

1. Modifique o seguinte código para retornar a resposta da requisição "GET" em formato XML:

Foi somente necessário a troca do formato, de /JSON/ para /XML/

```
teste.py 1 python-3.13.7-amd64.exe
C: > Users > Alegay > Desktop > atitt > teste.py > ...
1  import requests
2
3  url = 'https://viacep.com.br/ws/'
4  cep = '30140071'
5  formato = '/xml/'
6
7  r = requests.get(url + cep + formato)
8
9  if r.status_code == 200:
10     print()
11     print('XML : ')
12     print(r.text)
13     print()
14 else:
15     print('Nao houve sucesso na requisicao.')
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
JSON : {'cep': '30140-071', 'logradouro': 'Rua dos Aimorés', 'complemento': 'de 971/972 a 1399/1400', 'unidade': '', 'bairro': 'Boa Viagem', 'localidade': 'Belo Horizonte', 'uf': 'MG', 'estado': 'Minas Gerais', 'regiao': 'Sudeste', 'ibge': '3106200', 'gia': '', 'ddd': '31', 'siafi': '4123'}
```

PS C:\Users\Alegay> & C:\Users\Alegay\AppData\Local\Programs\Python\Python313\python.exe c:/Users/Alegay/Desktop/atitt/teste.py

```
XML :
<?xml version="1.0" encoding="UTF-8"?>
<xmlcep>
  <cep>30140-071</cep>
  <logradouro>Rua dos Aimorés</logradouro>
  <complemento>de 971/972 a 1399/1400</complemento>
  <unidade></unidade>
  <bairro>Boa Viagem</bairro>
  <localidade>Belo Horizonte</localidade>
  <uf>MG</uf>
  <estado>Minas Gerais</estado>
  <regiao>Sudeste</regiao>
  <ibge>3106200</ibge>
  <gia></gia>
  <ddd>31</ddd>
  <siafi>4123</siafi>
</xmlcep>
```

PS C:\Users\Alegay> []

2. Modifique o código anterior para que, usando uma estrutura de repetição, consulte 5 CEPs sequenciais. Suponha que o CEP da sua casa seja 30140-071, o código Python deverá enviar 5 requisições com os seguintes CEPs: 30140071, 30140072, 30140073, 30140074 e 30140075

Fiz um for começando do 071 e repetindo mais 4 vezes até o 075 para que ele me retornasse todas as requisições:

```
teste.py 1 python-3.13.7-amd64.exe
C:\Users\Alegay\Desktop> atttt > teste.py > ...
1  import requests
2
3  url = 'https://viacep.com.br/ws/'
4  cep = '30140071'
5  formato = '/xml/'
6
7  r = requests.get(url + cep + formato)
8
9  if r.status_code == 200:
10     print()
11     print('XML : ')
12     print(r.text)
13     print()
14 else:
15     print('Nao houve sucesso na requisicao.')
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
JSON : {'cep': '30140-071', 'logradouro': 'Rua dos Aimorés', 'complemento': 'de 971/972 a 1399/1400', 'unidade': '', 'bairro': 'Boa Viagem', 'localidade': 'Belo Horizonte', 'uf': 'MG', 'estado': 'Minas Gerais', 'regiao': 'Sudeste', 'ibge': '3106200', 'gia': '', 'ddd': '31', 'siafi': '4123'}
```

PS C:\Users\Alegay> & C:/Users/Alegay/AppData/Local/Programs/Python/Python313/python.exe c:/Users/Alegay/Desktop/atttt/teste.py

```
XML :
<?xml version="1.0" encoding="UTF-8">
<xmlcep>
  <cep>30140-071</cep>
  <logradouro>Rua dos Aimorés</logradouro>
  <complemento>de 971/972 a 1399/1400</complemento>
  <unidade></unidade>
  <bairro>Boa Viagem</bairro>
  <localidade>Belo Horizonte</localidade>
  <uf>MG</uf>
  <estado>Minas Gerais</estado>
  <regiao>Sudeste</regiao>
  <ibge>3106200</ibge>
  <gia></gia>
  <ddd>31</ddd>
  <siafi>4123</siafi>
</xmlcep>
```

PS C:\Users\Alegay>

3. Modifique o código para que a consulta seja feita com um endereço (nome de rua), ao invés do CEP. Veja a URL de exemplo: viacep.com.br/ws/MG/Belo Horizonte/Rua dos Aimores/json/

Fique atento pois, o resultado não será apenas um único registro.

