

# Guia rápido sobre Coleta de Dados na Web Python



```
[49] print('Ola Mundão!')
```

Ola Mundão!

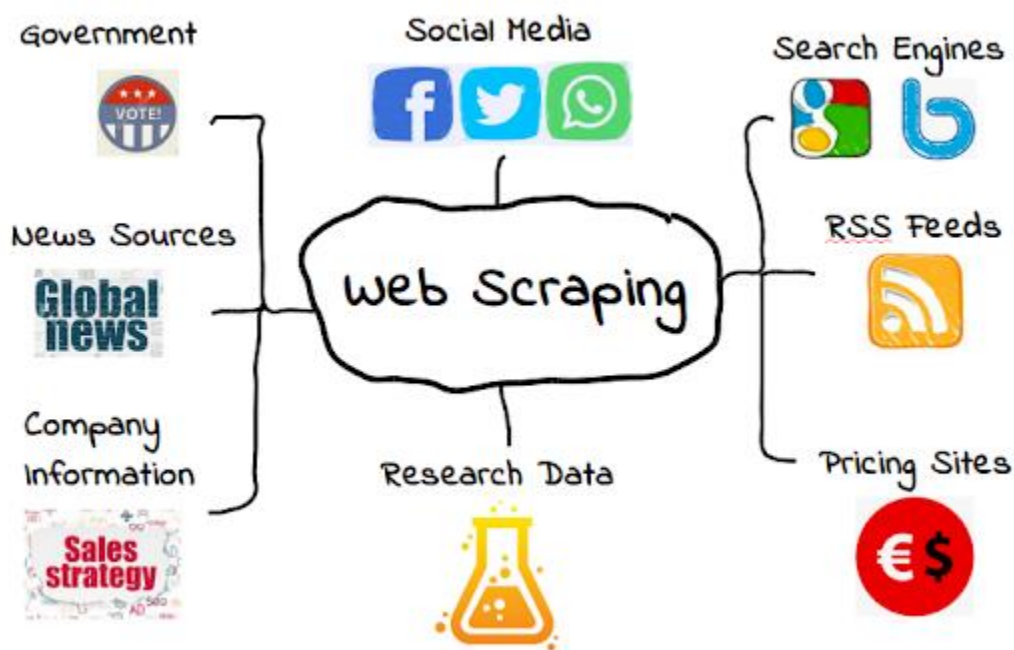
# Guia rápido sobre Coleta de Dados na Web

## O que é Web Scraping [ Coleta de Dados na Web ] ?

O **Web Scraping** permite a coleta de dados em sites específicos e pode gerar insights valiosos para o seu negócio.

Podemos gerar diversos benefícios com o Web Scraping:

- ✓ Gerar ideias valiosas de negócio;
- ✓ Economizar tempo e esforço;
- ✓ Coletar dados de fontes mais precisas;
- ✓ E outros.



## O que é o BeautifulSoup?

O **Beautiful Soup** Biblioteca Python projetada para facilitar a extração de dados nos documentos html e xml..

benefícios do Framework:




- ✓ Fornece alguns métodos simples para navegar, pesquisar e modificar uma árvore de análise;
- ✓ Bastante robusta para trabalhar com html/xml mal formatados;
- ✓ E outros.



Nesse exemplo vamos usar um site como referência do Wikipédia.

[https://pt.wikipedia.org/wiki/Lista\\_de\\_munic%C3%ADpios\\_de\\_Santa\\_Catarina](https://pt.wikipedia.org/wiki/Lista_de_munic%C3%ADpios_de_Santa_Catarina)

Municípios

#	Município	Código BGE	Localização
1	Abdon Batista	4200051	
2	Abelardo Luz	4200101	
3	Agrolândia	4200200	

Vamos fazer scraping dessa informação.

Vamos pegar essa tabela dos municípios e transformar em uma base de dados.

Vamos importar as bibliotecas externas que precisamos

```
[1] # ---- Importar as Libs necessarias

# Libs para web Scraping

# Request é uma lib usada para request https
import requests

# Soup é a lib usada para scraping
from bs4 import BeautifulSoup

# Lib para modelagem de Dados
import pandas as pd
```

## Vamos importar as bibliotecas externas que precisamos

```
# Carregando a pagina

# Salvar o link da pagina
Site = 'https://pt.wikipedia.org/wiki/Lista_de_munic%C3%ADpios_de_Santa_Catarina'

# Fazendo o carregando da pagina atraves do Request
Pagina = requests.get(Site)
```

