Avaliação 1 – Modulo 10 MBA BigData Unipe

Componentes:

* Thiago Silveira
* Thiago Rafael
* Danilo Padilha

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Implemente um programa que entre no site do UOL e imprima apenas a seguinte mensagem: A cotação atual do dólar é: <cotação>, onde <cotação> vai ser o valor capturado do site no momento. Procure uma forma de omitir as mensagens de log na execução do seu programa para aparecer apenas essa mensagem como saída.**
2. import scrapy
3. class UolSpider(scrapy.Spider):
4. name = 'uol\_spider'
5. start\_urls = ['https://www.uol.com.br/']
6. def parse(self,response):
7. titulos = response.css(".HU\_currency\_\_quote::text")
8. conteudo = titulos[0].get().strip()
9. if conteudo != "":
10. print('A cotação do dólar é: ',conteudo)

**SAIDA DO PROGRAMA DESTACADA EM AMARELO A SEGUIR:**

Microsoft Windows [versão 10.0.18363.1139]

(c) 2019 Microsoft Corporation. Todos os direitos reservados.

C:\Users\User\Documents\códigos>C:/Users/User/anaconda3/Scripts/activate

(base) C:\Users\User\Documents\códigos>conda activate base

(base) C:\Users\User\Documents\códigos>scrapy runspider exercicio01.py

2020-10-24 12:54:59 [scrapy.utils.log] INFO: Scrapy 2.3.0 started (bot: scrapybot)

2020-10-24 12:54:59 [scrapy.utils.log] INFO: Versions: lxml 4.5.2.0, libxml2 2.9.10, cssselect 1.1.0, parsel 1.6.0, w3lib 1.22.0, Twisted 20.3.0, Python 3.8.3 (default, Jul 2 2020, 17:30:36) [MSC v.1916 64 bit (AMD64)], pyOpenSSL 19.1.0 (OpenSSL 1.1.1g 21 Apr 2020), cryptography 2.9.2, Platform Windows-10-10.0.18362-SP0

2020-10-24 12:54:59 [scrapy.utils.log] DEBUG: Using reactor: twisted.internet.selectreactor.SelectReactor

2020-10-24 12:54:59 [scrapy.crawler] INFO: Overridden settings:

{'SPIDER\_LOADER\_WARN\_ONLY': True}

2020-10-24 12:54:59 [scrapy.extensions.telnet] INFO: Telnet Password: 8411227c73f62991

2020-10-24 12:54:59 [scrapy.middleware] INFO: Enabled extensions:

['scrapy.extensions.corestats.CoreStats',

'scrapy.extensions.telnet.TelnetConsole',

'scrapy.extensions.logstats.LogStats']

2020-10-24 12:55:00 [scrapy.middleware] INFO: Enabled downloader middlewares:

['scrapy.downloadermiddlewares.httpauth.HttpAuthMiddleware',

'scrapy.downloadermiddlewares.downloadtimeout.DownloadTimeoutMiddleware',

'scrapy.downloadermiddlewares.defaultheaders.DefaultHeadersMiddleware',

'scrapy.downloadermiddlewares.useragent.UserAgentMiddleware',

'scrapy.downloadermiddlewares.retry.RetryMiddleware',

'scrapy.downloadermiddlewares.redirect.MetaRefreshMiddleware',

'scrapy.downloadermiddlewares.httpcompression.HttpCompressionMiddleware',

'scrapy.downloadermiddlewares.redirect.RedirectMiddleware',

'scrapy.downloadermiddlewares.cookies.CookiesMiddleware',

'scrapy.downloadermiddlewares.httpproxy.HttpProxyMiddleware',

'scrapy.downloadermiddlewares.stats.DownloaderStats']

2020-10-24 12:55:00 [scrapy.middleware] INFO: Enabled spider middlewares:

['scrapy.spidermiddlewares.httperror.HttpErrorMiddleware',

'scrapy.spidermiddlewares.offsite.OffsiteMiddleware',

'scrapy.spidermiddlewares.referer.RefererMiddleware',

'scrapy.spidermiddlewares.urllength.UrlLengthMiddleware',

'scrapy.spidermiddlewares.depth.DepthMiddleware']

2020-10-24 12:55:00 [scrapy.middleware] INFO: Enabled item pipelines:

[]

2020-10-24 12:55:00 [scrapy.core.engine] INFO: Spider opened

2020-10-24 12:55:00 [scrapy.extensions.logstats] INFO: Crawled 0 pages (at 0 pages/min), scraped 0 items (at 0 items/min)