Troquei o formato dos parâmetros, e fui instruído a colocar a linha `dados = r.json()` para uma lista de dicionários e fazer um loop para os vários registros que retornariam.

```
teste.py 1 python-3.13.7-amd64.exe
C:\Users\Alegay\Desktop> atttt > teste.py > ...
1 import requests
2
3
4 uf = "MG"
5 cidade = "Belo Horizonte"
6 logradouro = "Rua dos Aimores"
7
8 url = f"https://viacep.com.br/ws/{uf}/{cidade}/{logradouro}/json/"
9
10 r = requests.get(url)
11
12 if r.status_code == 200:
13     print("\nResultados da consulta:\n")
14     dados = r.json()
15
16     for item in dados:
17         print(f"CEP: {item['cep']}")
18         print(f"Logradouro: {item['logradouro']}")
19         print(f"Bairro: {item['bairro']}")
20         print(f"Cidade: {item['localidade']}")
21         print(f"UF: {item['uf']}")
22         print("-" * 40)
23
24
25 else:
26     print("Não houve sucesso na requisição.")
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS +

```
UF: MG
-----
CEP: 30140-920
Logradouro: Rua dos Aimorés
Bairro: Funcionários
Cidade: Belo Horizonte
UF: MG
-----
CEP: 30140-904
Logradouro: Rua dos Aimorés
Bairro: Funcionários
Cidade: Belo Horizonte
UF: MG
-----
CEP: 30140-070
Logradouro: Rua dos Aimorés
Bairro: Funcionários
Cidade: Belo Horizonte
UF: MG
-----
CEP: 30140-078
Logradouro: Rua dos Aimorés
Bairro: Barro Preto
Cidade: Belo Horizonte
UF: MG
-----
PS C:\Users\Alegay>
```

4. Modifique o primeiro código de tal forma que o endereço "<https://viacep.com.br/abc/>" tente ser acessado. Exiba o código de retorno e o texto.

Pesquisando tentei puxar o site com a URL e um try para me retornar se deu certo ou não, e um print de retorno:

```
C:\Users\Alegay\Desktop> atttt > teste.py > ...
1  import requests
2
3  url = 'https://viacep.com.br/abc/'
4
5  try:
6      r = requests.get(url, timeout=10)
7      print('Status code:', r.status_code)
8      print('Resposta (texto):')
9      print(r.text)
10 except requests.RequestException as e:
11     print('Tentativa falhou:', e)
12
13
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
-----
CEP: 30140-078
Logradouro: Rua dos Aimorés
Bairro: Barro Preto
Cidade: Belo Horizonte
UF: MG
-----

PS C:\Users\Alegay> & C:/Users/Alegay/AppData/Local/Programs/Python/Python313/python.exe c:/Users/Alegay/Desktop/atttt/teste.py
Status code: 404
Resposta (texto):
<!DOCTYPE HTML>
<html lang="pt-br">
<head>
<title>ViaCEP 404</title>
<meta charset="utf-8"/>
<style type="text/css">h1{color:#555;text-align:center;font-size:4em}h2,h3{color:#666;text-align:center;font-size:3em}h3{font-size:1.5em}</style>
</head>
<body>
<h1>Http 404</h1>
<h3>Página Não Encontrada</h3>
<h3>{Not Found}</h3>
</body>
</html>

PS C:\Users\Alegay>
```

5. Observe o código abaixo:

Modifique-o para que os resultados sejam salvos em arquivo.

Pesquisei e tentei salvar o arquivo em txt com o retorno da busca, fazendo uma confirmação no if com o método “ r.status_code == 200:”

E fiz o arquivo ser salvo com o método “.write()” dentro de “With”

```
1 import requests
2
3 r = requests.get(
4     'http://www.google.com/search',
5     params={'q': 'elson de abreu'}
6 )
7
8 if r.status_code == 200:
9     print("Requisição bem sucedida! Salvando em arquivo...")
10
11     with open("resultado_google.txt", "w", encoding="utf-8") as f:
12         f.write(r.text)
13
14     print("Arquivo salvo como 'resultado_google.txt'")
15 else:
16     print("Não houve sucesso na requisição. Código:", r.status_code)
17     print("Não houve sucesso na requisição. Código:", r.status_code)
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS +

```
<head>
<title>ViaCEP 404</title>
<meta charset="utf-8"/>
<style type="text/css">h1{color:#555;text-align:center;font-size:4em}h2,h3{color:#666;text-align:center;font-size:3em}h3{font-size:1.5em}</style>
</head>
<body>
<h1>Http 404</h1>
<h3>Página Não Encontrada</h3>
<h3>{Not Found}</h3>
</body>
</html>
```

PS C:\Users\Alegay> & C:/Users/Alegay/AppData/Local/Programs/Python/Python313/python.exe c:/Users/Alegay/Desktop/atttt/teste.py
File "c:\Users\Alegay\Desktop\atttt\teste.py", line 2
r = requests.get('http://www.google.com/search', params={'q': 'elson
A
SyntaxError: unterminated string literal (detected at line 2)

PS C:\Users\Alegay> & C:/Users/Alegay/AppData/Local/Programs/Python/Python313/python.exe c:/Users/Alegay/Desktop/atttt/teste.py
Requisição bem sucedida! Salvando em arquivo...
Arquivo salvo como 'resultado_google.txt'

PS C:\Users\Alegay> & C:/Users/Alegay/AppData/Local/Programs/Python/Python313/python.exe c:/Users/Alegay/Desktop/atttt/teste.py
Arquivo salvo em: C:\Users\Alegay\resultado_google.txt

PS C:\Users\Alegay> []

[illegible]