

Extraíndo dados com Parsel



Live de Python # 239



1. Lendo HTML

Um básico necessário para conseguir extrair.

2. O parser


Conhecendo a biblioteca.

3. CSS e Xpath

Extraindo dados usando seletores

4. Pequenos projetinhos

Até dar o tempo

 main

 1 branch


 0 tags


Go to file

Code


About


Scraping de sites de imobiliárias para achar casas que caibam no meu orçamento

 Readme

 GPL-3.0 license

 16 stars










 3 watching

 0 forks

Report repository

Releases

No releases published

	dunossauro Atualização do readme	8c7ff46 on Aug 30, 2022	 22 commits
	utils	Simplificando a forma de executar	10 months ago
	.gitignore	Atualização do gitignore	10 months ago
	LICENSE	Initial commit	10 months ago
	Makefile	Simplificando a forma de executar	10 months ago
	README.md	Atualização do readme	10 months ago
	imobiliarias_example....	Exemplo de yaml das imobiliarias	10 months ago
	poetry.lock	adicionando isort e loguru no projet	10 months ago
	pyproject.toml	adicionando isort e loguru no projet	10 months ago
	run.py	Simplificando a forma de executar	10 months ago

<https://github.com/dunossauro/mudanca>

COM:
RENNE
ROCHA



CONHECENDO XPATH

0:00 / 1:45:25



Selenium com Python #palestra - Conhecendo XPATH com Renne Rocha

<https://www.youtube.com/watch?v=vuLNc2yCNY>



picpay.me/dunossauro



apoia.se/livedepython



pix.dunossauro@gmail.com



Ajude o projeto <3



Ademar Peixoto, Adilson Herculano, Adriano Ferraz, Alemão, Alexandre Harano, Alexandre Lima, Alexandre Takahashi, Alexandre Villares, Alex Lima, Alynne Ferreira, Alysson Oliveira, Ana Carneiro, Andre Azevedo, Andre Mesquita, Aquiles Coutinho, Arnaldo Turque, Aslay Clevisson, Aurelio Costa, Bernardo At, Bernardo Fontes, Bruno Almeida, Bruno Barcellos, Bruno Barros, Bruno Freitas, Bruno Lopes, Bruno Ramos, Caio Nascimento, Christiano Moraes, Damianth, Daniel Freitas, Daniel Wojcickoski, Danilo Boas, Danilo Segura, Danilo Silva, David Couto, David Kwast, Davi Govinho, Davi Souza, Delton Porfiro, Denis Bernardo, Diego Farias, Diego Guimarães, Dilenon Delfino, Diogo Paschoal, Diogo Silva, Edgar, Eduardo Silveira, Eduardo Tolmasquim, Elias Silva, Emerson Rafael, Eneas Teles, Erick Andrade, Érico Andrei, Everton Silva, Fabiano Tomita, Fabio Barros, Fábio Barros, Fabio Castro, Fábio Thomaz, Fabricio Patrocinio, Felipe Rodrigues, Fernanda Prado, Fernando Celmer, Firehouse, Flávio Meira, Francisco Neto, Francisco Silvério, Gabriel Espindola, Gabriel Mizuno, Gabriel Paiva, Gabriel Simonetto, Geandreson Costa, Geizelder, Gilberto Abrao, Giovanna Teodoro, Giuliano Silva, Guilherme Felitti, Guilherme Gall, Guilherme Silva, Guionardo Furlan, Gustavo Pereira, Gustavo Suto, Harold Gautschi, Heitor Fernandes, Helvio Rezende, Hugo Cosme, Igor Riegel, Italo Silva, Janael Pinheiro, Jean Victor, Joelson Sartori, Jônatas Oliveira, Jônatas Silva, Jon Cardoso, Jorge Silva, José Gomes, Joseíto Júnior, Jose Mazolini, Juan Felipe, Juan Gutierrez, Juliana Machado, Julio Franco, Júlio Gazeta, Julio Silva, Kaio Peixoto, Kálita Lima, Kaneson Alves, Leandro Miranda, Leandro Silva, Leo Ivan, Leonardo Mello, Leonardo Nazareth, Leon Solon, Luancomputacao Roger, Lucas Adorno, Lucas Carderelli, Lucas Mendes, Lucas Nascimento, Lucas Schneider, Lucas Simon, Lucas Valino, Luciano Filho, Luciano Ratamero, Luciano Teixeira, Luis Alves, Luis Eduardo, Luiz Duarte, Luiz Lima, Luiz Paula, Luiz Perciliano, Mackilem Laan, Marcelo Campos, Marcio Moises, Marco Mello, Marcos Gomes, Maria Clara, Marina Passos, Mateus Lisboa, Mateus Ribeiro, Mateus Silva, Matheus Silva, Matheus Vian, Mauricio Nunes, Mírian Batista, Mlevi Lsantos, Murilo Viana, Nathan Branco, Nicolas Teodosio, Otávio Carneiro, Patricia Minamizawa, Patrick Felipe, Paulo Tadei, Pedro Henrique, Pedro Pereira, Peterson Santos, Priscila Santos, Pydocs Pro, Pytonyc, Rafael Lopes, Rafael Romão, Rafael Veloso, Raimundo Ramos, Ramayana Menezes, Regis Santos, Renato Oliveira, Rene Bastos, Ricardo Silva, Riverfount, Rjribeiro, Robson Maciel, Rodrigo Barretos, Rodrigo Freire, Rodrigo Oliveira, Rodrigo Quiles, Rodrigo Ribeiro, Rodrigo Vaccari, Rodrigo Vieira, Rogério Nogueira, Ronaldo Silveira, Rui Jr, Samanta Cicilia, Téó Calvo, Thaynara Pinto, Thiago Araujo, Thiago Borges, Thiago Curvelo, Thiago Souza, Tiago Minuzzi, Tony Dias, Tyrone Damasceno, Uadson Emile, Valcilon Silva, Valdir Tegen, Vcwild, Vicente Marcal, Vinicius Stein, Vladimir Lemos, Walter Reis, William Vitorino, Willian Lopes, Wilson Duarte, Wilson Neto, Zeca Figueiredo



