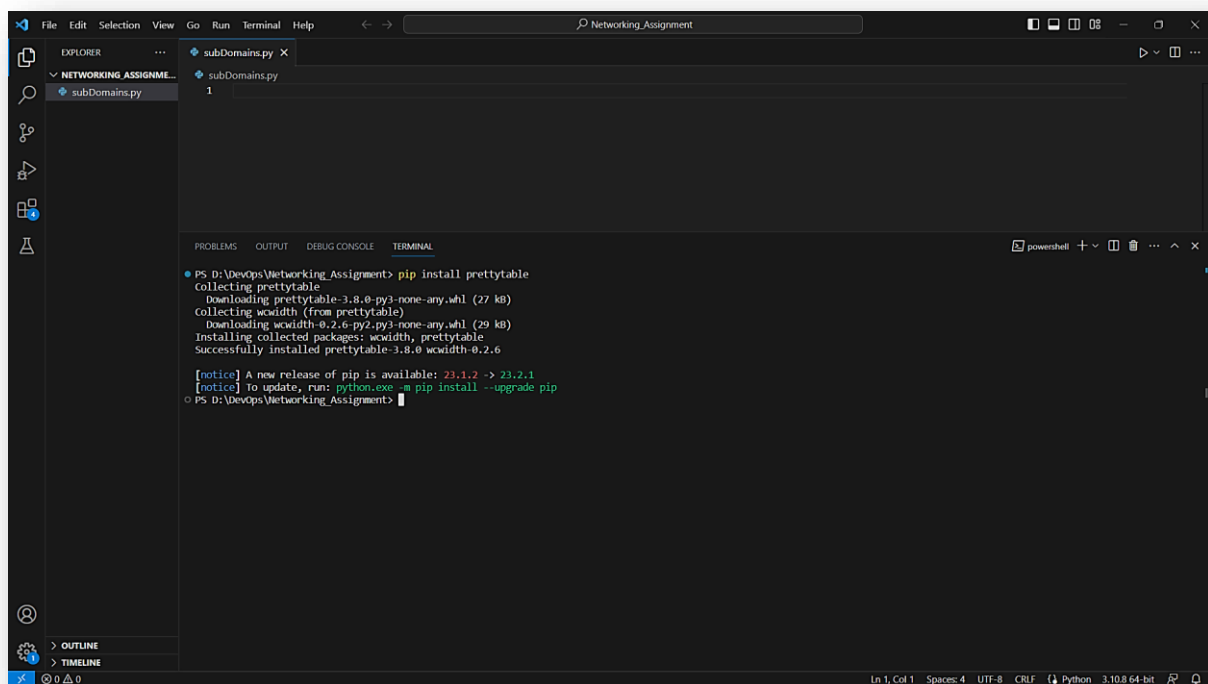


Subdomain Status Check

A website can have many subdomains and different services are running on them. Write a Python script to check the status of the subdomains which are up or down. The script should automatically check the status every min and should update it in tabular format on the screen. Write a detailed documentation of it.

Here is a step-by-step guide on how to check the status of the subdomains:

1. A **subdomain** is a prefix added to a domain name to separate a section of your website. Site owners primarily use subdomains to manage extensive sections that require their own content hierarchy, such as online stores, blogs, job boards or support platforms.
2. In this task we need to create a python script which automatically check the status of list of subdomains which are up or down.
3. The python script uses the requests library to send HTTP requests to the subdomains and check their status. Use the following command in the terminal to install the requests library : **pip install requests**



The screenshot shows a Visual Studio Code editor window titled 'Networking_Assignment'. The Explorer pane on the left shows a file named 'subDomains.py' under a folder named 'NETWORKING_ASSIGNME...'. The main editor area shows the content of 'subDomains.py', which contains a single line of code: `1`. Below the editor, the TERMINAL pane is active, showing the output of a command executed in a PowerShell terminal. The command was `pip install prettytable`, and the output shows the successful installation of 'prettytable-3.8.0-py3-none-any.whl' and 'wcwidth-0.2.6-py2.py3-none-any.whl'. A notice at the bottom of the terminal indicates that a new release of pip is available (23.1.2 -> 23.2.1) and suggests running `python.exe -m pip install --upgrade pip` to update it.

