.py
2  import streamlit as st
3
4  st.write('Olá Mundo')
5  st.image('https://via.placeholder.com/500x400/')
```

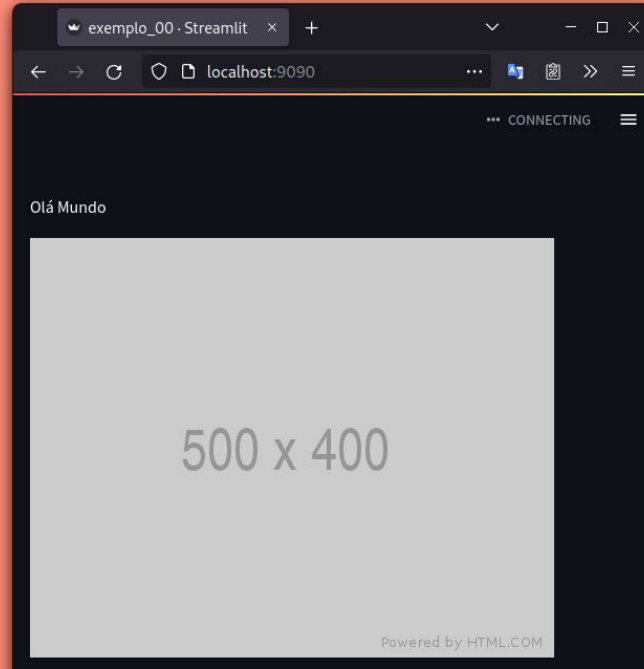
Um exemplo básico



```
1  # exemplo_00.py
2  import streamlit as st
3
4  st.write('Olá Mundo')
5  st.image('https://via.placeholder.com/500x400/')
— □ ×
```

```
— □ ×
streamlit run exemplo_00.py
```

A resposta desse comando



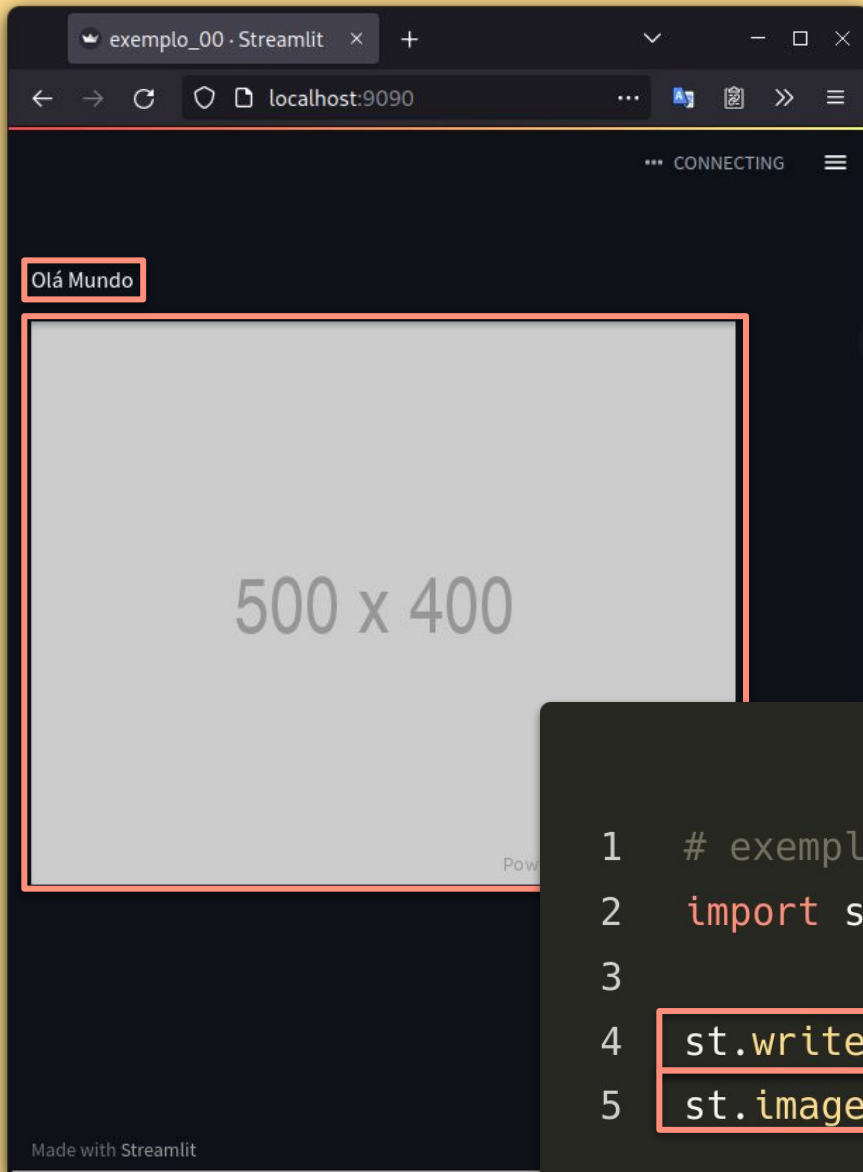
```
|.venv|py-3.10.6 babbage in ~/git/live-beta/live_streamlit  
o → streamlit run slides/exemplo_00.py
```

You can now view your Streamlit app in your browser.

Local URL: <http://localhost:8501>

Network URL: <http://192.168.15.72:8501>

Com isso, vimos dois Widgets



