

Nome: Marina Luiza de Jesus Ferreira

RA: 323111931

1-

The screenshot shows a Google Colab notebook interface. The code cell contains a Python script named Untitled2.py that uses the requests library to fetch XML data from a URL. The URL is https://vlacep.com.br/vn/ with a placeholder for a cep (zip code). The script prints the XML response if the status code is 200, or an error message if it's not. The XML response is then printed, showing details like state, city, neighborhood, complement, and address number. The Colab interface includes tabs for Comandos, Código, Texto, Executar tudo, and a sidebar with sharing and execution options.

```
import requests
url = 'https://vlacep.com.br/vn/'
cep = '30140071'
formato = 'xml' # alterado para XML
r = requests.get(url + cep + formato)

if r.status_code == 200:
    print()
    print(r.XML)
    print(r.text) # retorna o conteudo bruto em XML
else:
    print("Nao houve sucesso na requisicao.")

XML
<xml version="1.0" encoding="UTF-8">
<mlcep>
<csgp>30140-071</csgp>
<logradouro>rua dos Almoxarifes/Jogradouro</logradouro>
<complemento>de 971/972 a 1399/1400</complemento>
<unidade></unidade>
<bairro>Vila Olímpia/Bairro</bairro>
<localidade>Vila Olímpia/Horizonte</localidade>
<ufmgz/>
<estad>SP</estad>
<regiao>Sudeste/regiones</regiao>
<lbg>3106200</lbg>
<glav>glav</glav>
<ddid>ddid</ddid>
<siafi>4123</siafi>
</mlcep>
```

2-


```
<?xml version="1.0" encoding="UTF-8"?>
<cep>30140-074</cep>
<logradouro>rua dos Almeidass</logradouro>
<complemento>1801/1802 a 2369/2379</complemento>
<bairro>Centro</bairro>
<localidade>Belo Horizonte</localidade>
<uf>MG</uf>
<estado>Minas Gerais</estado>
<regiao>Sudeste</regiao>
<lrg>31862000</lrg>
<gl></gl>
<ddd>31</ddd>
<siafi>4123</siafi>
</siafep>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<cep>30140-074</cep>
<logradouro>rua dos Almeidass</logradouro>
<complemento>1801/1802 a 2369/2379</complemento>
<bairro>Centro</bairro>
<localidade>Belo Horizonte</localidade>
<uf>MG</uf>
<estado>Minas Gerais</estado>
<regiao>Sudeste</regiao>
<lrg>31862000</lrg>
<gl></gl>
<ddd>31</ddd>
<siafi>4123</siafi>
</siafep>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<cep>30140-075</cep>
<logradouro>rua dos Almeidass</logradouro>
<complemento>1801/1802 a 2369/2379</complemento>
<bairro>Centro</bairro>
<localidade>Belo Horizonte</localidade>
<uf>MG</uf>
<estado>Minas Gerais</estado>
<regiao>Sudeste</regiao>
<lrg>31862000</lrg>
<gl></gl>
<ddd>31</ddd>
<siafi>4123</siafi>
</siafep>
```

20:08 Python 3

25°C Limpo POR 2009 PIBZ 17/09/2025

3-

The screenshots show three consecutive executions of a Jupyter Notebook cell. The code uses the requests library to get data from [Viacep](https://viacep.com.br/we/) and then parses the XML response using ElementTree to extract address components like cep, logradouro, complemento, bairro, localidade, and unidade.

```
import requests
url = "https://viacep.com.br/we/"
uf = "MG"
cidade = "Belo Horizonte"
logradouro = "Rua dos Almores"
formato = '>/xml'
r = requests.get(url + uf + '/' + cidade + '/' + logradouro + formato)

if r.status_code == 200:
    print()
    print("Resultado da busca por endereço:")
    print(r.text)
    print()
else:
    print("Não houve sucesso na requisição.")

Resultado da busca por endereço:
<xml version="1.0" encoding="utf-8">
<cep>30140-071</cep>
<logradouro>Rua dos Almores</logradouro>
<complemento>971/972 a 1399/1400</complemento>
<bairro>Santo Agostinho</bairro>
<localidade>Belo Horizonte</localidade>
<uf>MG</uf>
<estado>Minas Gerais</estado>
<regiao>Sudeste</regiao>
<lote>3102000</lote>
<gfa>/gfa</gfa>
<ddd>31</ddd>
<siafi>4123</siafi>
</endereco>
</endereco>
```