## Transpor as informações do Request para o Soup

```
[3] # Coletando as infos do request e passar para o Soup os dados
Coleta = BeautifulSoup(Pagina.text, 'html.parser')
Coleta
```

```
<!DOCTYPE html>

<html class="client-nojs" dir="ltr" lang="pt">
<head>
<meta charset="utf-8"/>
<title>Lista de municípios de Santa Catarina - Wikipédia, a enciclo
<script>document.documentElement.className="client-js";RLCONF={"wgB
"wgRelevantPageIsProbablyEditable":!0,"wgRestrictionEdit":[],"wgRes
"skins.vector.styles":"ready","skins.vector.icons":"ready","mediawi
"ext.navigationTiming","ext.uls.compactlinks","ext.uls.interface","
<script>(RLQ=window.RLQ||[]).push(function(){mw.loader.implement("u
});});</script>
<link href="/w/load.php?lang=pt&modules=ext.cite.styles%7Cext.d
<script async="" src="/w/load.php?lang=pt&modules=startup&o
<meta content="" name="ResourceLoaderDynamicStyles"/>
<link href="/w/load.php?lang=pt&modules=site.styles&only=st
<meta content="MediaWiki 1.37.0-wmf.6" name="generator"/>
<meta content="origin" name="referrer"/>
```

→ Coletamos toda a pagina

## Extraindo o titulo da pagina

```
[4] # Coletando o titulo da Pagina
print( Coleta.title )

# Coletando apenas o texto
print( Coleta.title.string )

# Extrair o nome da Tag
print( Coleta.title.name )
```

```
<title>Lista de municípios de Santa Catarina - Wikipédia, a enciclo
Lista de municípios de Santa Catarina - Wikipédia, a enciclopédia l
title
```

## Podemos buscar uma tag especifica de acordo com o nome dela

```
[5] # Buscando uma classe pelo nome dela
Busca = Coleta.find(class_='p-search--show-thumbnail')
print(Busca)

<div class="p-search--show-thumbnail" id="p-search" role="search">
<h3>
<label for="searchInput">Busca</label>
</h3>
<form action="/w/index.php" id="searchform">
<div data-search-loc="header-moved" id="simpleSearch">
<input accesskey="f" autocapitalize="sentences" id="searchInput" na
<input name="title" type="hidden" value="Especial:Pesquisar"/>
<input class="searchButton mw-fallbackSearchButton" id="mw-searchBu
<input class="searchButton" id="searchButton" name="go" title="Ir p
</input></div>
</form>
</div>
```

## Podemos buscar todas as tags da pagina

```
[16] # Retornar todas as tags da Pagina com um parametro
print( Coleta.p )
print( Coleta.div )
#print( Coleta.a )

<p>Os <b>municípios de Santa Catarina</b> são as subdivisões políti
</p>
<div class="mw-page-container">
<a class="mw-jump-link" href="#content">Saltar para o conteúdo</a>
<div class="mw-page-container-inner">
<input checked="" class="mw-checkbox-hack-checkbox" id="mw-sidebar-
<header class="mw-header">
<label aria-controls="mw-panel" class="mw-checkbox-hack-button mw-u
Alternar barra lateral
```

## Identificando a nomenclatura de uma Tag

```
[7] # Identificar o nome da classe de uma tag
print( Coleta.a['class'] )
print( Coleta.span['class'] )
```

```
['mw-jump-link']
['mw-logo-container']
```

## Identificar todas as Tag de acordo com um parâmetro



```
# Identificar todas as Tags com um parametro
Coleta.find_all('a')
```

```
[<a class="mw-jump-link" href="#content">Saltar para o conteúdo</a>
<a class="mw-logo" href="/wiki/Wikip%C3%A9dia:P%C3%A1gina_principal"
<img alt="" aria-hidden="true" class="mw-logo-icon" height="50" s
<span class="mw-logo-container">

</a>,
<a accesskey="n" href="/wiki/Especial:Minha_discuss%C3%A3o" title:
<a accesskey="y" href="/wiki/Especial:Minhas_contribui%C3%A7%C3%B!
<a href="/w/index.php?title=Especial:Criar_conta&returnto=Lis
<a accesskey="o" href="/w/index.php?title=Especial:Entrar&reti
<a accesskey="z" href="/wiki/Wikip%C3%A9dia:P%C3%A1gina_principal
<a href="/wiki/Portal:Conte%C3%BAdo_destacado">Conteúdo destacado
<a href="/wiki/Portal:Eventos_atuais" title="Informação temática :
<a href="/wiki/Wikip%C3%A9dia:Esplanada">Esplanada</a>,
<a accesskey="x" href="/wiki/Especial:Alcatraz" title="Carro
<a href="/wiki/Portal:%C3%8Dndice">Portais</a>,
<a href="/wiki/Wikip%C3%A9dia:Informe_um_erro">Informar um erro</.
```