```
1 # exemplo_00.py
2 import streamlit as st
3
4 st.write('Olá Mundo')
5 st.image('https://via.placeholder.com/500x400/')
```

Nosso primeiro APP



Criemos um visualizador de arquivos com 30 linhas.

Desta forma poderemos conhecer as **categorias** de widgets.

- **Texto**
 - markdown
 - code
- **Mídia**
 - audio
 - image
- **Gráfico**
 - Linha
- **Visualização**
 - dataframe
 - json
- **Input**
 - fileuploader
- **status**
 - error

```

1  from json import loads
2  import streamlit as st
3  import pandas as pd
4
5  st.markdown(
6      """
7  # Exibidor de arquivos
8
9  ## Suba um arquivo e vejamos o que acontece :smile:
10 """
11 )
12
13 arquivo = st.file_uploader(
14     'Suba um arquivo', type=['jpg', 'csv', 'json', 'py', 'wav']
15 )
16
17 if arquivo:
18     match arquivo.type.split('/'):
19         case 'image', 'jpeg' | 'png':
20             st.image(arquivo)
21         case 'text', 'csv':
22             df = pd.read_csv(arquivo)
23             st.dataframe(df)
24             st.line_chart(df)
25         case 'application', _:
26             st.json.loads(arquivo.read())
27         case 'text', 'x-python':
28             st.code(arquivo.read().decode())
29         case 'audio', _:
30             st.audio(arquivo)
31     else:
32         st.error('Arquivo ainda não carregado!')

```

Exibidor de arquivos

Suba um arquivo e vejamos o que acontece 😊

Suba um arquivo



Drag and drop file here

Limit 200MB per file • JPG, CSV, JSON, PY, WAV

Browse files



exemplo_01.py 0.8KB



```

from json import loads
import streamlit as st
import pandas as pd

st.markdown(
    """
    # Exibidor de arquivos

    ## Suba um arquivo e vejamos o que acontece :smile:
    """
)

arquivo = st.file_uploader(
    'Suba um arquivo', type=['jpg', 'csv', 'json', 'py', 'wav']
)

if arquivo:
    print(arquivo.type)
    match arquivo.type.split('/'):
        case 'image', 'jpeg' | 'png':
            st.image(arquivo)
        case 'text', 'csv':
            df = pd.read_csv(arquivo)
            st.dataframe(df)
            st.line_chart(df)

```


Conceitos



A execução do streamlit é um **loop de eventos** e toda vez que uma ação é concretizada ele pausa o loop para esperar alguma nova ação.

- Widgets são adicionados a baixo
- Código python controla o fluxo
- **Código python pode ser chamado dentro do app**

Nosso segundo app



Colocando marca d'agua em fotos (de maneira simples) usando pillow.

Vamos aprender a usar:

- Integração de biblioteca externa
- Callbacks
- Outros inputs