2020-10-24 12:55:00 [scrapy.extensions.telnet] INFO: Telnet console listening on 127.0.0.1:6023

2020-10-24 12:55:00 [scrapy.core.engine] DEBUG: Crawled (200) <GET https://www.uol.com.br/> (referer: None)

A cotação do dólar é: 5,627

2020-10-24 12:55:00 [scrapy.core.engine] INFO: Closing spider (finished)

2020-10-24 12:55:00 [scrapy.statscollectors] INFO: Dumping Scrapy stats:

{'downloader/request\_bytes': 218,

'downloader/request\_count': 1,

'downloader/request\_method\_count/GET': 1,

'downloader/response\_bytes': 86959,

'downloader/response\_count': 1,

'downloader/response\_status\_count/200': 1,

'elapsed\_time\_seconds': 0.436019,

'finish\_reason': 'finished',

'finish\_time': datetime.datetime(2020, 10, 24, 15, 55, 0, 565761),

'log\_count/DEBUG': 1,

'log\_count/INFO': 10,

'response\_received\_count': 1,

'scheduler/dequeued': 1,

'scheduler/dequeued/memory': 1,

'scheduler/enqueued': 1,

'scheduler/enqueued/memory': 1,

'start\_time': datetime.datetime(2020, 10, 24, 15, 55, 0, 129742)}

2020-10-24 12:55:00 [scrapy.core.engine] INFO: Spider closed (finished)

(base) C:\Users\User\Documents\códigos>

**2) Implemente um programa que receba um produto como parâmetro e liste o nome e o preço de todos esses produtos no mercado livre, com paginação incluída. Busque uma forma de passar um parâmetro para o seu programa.**

from selenium import webdriver

from selenium.webdriver.common.by import By

from selenium.common.exceptions import NoSuchElementException

from selenium.common.exceptions import TimeoutException, WebDriverException

from selenium.webdriver.support.ui import WebDriverWait

import json

def finds(browser, by, expression):

    try:

        return browser.find\_elements(by, expression)

    except NoSuchElementException as nse:

        return None

def find(browser, by, expression):

    try:

        return browser.find\_element(by, expression)

    except NoSuchElementException as nse:

        return None

a=input("Selecione o item\n")

output  = open('output.json', 'w', encoding='utf-8')

driver = webdriver.Firefox()

driver.get("https://lista.mercadolivre.com.br/" + a)

items = finds(driver, By.CLASS\_NAME, 'ui-search-result\_\_content-wrapper')

for item in items:

    descricao = find(item, By.CLASS\_NAME, 'ui-search-item\_\_title').get\_property("textContent")

    preco = find(item, By.CLASS\_NAME, 'ui-search-price').get\_property("textContent")

    print(descricao)

    print(preco)

    item = {'descricao': descricao, 'preco': preco}

    output.write(json.dumps(item))

    output.write('\n')

driver.close()

Microsoft Windows [versão 10.0.18363.1139]

(c) 2019 Microsoft Corporation. Todos os direitos reservados.

C:\Users\User\Documents\códigos>C:/Users/User/anaconda3/Scripts/activate

(base) C:\Users\User\Documents\códigos>conda activate base

(base) C:\Users\User\Documents\códigos>python exercicio02.py

Selecione o item

polo tsi

Volkswagen Polo 1.0 Tsi Highline 200 Aut. 5p

R$79.990

Volkswagen Polo 1.0 Tsi Comfortline 200 Aut. 5p

R$71.990

Volkswagen Polo Polo 200 Tsi Highline (aut) (flex)

R$67.990

Volkswagen Polo 1.0 Tsi Highline 200 Aut. 5p

R$69.989

Volkswagen Polo Polo Highline 200 Tsi 1.0 Flex 12v Aut.

R$68.890

Volkswagen Polo 1.0 Tsi Comfortline 200 Aut. 5p

R$51.790

Volkswagen Polo Polo Highline 200 Tsi 1.0 Flex 12v Aut.

R$69.990

Volkswagen Polo 1.0 Tsi Highline 200 Aut. 5p

R$69.800

Volkswagen Polo 1.0 Tsi Comfortline 200 Aut. 5p

R$59.990

Volkswagen Polo 1.0 200 Tsi Comfortline

R$69.800

Volkswagen Polo Polo Highline 200 Tsi 1.0 Flex 12v Aut.

R$74.990

Volkswagen Polo Polo Highline 200 Tsi 1.0 Flex 12v Aut.