## Procurar um Id pelo seu nome

```
[9] # Procurar um id específico da pagina
Coleta.find(id='p-search')
```

```
<div class="p-search--show-thumbnail" id="p-search" role="search">
<h3>
<label for="searchInput">Busca</label>
</h3>
<form action="/w/index.php" id="searchform">
<div data-search-loc="header-moved" id="simpleSearch">
<input accesskey="f" autocapitalize="sentences" id="searchInput" name
<input name="title" type="hidden" value="Especial:Pesquisar"/>
<input class="searchButton mw-fallbackSearchButton" id="mw-searchButt
<input class="searchButton" id="searchButton" name="go" title="Ir par
</input></div>
</form>
</div>
```

## Percorrer uma consulta e extrair um informação em específico



```
# Percorrer em um loop para coletar os links da pagina
for link in Coleta.find_all('a'):
    print( link.get('href') )
```

```
#content
/wiki/Wikip%C3%A9dia:P%C3%A1gina_principal
/wiki/Especial:Minha_discuss%C3%A3o
/wiki/Especial:Minhas_contribui%C3%A7%C3%B5es
/w/index.php?title=Especial:Criar_conta&returnto=Lista+de+munic%C3%A
/w/index.php?title=Especial:Entrar&returnto=Lista+de+munic%C3%ADpios
/wiki/Wikip%C3%A9dia:P%C3%A1gina_principal
/wiki/Portal:Conte%C3%BAdo_destacado
```

Retornara apenas os textos da pagina

# Coletar apenas o que for texto da Pagina

print( Coleta.get\_text() )

Os municípios de Santa Catarina são as subdivisões políticas do est

A primeira subdivisão criada na então Capitania de Santa Catarina f

O maior município em área é Lages, que cobre uma área de mais de 2

Índice:    ▲   ·   A B C D E F G H I J K L M N O P Q R S T U V W X Y

Vamos para o Case

Vamos verificar em qual tag esta nossos dados

Índice:    ▲   ·   A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Municípios

		Código IBGE	Localização
1	Abdon Batista	4200051	

Elements

<table id="toc" class="noprint">

uto; padding:4px 10px; border:1px

round:#f9f9f9;">...</table>

<h2>...</h2>

<table class="wikitable sortable

n" style="text-align:center;">

<thead>...</thead>

<tbody>

<tr id="A">

<td>1</td>

<td style="text-align:left

<a href="/wiki/Abdon\_Bat

na)" title>Abdon Batista

</td>

Os dados estão em uma **tabela** em formato HTML.

Pela ordem da página os dados estão na seguinte tags :

- 1. Table
- 2. Thead
- 3. Tbody
- 4. Tr
- 5. Td
- 6. a

A nossa informação estão na Tag 'a'

Então vamos minerar essas infos e buscar a informação na Tag a

## Vamos identificar onde esta os dados no site

```
[13] # Coletando todas as Tbody e transformando em uma lista
      Tabelas = list( Coleta.find_all('tbody') )
```

## Criando o script para coletar essa info

```
[14] # Listas para auxiliar na construção da Base
      Cidades = []
      Estado = []
      Id = []

      # Variaveil para ser usada no 'For'
      Loop = 0

      # Loop nas Tags 'a'
      for Text in Tabelas[1].find_all('a'):

          # Extraindo apenas o texto da Tag
          Cidade = Text.string

          # Caso o valor venha vazio será ignorado
          if Cidade == None:
              pass

          # Se o valor for uma cidade, entra nosso 'else'
          else:
              # Salvando os dados nas listas
              Cidades.append( Cidade )
              Estado.append( 'Santa Catarina' )
              Id.append( Loop )

              # Somando o Loop para virar o ID na base de dados
              Loop += 1

      # Criando um Dicionario para estruturar os dados
      Dicionario = {
          'Id' : Id,
          'Cidade' : Cidades,
          'Estado' : Estado
      }

      # Passando o Dicionario como parametro no Pandas
      Base_Cidades = pd.DataFrame( Dicionario, )
```

```
[15] # Verificando nossa nova base de dados
      Base_Cidades
```

	Id	Cidade	Estado
0	0	Município	Santa Catarina
1	1	Abdon Batista	Santa Catarina

Todas as cidades agora  
em uma base de dados



## Final

Esse guia rápido é para ter conhecimentos prévios sobre como utilizar a biblioteca **Beautiful Soup** para processos de Web Scraping.

Caso queira mais informações, acesse a documentação oficial do framework.

Guia da documentação caso queira mais detalhes

<https://www.crummy.com/software/BeautifulSoup/bs4/doc/>



**Odemir Depieri Jr**

Software Engineer Sr  
Tech Lead  
Specialization AI