```
pip install pillow
```

Nosso segundo app



Colocando marca d'agua em fotos (de maneira simples) usando **pillow**.

Vamos aprender a usar:

- Integração de biblioteca externa
- Callbacks
- Outros inputs

`pip install pillow`

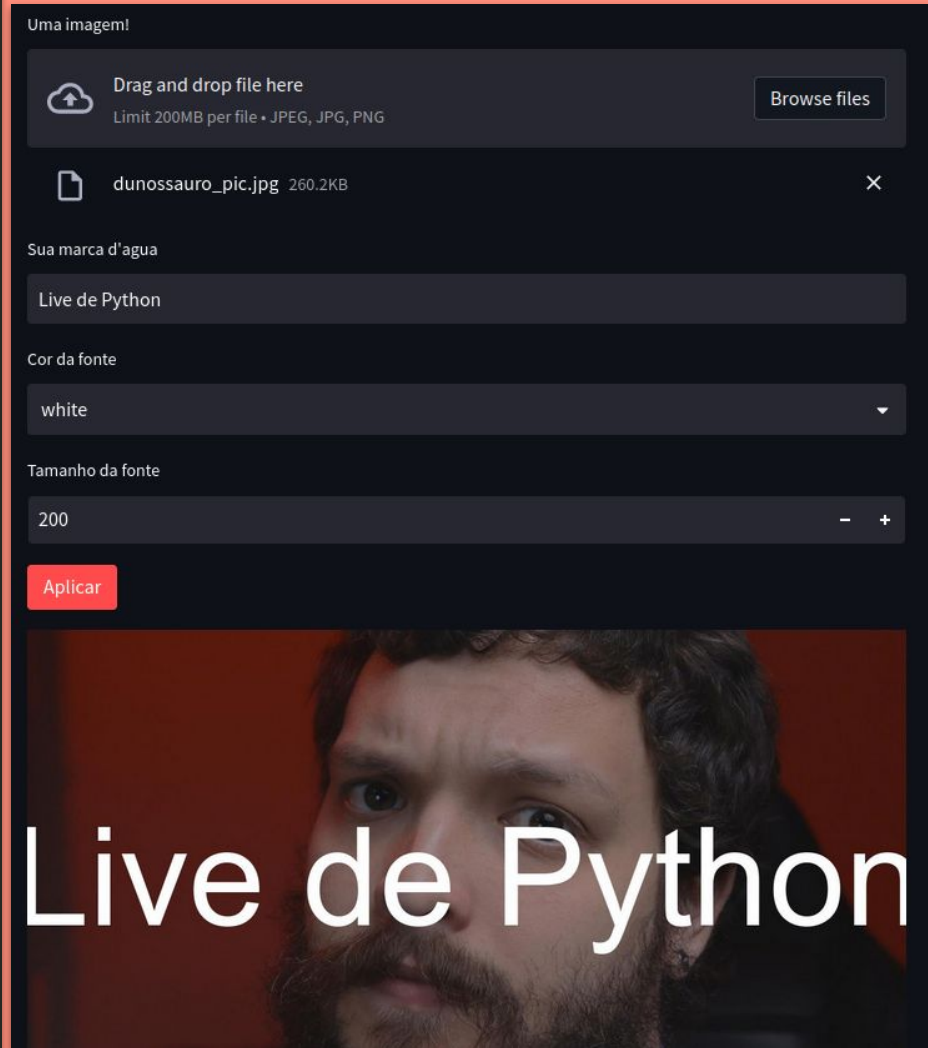


<https://youtu.be/kT85FfTE22E>