R$75.990

Volkswagen Polo 1.0 Tsi Comfortline 200 Aut. 5p

R$57.900

Volkswagen Polo 1.0 200 Tsi Comfortline

R$62.269

Volkswagen Polo 1.0 Tsi Highline 200 Aut. 5p

R$69.900

Volkswagen Polo 1.0 Tsi Comfortline 200 Aut. 5p

R$59.500

Volkswagen Polo 1.0 200 Tsi Highline Automático

R$68.900

Volkswagen Polo Polo Highline 200 Tsi 1.0 Flex 12v Aut

R$69.900

Volkswagen Polo 1.0 Tsi Comfortline 200 Aut. 5p

R$62.980

Volkswagen Polo 1.0 Tsi Highline 200 Aut. 5p

R$74.000

Volkswagen Polo 1.0 200 Tsi Comfortline

R$73.900

Volkswagen Polo 1.0 200 Tsi Comfortline Automático

R$71.500

Volkswagen Polo 1.0 Tsi Comfortline 200 Aut. 5p

R$57.900

Volkswagen Polo 1.0 Tsi Comfortline 200 Aut. 5p

R$69.870

Volkswagen Polo 1.0 200 Tsi Comfortline Automático

R$68.613

Volkswagen Polo 1.0 200 Tsi Comfortline Automático

R$68.613

Volkswagen Polo 1.0 200 Tsi Comfortline Automático

R$74.900

Volkswagen Polo 1.0 200 Tsi Highline

R$63.990

Volkswagen Polo 1.0 200 Tsi Comfortline

R$60.990

Volkswagen Mlb24278 Polo 1.0 200 Tsi Comfortline Automático

R$62.990

Volkswagen Polo 1.4 Gts 250 Tsi Aut. 5p

R$153.999

Volkswagen Polo 1.0 Tsi Highline 200 Aut. 5p

R$76.990

Volkswagen Polo Polo Comfortline 200

R$61.900

Volkswagen Polo 1.0 Tsi Highline 200 Aut. 5p

R$106.999

Volkswagen Polo 1.0 200 Tsi Highline Automático

R$88.990

Volkswagen Polo 1.0 200 Tsi Comfortline Automático

R$81.990

Volkswagen Polo 1.0 200 Tsi Highline Automático

R$63.890

Volkswagen Polo 1.0 Tsi Comfortline 200 Aut. 5p

R$68.990

Volkswagen Polo 1.o Tsi Comfortline Automático

R$69.900

Volkswagen Polo 1.0 Tsi Highline 200 Aut. 5p

R$69.900

Volkswagen Polo 1.0 Tsi Comfortline 200 Aut. 5p

R$75.890

Volkswagen Polo 1.0 200 Tsi Highline Automático

R$63.900

Volkswagen Polo 1.0 Tsi Highline 200 Aut. 5p

R$72.900

Volkswagen Polo 1.0 Tsi Comfortline 200 Aut. 5p

R$63.300

Volkswagen Polo 1.0 200 Tsi Comfortline Automático

R$67.890

Volkswagen Polo 1.0 Tsi Comfortline 200 Aut. 5p

R$71.990

Volkswagen Polo 1.0 Tsi Comfortline 200 Aut. 5p

R$57.900

Volkswagen Polo Gts 1.4 250 Tsi

R$95.990

(base) C:\Users\User\Documents\códigos>

**3) Quais cuidados devem ser observados ao capturar dados de um site?**

Os cuidados recomendados, ao capturar dados de um site, são:

* O conteúdo capturado não tem direitos autorais;
* O scraping não onera os serviços do site;
* O scraper não viola os termos de uso do site; e
* O scraper não coleta informações sensíveis.

**4) Quais ameaças capturas automáticas proporcionam para sistemas web?**

A captura automática possibilita que hackers consigam acessar o conteúdo de outras empresas, prejudicando o negócio, como por exemplo:

* Sobrecarregando os servidores do sistema e impedindo o acesso de usuários e clientes;
* Usufruindo de conteúdos que são protegidos por direitos autorais;
* Repassando informação de produtos e serviços para a concorrência.

**5) Você diria que *bots* ou *crawlers* são programas facilmente paralelizáveis? Se sim, explique como isso seria implementado dando um exemplo.**

Resposta: Sim, são programas facilmente paralelizáveis. Como exemplo, poderia ser criado uma tarefa que pode ser dividida em subtarefas, no qual não há necessidade de ordem na execução bem como troca de informações entre essas subtarefas.