Obrigado você



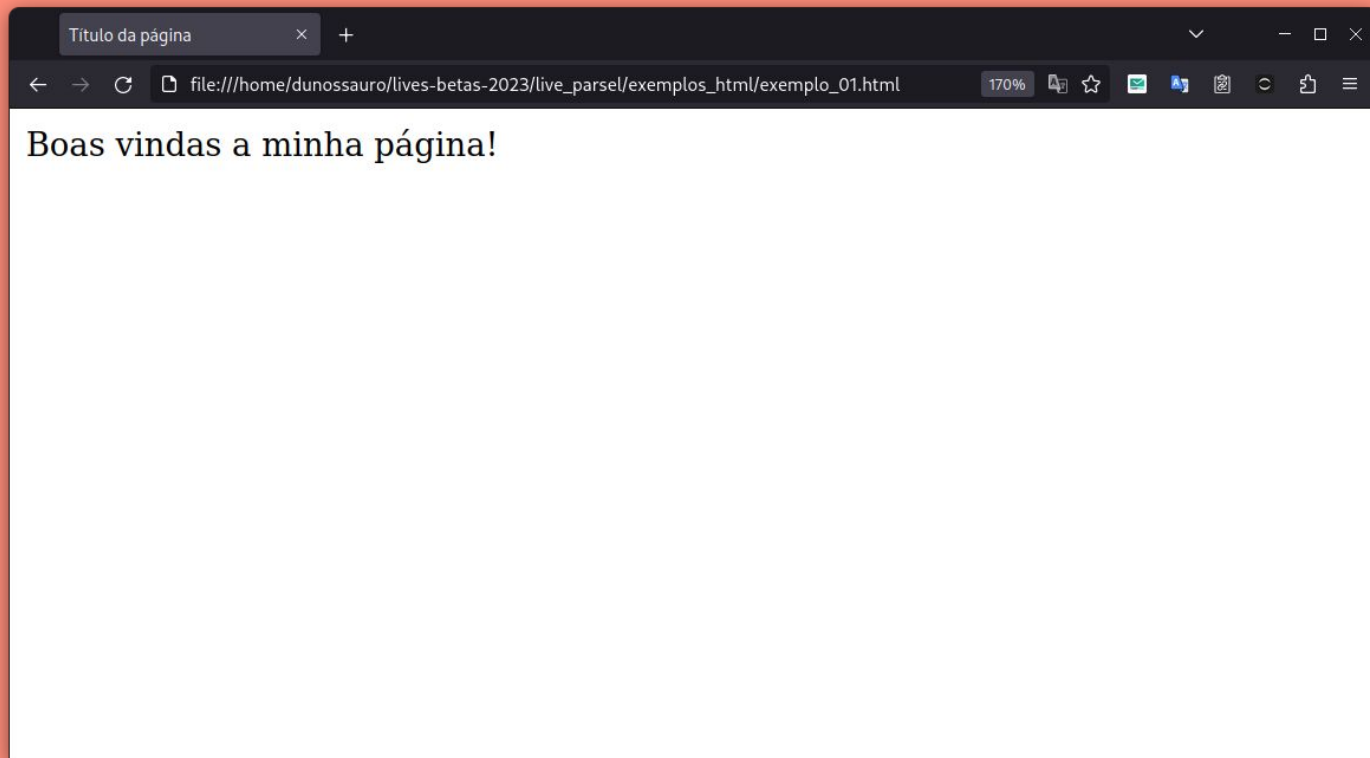
Um básico
necessário

HTML

HTML



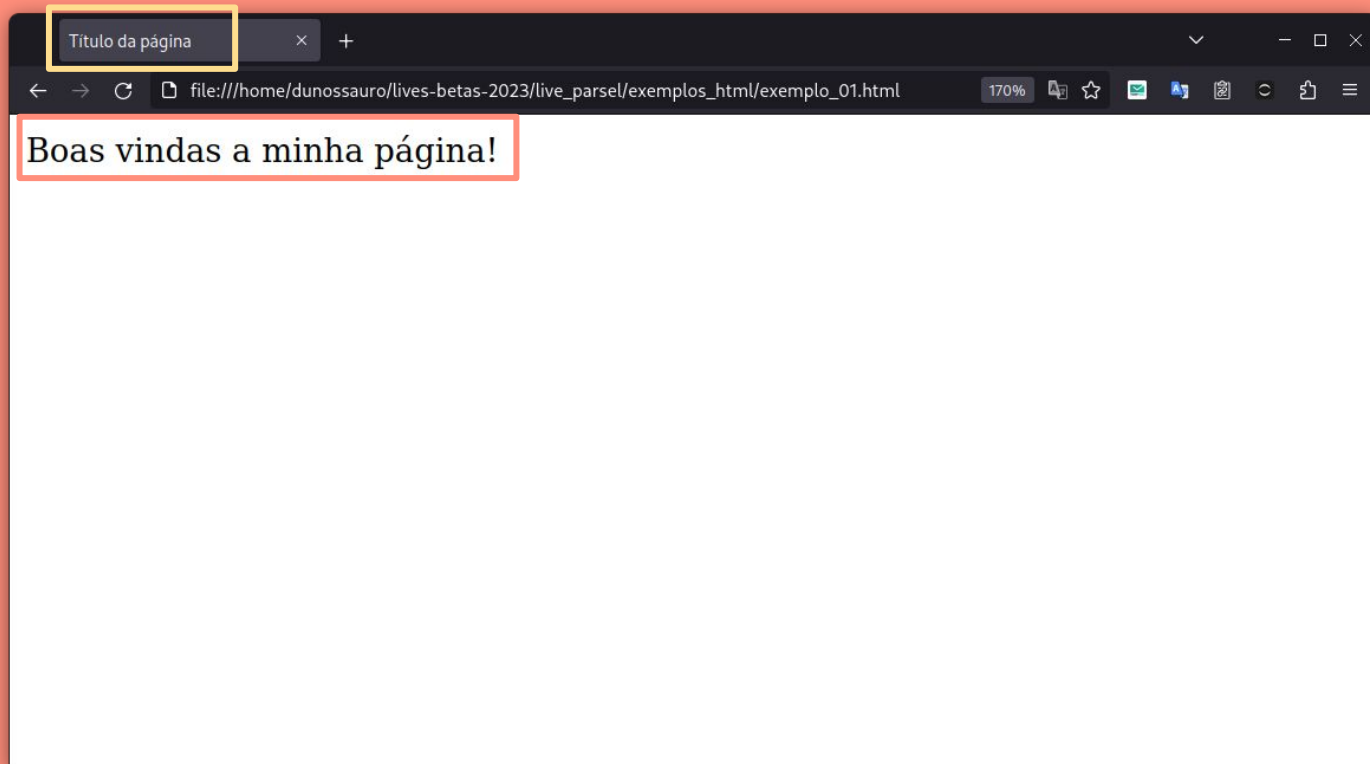
HTML é uma **linguagem de marcação** de **hipertexto**. Basicamente é uma estrutura composta por marcações [ou elementos [, ou tags]] e atributos.