```

1 import streamlit as st
2 from PIL import Image, ImageDraw, ImageFont
3
4
5 def text_on_image(image, text, font_size, color):
6     img = Image.open(image)
7     font = ImageFont.truetype('arial.ttf', font_size)
8     draw = ImageDraw.Draw(img)
9     iw, ih = img.size
10    fw, fh = font.getsize(text)
11    draw.text(
12        ((iw - fw) / 2, (ih - fh) / 2),
13        text,
14        fill=color,
15        font=font,
16    )
17    img.save('last_image.jpg')
18
19
20 image = st.file_uploader('Uma imagem!', type=['jpeg', 'jpg', 'png'])
21 text = st.text_input("Sua marca d'agua")
22 color = st.selectbox('Cor da fonte', ['black', 'white', 'red', 'green'])
23 font_size = st.number_input('Tamanho da fonte', min_value=50)
24
25 if image:
26     result = st.button(
27         'Aplicar',
28         type='primary',
29         on_click=text_on_image,
30         args=(image, text, font_size, color)
31     )
32 if result:
33     st.image('last_image.jpg')
34     with open('last_image.jpg', 'rb') as file:
35         st.download_button(
36             'Baixar',
37             file_name='image.jpg',
38             data=file,
39             mime='image/jpg'
40         )

```

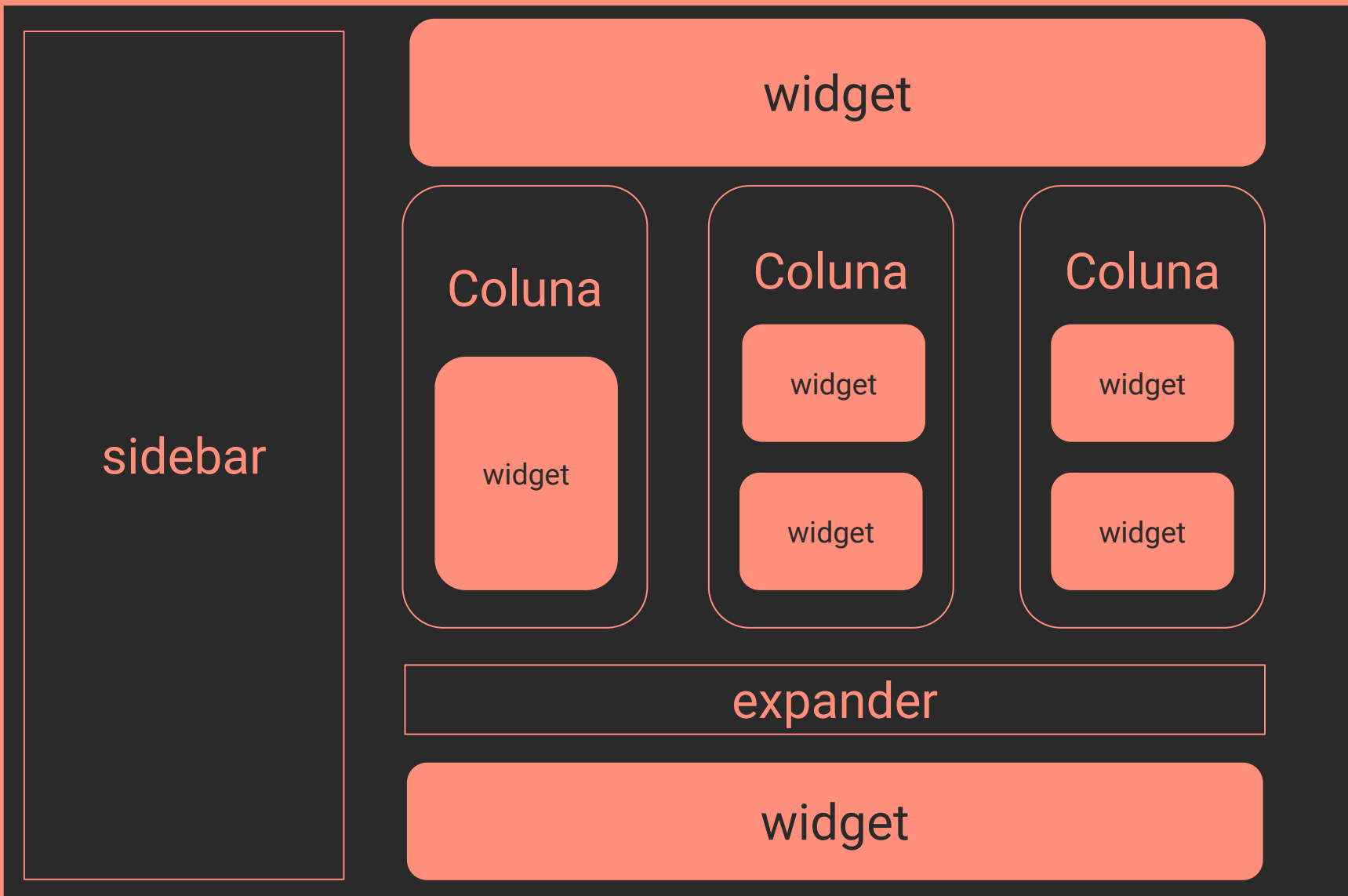


Layouts



Layout são formas de organizar, mostrar/esconder widgets. Temos algumas opções interessantes:

- Colunas
- Expansor
- Abas
- Barra lateral
- etc.



Mais um Appzinho



Agora vamos combater com spacy e aprender os layouts. Vamos precisar de:

- Layouts
 - Colunas
 - Sidebar
 - Expander
- Componentes
 - HTML
- Inputs
 - TextArea

