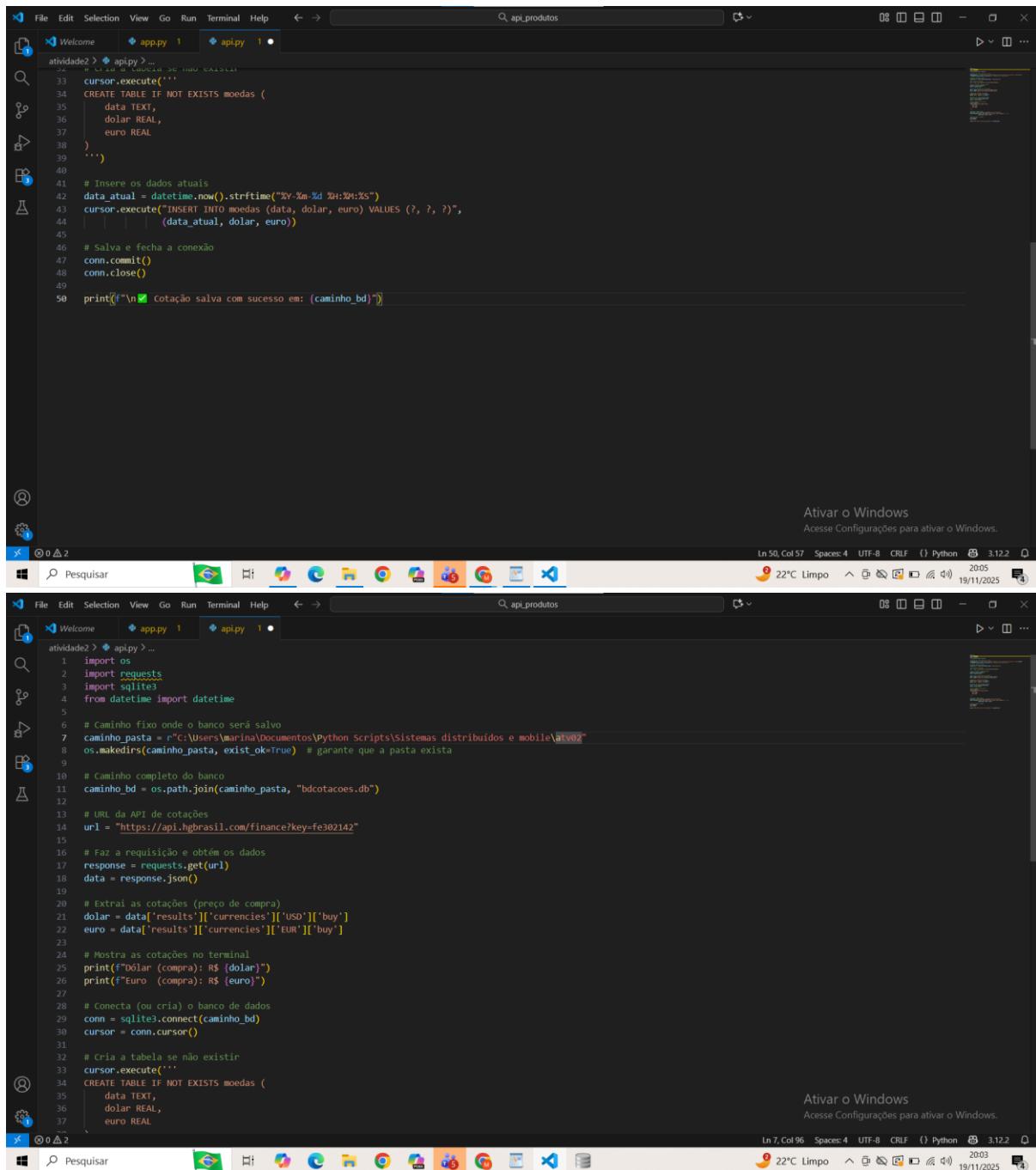


Atividade 02



```
atividade2 > app.py 1 api.py 1
32 # Cria a tabela se não existir
33 cursor.execute('''
34 CREATE TABLE IF NOT EXISTS moedas (
35     data TEXT,
36     dolar REAL,
37     euro REAL
38 )
39 ...
40
41 # Insere os dados atuais
42 data_atual = datetime.now().strftime("%Y-%m-%d %H:%M:%S")
43 cursor.execute("INSERT INTO moedas (data, dolar, euro) VALUES (?, ?, ?)",
44                 (data_atual, dolar, euro))
45
46 # Salva e fecha a conexão
47 conn.commit()
48 conn.close()
49
50 print(f"\nCotação salva com sucesso em: {caminho_bd}")
```

```
atividade2 > app.py 1 api.py 1
1 import os
2 import requests
3 import sqlite3
4 from datetime import datetime
5
6 # Caminho fixo onde o banco será salvo
7 caminho_pasta = r"C:\Users\marina\Documentos\Python Scripts\Sistemas distribuídos e mobile\ativ02"
8 os.makedirs(caminho_pasta, exist_ok=True) # garante que a pasta exista
9
10 # Caminho completo do banco
11 caminho_bd = os.path.join(caminho_pasta, "bdcotacoes.db")
12
13 # URL da API de cotações
14 url = "https://api.hgbrasil.com/finance?key=fe302142"
15
16 # Faz a requisição e obtém os dados
17 response = requests.get(url)
18 data = response.json()
19
20 # Extrai as cotações (preço de compra)
21 dolar = data['results'][0]['currencies']['USD']['buy']
22 euro = data['results'][0]['currencies']['EUR']['buy']
23
24 # Mostra as cotações no terminal
25 print(f"Dólar (compra): R$ {dolar}")
26 print(f"Euro (compra): R$ {euro}")
27
28 # Conecta (ou cria) o banco de dados
29 conn = sqlite3.connect(caminho_bd)
30 cursor = conn.cursor()
31
32 # Cria a tabela se não existir
33 cursor.execute('''
34 CREATE TABLE IF NOT EXISTS moedas (
35     data TEXT,
36     dolar REAL,
37     euro REAL
38 )
```