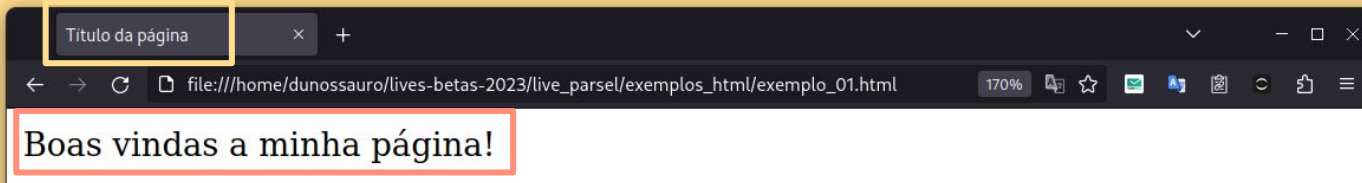
HTML



HTML é uma **linguagem de marcação** de **hipertexto**. Basicamente é uma estrutura composta por marcações [ou elementos [, ou tags]] e atributos.

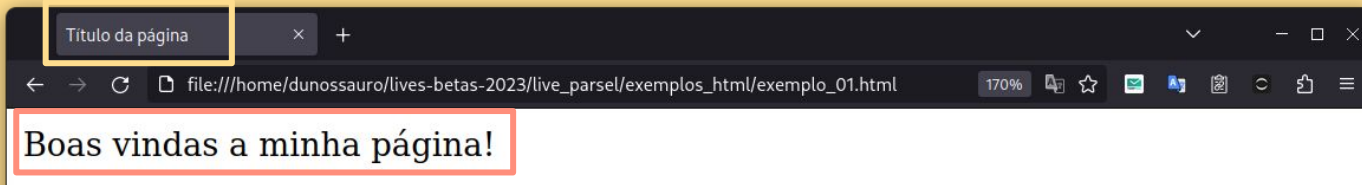


O código da página



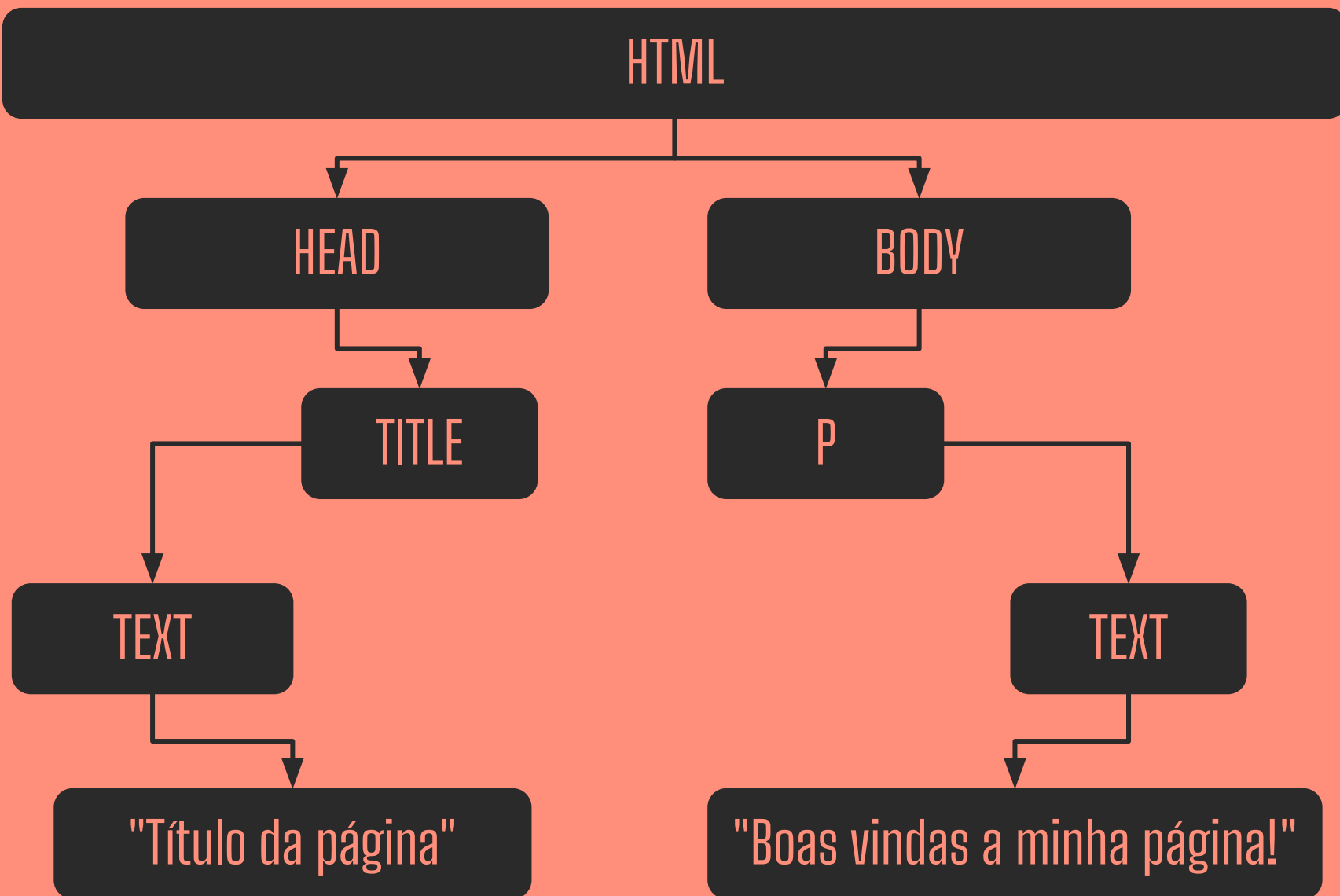
```
1  <html>
2    <head>
3      <title>Título da página</title>
4    </head>
5    <body>
6      <p>Boas vindas a minha página!</p>
7    </body>
8  </html>
```