```
pip install spacy
```

```
spacy download pt_core_news_lg
```

Mais um Appzinho

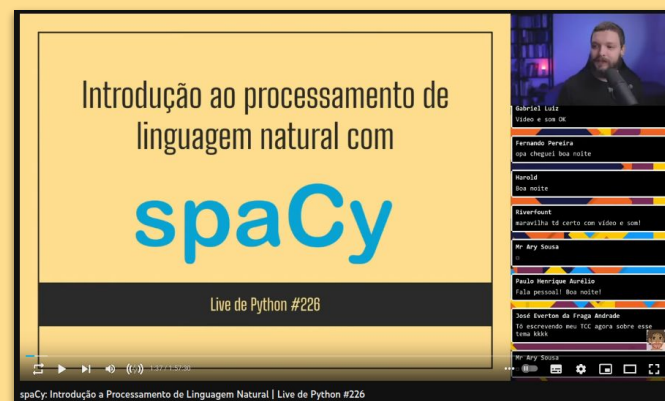


Agora vamos combater com **spacy** e aprender os layouts. Vamos precisar de:

- Layouts
 - Colunas
 - Sidebar
 - Expander
- Componentes
 - HTML
- Inputs
 - TextArea

```
pip install spacy
```

```
spacy download pt_core_news_lg
```



<https://youtu.be/Vr9QXpELdrs>

×

Escolha a categoria:

Gramática

Filtro

PROPN × VERB × NOUN ×

ADP × CONJ × DET ×

Bote um texto aqui!

Eduardo foi a feira.

✓

Dados do Spacy

Token	Tag	Morph
Eduardo	PROPN	Gender=Masc Number=Sing
a	DET	Definite=Def Gender=Fem Number=Sing PronType=Art
feira	NOUN	Gender=Fem Number=Sing

Token	Tag	Morph
Eduardo	PROPN	Gender=Masc Number=Sing
a	DET	Definite=Def Gender=Fem Number=Sing PronType=Art
feira	NOUN	Gender=Fem Number=Sing

Instalando coisas
externas

Plug
ins

Componentes externos



O Streamlit tem diversos componentes externos

<https://streamlit.io/components>

Usemos um bem simples, mas divertido, o ACE e criar um editor de textos

```
pip install streamlit-ace
```

```
1  import streamlit as st
2  from streamlit_ace import st_ace, THEMES, LANGUAGES
3
4  st.title('Nosso Editor de textos')
5
6  c1, c2 = st.columns([3, 1])
7
8  with c1:
9      content = st_ace(
10         theme=c2.selectbox('Tema', options=THEMES, index=0),
11         font_size=c2.slider('Fonte', 5, 24, 14),
12         language=c2.selectbox('Linguagem', options=LANGUAGES, index=121),
13         tab_size=c2.slider('Tab size', 1, 8, 4),
14         min_lines=30,
15     )
16     content
```

Deploy

Colocando uma
aplicação no
streamlit cloud

O que você precisa?



Antes:

- Uma conta no github
- Uma conta no streamlit
- Linkar sua conta do github no streamlit

Deploy:

- Criar um arquivo streamlit
- Criar um requirements.txt
- Colocar no repositório
- Linkar seu repositório no streamlit

Pronto!

Links



- Nosso app inicial: <https://dunossauro-test-streamlit-app-ly9jox.streamlit.app/>
- Gerado na live: <https://dunossauro-live-de-python-streamlit-app-sdi3ei.streamlit.app/>
- Site oficial: <https://streamlit.io/>
- Documentação: <https://docs.streamlit.io>
- API e Widgets: <https://docs.streamlit.io/library/api-reference>
- cheatsheet: <https://docs.streamlit.io/library/cheatsheet>
- Componentes externos: <https://streamlit.io/components>
- Ace: <https://github.com/okld/streamlit-ace>



picpay.me/dunossauro



apoia.se/livedepython



pix.dunossauro@gmail.com



Ajude o projeto <3