The second execution shows a more detailed XML structure with additional fields like cep, logradouro, complemento, bairro, localidade, unidade, and siafi.

```
<endereco>
<cep>30140-071</cep>
<logradouro>Rua dos Almores</logradouro>
<complemento>971/972 a 1399/1400</complemento>
<bairro>Santo Agostinho</bairro>
<localidade>Belo Horizonte</localidade>
<uf>MG</uf>
<estado>Minas Gerais</estado>
<regiao>Sudeste</regiao>
<lote>3102000</lote>
<gfa>/gfa</gfa>
<ddd>31</ddd>
<siafi>4123</siafi>
</endereco>
<endereco>
<cep>30140-074</cep>
<logradouro>Rua dos Almores</logradouro>
<complemento>1801/1802 a 2309/2370</complemento>
<bairro>Centro Funcionários</bairro>
<localidade>Belo Horizonte</localidade>
<uf>MG</uf>
<estado>Minas Gerais</estado>
<regiao>Sudeste</regiao>
<lote>3102000</lote>
<gfa>/gfa</gfa>
<ddd>31</ddd>
<siafi>4123</siafi>
</endereco>
<endereco>
<cep>30140-075</cep>
<logradouro>Rua dos Almores</logradouro>
<complemento>531/532 a 969/970</complemento>
<bairro>Centro Funcionários</bairro>
<localidade>Belo Horizonte</localidade>
<uf>MG</uf>
<estado>Minas Gerais</estado>
<regiao>Sudeste</regiao>
<lote>3102000</lote>
<gfa>/gfa</gfa>
<ddd>31</ddd>
<siafi>4123</siafi>
</endereco>
<endereco>
<cep>30140-920</cep>
<logradouro>Rua dos Almores</logradouro>
<complemento>66</complemento>
<bairro>Centro Funcionários</bairro>
<localidade>Belo Horizonte</localidade>
<uf>MG</uf>
<estado>Minas Gerais</estado>
<regiao>Sudeste</regiao>
```

The third execution shows the final part of the XML structure, which includes a unit (unidade) named 'Clínica Hermes Pardini'.

```
<endereco>
<cep>30140-920</cep>
<logradouro>Rua dos Almores</logradouro>
<complemento>66</complemento>
<bairro>Centro Funcionários</bairro>
<localidade>Belo Horizonte</localidade>
<uf>MG</uf>
<estado>Minas Gerais</estado>
<regiao>Sudeste</regiao>
```

The image shows three vertically stacked screenshots of the Google Colab interface. Each screenshot displays a Jupyter notebook cell containing XML code. The XML code is identical across all three instances:

```
<gl></gl>
<ddd>31</ddd>
<siafi>4123</siafi>
</endereco>
<cep>30140-084</cep>
<logradouro>Rua dos Almorés</logradouro>
<complemento>462</complemento>
<unidade>Edifício Manhattan</unidade>
<bairro>Funcionários</bairro>
<localidade>Belo Horizonte</localidade>
<uf>MG</uf>
<estado>Minas Gerais</estado>
<regiao>Sudeste</regiao>
<ibge>31062000</ibge>
<gl></gl>
<ddd>31</ddd>
<siafi>4123</siafi>
</endereco>
<cep>30140-070</cep>
<logradouro>Rua dos Almorés</logradouro>
<complemento>de 3301/3302 ao fim</complemento>
<unidade></unidade>
<bairro>Barro Preto</bairro>
<localidade>Belo Horizonte</localidade>
```

Below the XML code, each screenshot shows a terminal window with the Python 3 interpreter running. The terminal output includes the timestamp "✓ 20:10", the command "Bolsa ultrapassa 145...", and the identifier "PTB2 17/09/2025".

4-

```
import requests
url = "https://viacep.com.br/abc/"
r = requests.get(url)
print('Status code:', r.status_code)
print('Resposta:')
print(r.text)
```

The output shows:

```
Status code: 404
Resposta:
<!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>
```

5-

```
import requests
r = requests.get('http://www.google.com/search', params={'q': 'elson de abreu'})
if r.status_code == 200:
    with open('resultado_google.html', 'w', encoding='utf-8') as f:
        f.write(r.text)
    print('Resultado salvo no arquivo "resultado_google.html".')
else:
    print("Não houve sucesso na requisição.")
```

The output shows:

```
Resultado salvo no arquivo "resultado_google.html".
```