INSTITUTO FEDERAL
DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
Brasília
Campus Brasília

TECNOLOGIA EM SISTEMAS PARA INTERNET

Lucas Mateus Silva
Emanuelle Ferreira da Silva
Neander Wendel Nobre Teixeira

RELATÓRIO DE PRÁTICA INTEGRADA DE CIÊNCIA DE DADOS E INTELIGÊNCIA ARTIFICIAL

Brasília - DF

19/09/2020

Sumário

1. Objetivos

Esse relatório é tem como objetivo descrever os objetivos principais.

- Desenvolver um script Python para executar a coleta de dados.
- Gerar um arquivo CSV
- Realizar a armazenagem dos dados

2. Descrição do problema

- Foi encontrando ao longo do desenvolvimento do projeto problemas como ao ter acesso às paginas disponibilizadas no canvas, são gerada varias paginas com uma quantidade enorme de dados e com um link para cada uma delas, dificultando a coleta dos dados por serem realizada uma de cada vez.
- A utilização da biblioteca possibilitou salvar, organizar, executar os códigos. Contudo ao salvar os dados com a extensão CSV o texto se torna puro ao ter acesso ao código.
- Em relação ao acesso do código fonte (HTML) da pagina, os dados se encontravam dentro da tags (a) e (TD) para a extração dos dados em que foi preciso coletar informações do texto e os links das tags descritas, dificultando a compreensão e a organização do código.

3. Desenvolvimento

- Na primeira etapa do projeto utilizou as bibliotecas sugeridas e disponibilizadas no canvas como a biblioteca request que foi utilizada a para fazer a requisição do site por meio para pegar o conteúdo por meio do get, a biblioteca Pandas para salva, organizar e executas código alem de salvar na extensão csv, a biblioteca beautifulsoup para fazer a coleta dos dados. A linguagem de programação escolhida foi o Python, pois apresenta ser a linguagem mais indicada para desenvolver projetos em relação a analise, a coleta, a exploração, a organização e execução de dados.

3.1 Código implementado

```
'''' importando as bibliotecas '''

import requests
from bs4 import BeautifulSoup
import pandas as pd

'''acessado o site '''
url = "http://www.nuforc.org/"

'''criando um objeto chamado Page e fazendo uma requisição na pagina e
pegando a url por meio do get '''
page = requests.get(url)
page.encoding = 'utf-8'
'''Busca o conteúdo da página, criando um objeto chamado soup'''
soup = BeautifulSoup(page.text, 'html.parser')
'''salva os links na variável LINK '''
link = soup.a['href']
'''Acessa o Link seguinte '''
html = requests.get(url+link)
#Busca o conteúdo da página
soup2 = BeautifulSoup(html.text, 'html.parser')
''' Busca todo as tags para link'''
links = soup2.find_all('a')
dicionario_datas_links = {}
'''Percorre os links acima'''
for i in links:
  if '1996' in str(i):
```

```
      break
   if '.html' and '0' in str(i):
   '''Adiciona a data e link em um dicionário'''
       dicionario_datas_links[i.string] =  i['href']
subpagina = url+'webreports'
'''Entrando no link'''
lista_data_hora = []
'''Percorre  o dicionario'''
for chave, valor in dicionario_datas_links.items():
   html2 = requests.get(subpagina+'/'+valor)
   print('Link ', html2.url, ' - Status: ', html2.status_code)
   lista_data_hora.append(html2.url)
'''Acessando a quarta pagina'''
 '''Acessando o link'''

'''ultima pagina'''
csv_file = {}
result = []
texto = []
"""
for link in lista_data_hora:
   html3 = requests.get(str(link))
   soup3 = BeautifulSoup(html3.text, 'html.parser')
   links = soup3.find_all('a')
   for i in links:
      print(i['href'])
      lista_data_hora.append(subpagina+'/'+i['href'])
   links = []
   for i in lista_data_hora:
      html4 = requests.get(i)
      soup4 = BeautifulSoup(html4.text, 'html.parser')
      tag = soup4.find_all('td')
      for i in tag:
         links.append(i)
         #file = open('saida.txt', 'w')
         for i in links:

            #file.write(i.get_text())
            #file.write("\n")
            texto.append(i.get_text())

         result.append(texto)
         texto = []
         print("Adicionando")
         print()
         print(result)
         print()


      print('-----------------')

      print('ESCREVENDO')
```

```
        print('-----------------------')
        escrever_csv = pd.DataFrame(result)
        escrever_csv.to_csv('log.csv', index=False, sep=',')

    print('-----------------')


print('ESCREVENDO')

        print('------‘)
```

4. Considerações Finais

- Nessa primeira parte do projeto de ciência de dados e inteligência artificial, podemos observar ao longo do desenvolvimento do código o quão é importante e necessário o trabalho de um cientista de dados. Além de possibilitar trabalhar com novas tecnologias e bibliotecas como request, beautifulsoup e o pandas para uma melhor compreensão, organização e coleta dos dados.
- Durante o desenvolvimento todo o grupo em alguma parte do projeto apresentou dificuldades, porém com as reuniões realizadas foi possível discutir e resolve-las.

Referências

RICHARDSON, Leonard. Beautiful soup documentation**. Beautiful Soup 4.9.0 Documentation**. Estados Unidos, c2004-2020. Disponível em: <https://www.crummy.com/software/BeautifulSoup/bs4/doc/>. Acesso em: 12 Set. de 2020

GUIA Rápido. **Requests.** Estados Unidos**,** c2013. Disponível em: <https://requests.readthedocs.io/pt_BR/latest/user/quickstart.html>. Acesso em: 11 Set. de 2020

FIGUEIREDO, Vinicius. Seus Primeiros Passos com Data Scientist: Introdução ao Pandas. **Data Hackers**. São Paulo, 30 de maio de 2018. Disponível em: <https://medium.com/data-hackers/uma-introdu%C3%A7%C3%A3o-simples-ao-pandas-1e15eea37fa1>. Acesso em: 08 de Set. de 2020

HERBERT, Anthony. Como começar a usa a biblioteca request em Python. **Community**. Estados Unidos, 19 de Mar de 2020. Disponível em: <https://www.digitalocean.com/community/tutorials/how-to-get-started-with-therequests-library-in-python-pt>. Acesso em: 08 de Set.2020

TAGLIAFERRI, Lisa. Como trabalhar com dados da web usando Requests e Beautiful soup com Python 3. **Community**. Estados Unidos, 09 de Jul de 2018. Disponível em: <
https://www.digitalocean.com/community/tutorials/como-trabalhar-com-dados-da-web-usando-requests-e-beautiful-soup-com-python-3-ptt>. Acesso em: 08 de Set.2020