O código da página

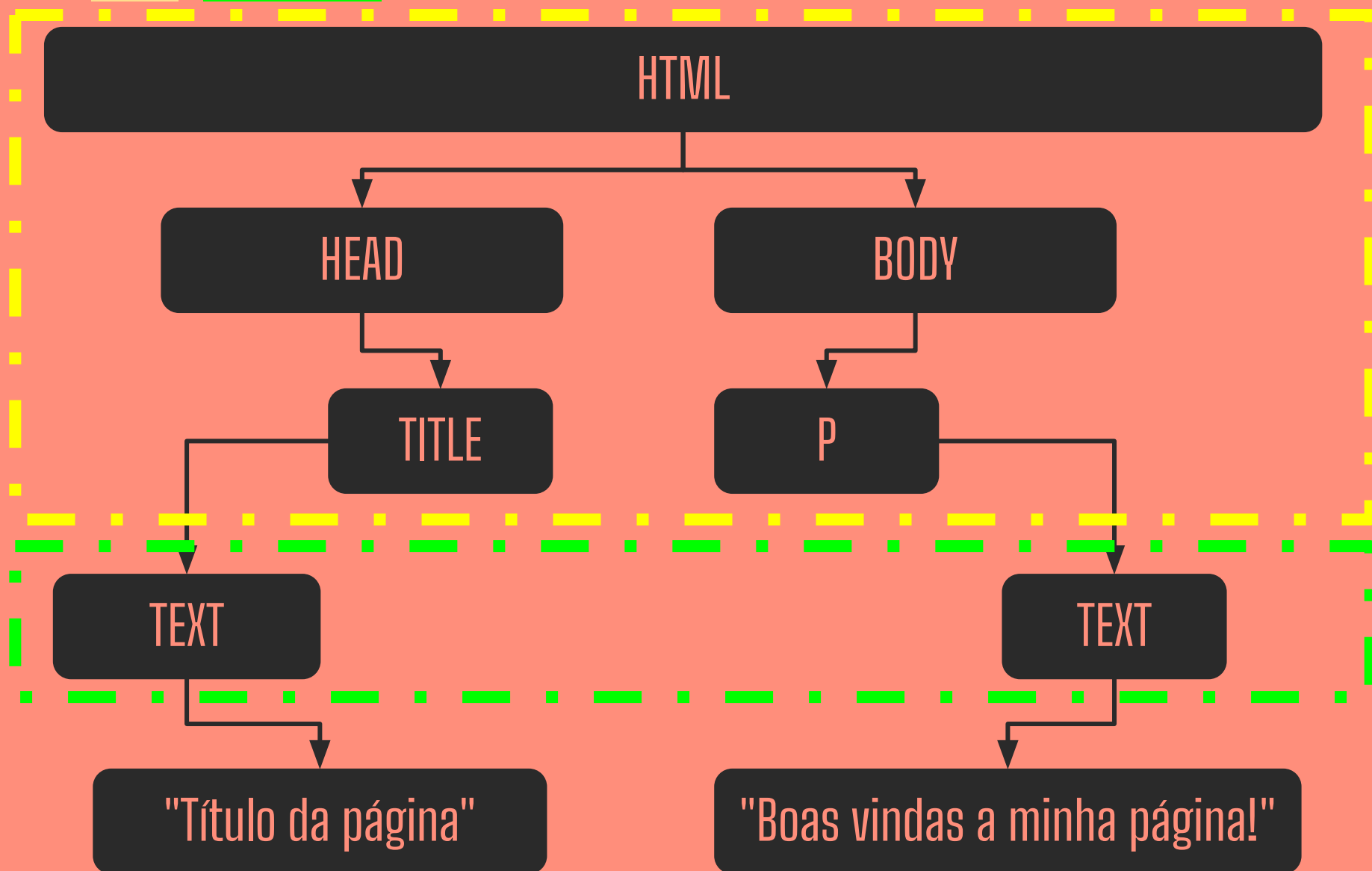


```
1  <html>
2    <head>
3      <title>Título da página</title>
4    </head>
5    <body>
6      <p>Boas vindas a minha página!</p>
7    </body>
8  </html>
```

Uma espécie de árvore (DOM)



Tags, Atributos e valores



Uma declaração um pouco mais complexa




Eduardo Mendes


Pythonista / Youtuber / Músico

"Apenas um rapaz latino americano"


Social

 Youtube

 Twitter

 Mastodon

 Twitch

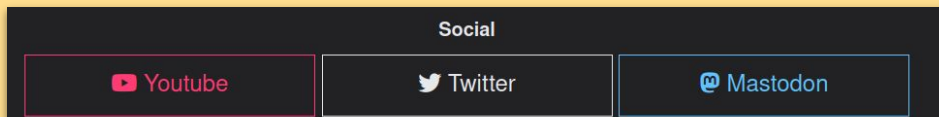
 Instagram

 TikTok

Apoie o meu trabalho

<https://dunossauro.com/>

Uma declaração um pouco mais complexa



```
1  <div class="row">
2    <h5>Social</h5>
3
4    <button class="btn btn-error btn-ghost" onclick="window.open('...')">
5      <div class="button">
6        <i class="fa-brands fa-youtube"></i>
7        Youtube
8      </div>
9    </button>
10
11   <button class="btn btn-default btn-ghost" onclick="window.open('...')">
12     <div class="button">
13       <i class="fa-brands fa-twitter"></i>
14       Twitter
15     </div>
16   </button>
17
18   <button class="btn btn-primary btn-ghost" onclick="window.open('...')">
19     <div class="button">
20       <i class="fa-brands fa-mastodon"></i>
21       Mastodon
22     </div>
23   </button>
24 </div>
```

```

4     <button class="btn btn-error btn-ghost" onclick="window.open('...')">
5         <div class="button">
6             <i class="fa-brands fa-youtube"></i>
7             Youtube
8         </div>
9     </button>

```

TAG	button
class	btn, btn-error, btn-ghost

TAG	div
class	button

TAG	i
class	fa-brands fa-youtube

Tags, atributos e identificadores



Existem uma infinidade de Elementos [, ou tags]. Cada tag tem seus atributos específicos. Mas também existem tags globais que podem ser usados para todos os elementos:

- **Exemplo de tags:** div, h1, html, form, aside, a, nav, p, etc.
- **Atributos globais:** id, class, hidden, style, lang, etc.

