



TERMO DE CONDUTA

Declaro assumir o compromisso de confidencialidade e de sigilo escrito, fotográfico e verbal sobre as questões do exame ou avaliação pessoal que me serão apresentadas, durante o curso desta disciplina. Comprometo-me a não revelar, reproduzir, utilizar ou dar conhecimento, em hipótese alguma, a terceiros, e a não utilizar tais informações para gerar benefício próprio ou de terceiros. Reitero minha ciência de que não poderei fazer cópia manuscrita, registro fotográfico, filmar ou mesmo gravar os enunciados que me são apresentados.

Também me comprometo a não postar as soluções das questões em nenhum site ou arquivo na nuvem, bem como não as enviarei, para qualquer recipiente, seja por correio eletrônico, ou por qualquer outro meio disponível.

Declaro, ainda, estar ciente de que o não cumprimento de tais normas caracteriza infração ética, podendo acarretar punição de acordo com as regras da minha universidade.

Ciente,

Assinatura do aluno



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância
Curso de Tecnologia em Sistemas de Computação
Disciplina: Programação com Interfaces Gráficas
APX2 2º semestre de 2021.

Nome -

Assinatura -

1. (10 pontos) Nessa AP2¹, você irá escrever um *Web Scraper* para obter as temperaturas mínima e máxima de uma cidade.

Web scraping é uma coleta de dados da web, onde são usados *scripts* e programas para “raspar” informações de certos sites, que poderão ser usadas para análises futuras.

Por exemplo, usando o site do ClimaTempo², pode-se obter as informações sobre o tempo de qualquer cidade do Brasil, e assim imprimir no terminal, as temperaturas mínima e máxima do dia.

Temperatura em RIODEJANEIRO-RJ:

- Mínima: 19°
- Máxima: 23°

Sua tarefa é implementar uma interface gráfica para exibir as temperaturas mínima e máxima do dia, escrevendo para isso a função `get_city_tmin_and_tmax` e seguindo as recomendações abaixo:

¹Requer BeautifulSoup >= 4.7 - <https://pypi.org/project/beautifulsoup4/>

²<https://www.climatempo.com.br>

Código 1: BeautifulSoup

```
#!/usr/bin/env python

from bs4 import BeautifulSoup
import requests
import sys

##
# Retorna uma tupla com o nome da cidade indicada,
# e suas temperaturas mínima e máxima.
#
# Cerca de uma dúzia de linhas de código dão conta do recado!
#
# @param index índice de uma cidade [0..9]
# @param offline quando usar uma página pré-baixada.
# @param debug estado de depuração: imprime o código html do site.
# @return (nome da cidade, temperatura mínima, temperatura máxima)
#
def get_city_tmin_and_tmax(index, offline=False, debug=False):
    # ....

    if not offline:
        html = requests.get(
            "https://www.climatempo.com.br/previsao-do-tempo/cidade/" +
            ...).content
    else:
        # html pré-baixado, para o caso do Climatempo estar fora do ar
        html = open("riodejaneiro-rj.html", 'r').read()

    soup = BeautifulSoup(html, 'html.parser')
    if debug:
        print(soup.prettify())

    # ....

    return cidade, tmin, tmax
```

1. Você deve escolher dez cidades quaisquer para criar um menu de cidades no seu programa.
2. Cada cidade possui um código que pode ser obtido selecionando-se uma cidade³ no site e inspecionando-se a URL criada. Por exemplo, a cidade do Rio de Janeiro é identificada como: "321/riodejaneiro-rj", e portanto, a URL correspondente será:

"https://www.climatempo.com.br/previsao-do-tempo/cidade/321/riodejaneiro-rj"

Dessa forma, sugiro que se crie uma lista de cidades a ser utilizada na implementação.

3. Use elementos de interface do tkinter adequados a seleção de uma das dez cidades e exibição das suas temperaturas.
4. Adicione qualquer outra característica/opção que torne o seu programa mais interessante.

Não esqueça de implementar uma função help com as instruções sobre a utilização do seu código.

A figura 1 exibe o trecho do fonte da página do Climatempo que deve ser utilizado para extrair as temperaturas. Pode-se empregar qualquer navegador para obter essa informação, mas geralmente, usa-se o Google Chrome.

³<https://www.climatempo.com.br/previsao-do-tempo?page=HOJE>

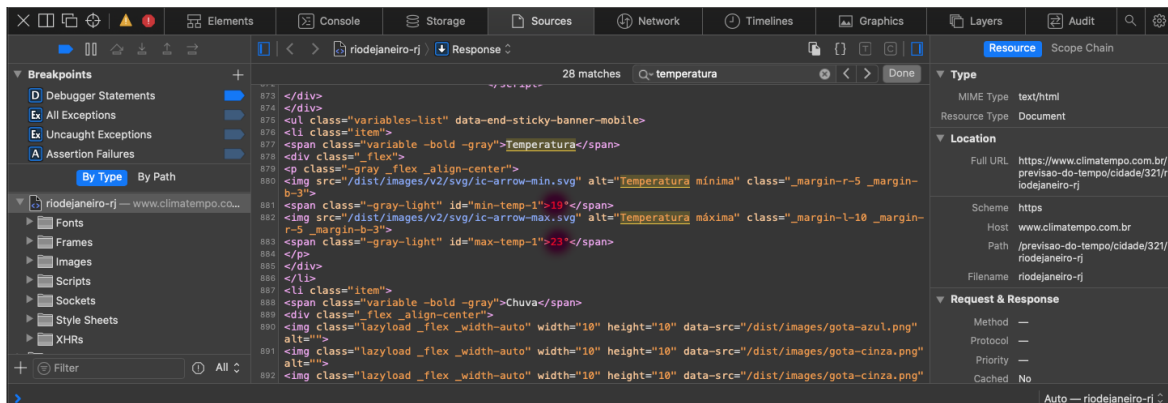


Figura 1: Trecho do Fonte do Climatedo no Safari.

Para ajudá-lo nessa tarefa, existem instruções em vários sites ^{4 5 6} que explicam, detalhadamente, como implementar um *scraper*. Infelizmente, toda vez que estrutura do Climatedo mudar, você deverá adaptar as chamadas ao BeautifulSoup⁷ para fazê-lo funcionar novamente (mesmo porque, não haveria diversão alguma, se tudo já estivesse pronto, não é mesmo?).

Caso o site do Climatedo esteja fora do ar, ou haja algum problema de conexão, é possível utilizar uma página pré-baixada ^{8 9 10} com os dados do riodejaneiro-rj, conforme pode ser visto no código 1.

Você deve entregar um programa que funcione, ou seja, enviar todos arquivos .py, imagens, e qualquer outro recurso necessário à execução do programa. O único pacote que deve ser instalado é o BeautifulSoup. Não serão aceitos programas idênticos ou que importem qualquer outro pacote externo.

Boa sorte.

⁴<https://blog.geekhunter.com.br/como-fazer-um-web-scraping-python/>

⁵<https://realpython.com/beautiful-soup-web-scraper-python/>

⁶<https://www.edureka.co/blog/web-scraping-with-python/>

⁷<https://www.crummy.com/software/BeautifulSoup/bs4/doc/index.html>

⁸<http://orion.lcg.ufrj.br/python/public/riodejaneiro-rj.html>

⁹<http://orion.lcg.ufrj.br/python/public/saopaulo-sp.html>

¹⁰<http://orion.lcg.ufrj.br/python/public/previsao-do-tempo.html>