<https://developer.mozilla.org/en-US/docs/Web/HTML/Element>
<https://developer.mozilla.org/en-US/docs/Web/HTML/Attributes>

Mais um exemplo



Python

Criada em 1991

Haskell

Criada em 1990

Lisp

Criada em 1958

Prolog

Criada em 1972

```
<body>
  <div id="python">
    <h2>Python</h2>
    <p>Criada em 1991</p>
  </div>
```

```

  <div id="haskell">
    <h2>Haskell</h2>
    <p>Criada em 1990</p>
  </div>
```

```

  <div id="lisp">
    <h2>Lisp</h2>
    <p>Criada em 1958</p>
  </div>
```

```

  <div id="prolog">
    <h2>Prolog</h2>
    <p>Criada em 1972</p>
  </div>
```

```
</body>
```

Parsel

Uma introdução



O Parsel nasceu dentro do scrapy, um framework completo para raspagem de dados.

- Mantido pelo time do Scrapy (primeira release em 2010)
- Separação inicial do código aconteceu em 2015
- Atualmente na release **1.8.1**
- Assim como o **bs4**, ele também usa o **lxml** como base

A carinha do parsel [exemplo_00.py]



```
from parsel import Selector

html = """
<html>
  <head>
    <title>Título da página</title>
  </head>
  <body>
    <p>Boas vindas a minha página!</p>
  </body>
</html>
"""

sel = Selector(text=html)
```

```
pip install parse
```



Instalação



Os seletores e a extração



Para buscar por dados no texto, temos basicamente dois métodos para o seletor:

- `selector.css()`: Busca usando seletores css
- `selector.xpath()`: Busca usando xpath

Para extrair os dados, podemos usar os métodos:

- `selector.get()`: Retorna um texto
- `selector.getall()`: Retorna uma lista de textos
- `selector.attrib`: Retorna um dicionário com os atributos do elemento

Um uso básico



```
from parsel import Selector

html = """
<html>
  <head>
    <title>Título da página</title>
  </head>
  <body>
    <p>Boas vindas a minha página!</p>
  </body>
</html>
"""

sel = Selector(text=html)
elementos = sel.css('*') # todas as tags
type(elementos) # SelectorList
elementos.getall() # todas as tags
```


A interface do parsel é uma forma de builder



Podemos ir afinilando as buscas dentro de outras buscas.

```
from parsel import Selector
sel = Selector(text=html)
sel.css('*').css('body').css('p').getall()
```

SelectorList



O SelectorList facilita para iterar também. Você pode pegar os valores de forma posicional. É bastante interessante para debug

```
from parsel import Selector
sel = Selector(text=html)
sel.css('*')[0].get()
```

Seleto res

CSS / XPATH

Seletores CSS

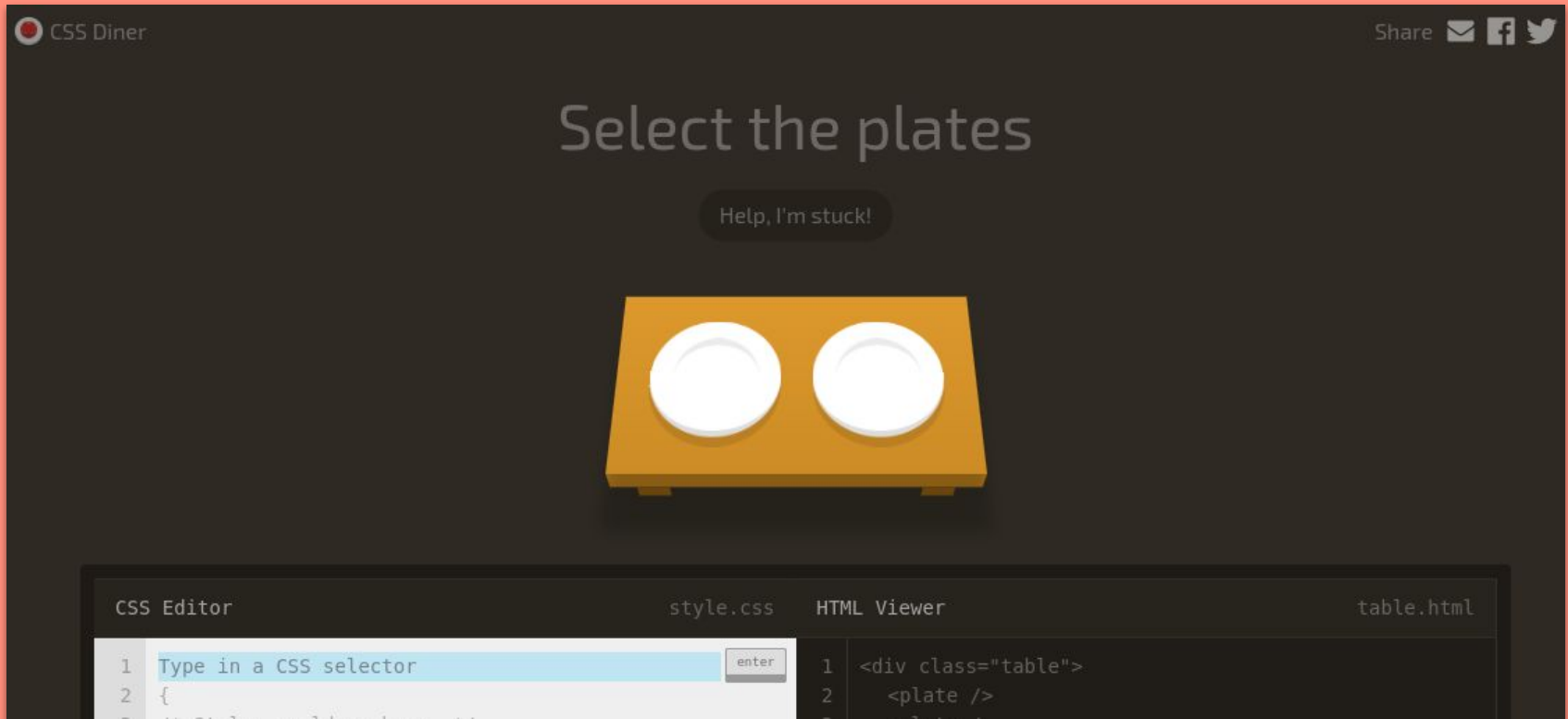


Seletores CSS são formas de encontrar elementos na árvore usando a sintaxe do CSS. Os elementos podem ser encontrados por:

- **tag:** a, body, form, button, etc.
- **id:** #identificador
- **classe:** .nome-da-classe, .nome-da-classe.nome-de-outra-classe
- **atributo:** [atributo], [atributo=valor], [atributo~=valor]

https://developer.mozilla.org/pt-BR/docs/Web/CSS/CSS_Selectors

Para quem quer aprender seletores CSS



<https://flukeout.github.io/>

Pegaremos os títulos e links de todas
as lives de python usando css
selector

<https://www.youtube.com/@Dunossauro/streams>



Primeiro projetinho!



Seletores XPATH



Seletores XPATH são formas de encontrar elementos na árvore usando a sintaxe do XPATH. Os elementos podem ser encontrados por:

- **Tag:** //h1, //div/p, /body
- **Atributos:**
 - **id:** //*[@id="id"]
 - **classe:** //*[@class="class"]
 - **Atributos gerais:** //input[@type="submit"]

Um lugar legal para aprender como fazer



Xpath cheatsheet

Xpath test bed

Test queries in the Xpath test bed:

Xpath test bed
(whitebeam.org)



Browser console

```
$x("//div")
```

Works in Firefox and Chrome.

Selectors

Descendant selectors

h1	//h1	?
div p	//div//p	?
ul > li	//ul/li	?
ul > li > a	//ul/li/a	

Attribute selectors

#id	//*[@id="id"]	?
.class	//*[@class="class"] ...kinda	
input[type="submit"]	//input[@type="submit"]	
a#abc[for="xyz"]	//a[@id="abc"][@for="xyz"]	?

<https://devhints.io/xpath>

Agora vamos raspar as respostas do
meu nick no duckduckgo usando
xpath

<https://html.duckduckgo.com/>



Segundo projetinho!





picpay.me/dunossauro



apoia.se/livedepython



pix.dunossauro@gmail.com



Ajude o projeto <3