4. The prettytable library is used to display the status of the subdomains in a tabular format on the screen. : **pip install prettytable**
5. you can run the script by navigating to its directory in a terminal and running the command : **python subDomains.py**

The screenshot shows a VS Code window with a terminal open. The terminal output shows the installation of 'prettytable' and 'requests' packages. The 'prettytable' installation is successful. The 'requests' installation is also successful, with a notice about a new release of pip available (23.1.2 -> 23.2.1). The terminal output is as follows:

```
PS D:\Devops\Networking_Assignment> pip install prettytable
Collecting prettytable
  Downloading prettytable-3.8.0-py3-none-any.whl (27 kB)
Collecting wcwidth (from prettytable)
  Downloading wcwidth-0.2.6-py2.py3-none-any.whl (29 kB)
Installing collected packages: wcwidth, prettytable
Successfully installed prettytable-3.8.0 wcwidth-0.2.6

[notice] A new release of pip is available: 23.1.2 -> 23.2.1
[notice] To update, run: python.exe -m pip install --upgrade pip
PS D:\Devops\Networking_Assignment> pip install requests
Collecting requests
  Downloading requests-2.31.0-py3-none-any.whl (62 kB)
    62.6/62.6 kB 418.7 kB/s eta 0:00:00
Collecting charset-normalizer<4,>=2 (from requests)
  Downloading charset-normalizer-3.2.0-cp310-cp310-win_amd64.whl (96 kB)
    96.9/96.9 kB 463.5 kB/s eta 0:00:00
Collecting idna<4,>=2.5 (from requests)
  Downloading idna-3.4-py3-none-any.whl (61 kB)
    61.5/61.5 kB 545.7 kB/s eta 0:00:00
Collecting urllib3<3,>=1.21.1 (from requests)
  Downloading urllib3-2.0.4-py3-none-any.whl (123 kB)
    123.9/123.9 kB 607.9 kB/s eta 0:00:00
Collecting certifi<2017.4.17 (from requests)
  Downloading certifi-2023.7.22-py3-none-any.whl (158 kB)
    158.3/158.3 kB 632.9 kB/s eta 0:00:00
Installing collected packages: urllib3, idna, charset-normalizer, certifi, requests
Successfully installed certifi-2023.7.22 charset-normalizer-3.2.0 idna-3.4 requests-2.31.0 urllib3-2.0.4

[notice] A new release of pip is available: 23.1.2 -> 23.2.1
[notice] To update, run: python.exe -m pip install --upgrade pip
PS D:\Devops\Networking_Assignment>
```

6. Output :

The screenshot shows a VS Code window with a Python script named 'subDomains.py' open. The script is designed to check the status of subdomains. The output of the script is displayed in the terminal, showing a table of subdomains and their status.

```
8 # check status function is utilized to check the status of subdomains.
9 def check_status(subdomains):
10     # Create a table to display the results.
11     table = PrettyTable()
12     table.field_names = ["Subdomain", "Status"]
13
14     # Check the status of each subdomain using for loop.
15     for subdomain in subdomains:
16         # using try and except block to verify the status of the domains using requests.get() function from requests module.
17         try:
18             response = requests.get(f'http://{subdomain}', timeout=5)
19             if response.status_code == 200:
20                 # add_row() function from prettytable module is used to add rows of data to the table.
21                 table.add_row([subdomain, 'UP'])
22             else:
```

The output of the script is as follows:

```
Installing collected packages: urllib3, idna, charset-normalizer, certifi, requests
Successfully installed certifi-2023.7.22 charset-normalizer-3.2.0 idna-3.4 requests-2.31.0 urllib3-2.0.4

[notice] A new release of pip is available: 23.1.2 -> 23.2.1
[notice] To update, run: python.exe -m pip install --upgrade pip
PS D:\Devops\Networking_Assignment> python .\subDomains.py
```

Subdomain	Status
store.playstation.com	UP
uk.yahoo.com	UP
us.yahoo.com	UP
support.yotpo.com	UP
es.wikipedia.org	UP
eu.wikipedia.org	UP

Subdomain	Status
store.playstation.com	UP
uk.yahoo.com	DOWN
us.yahoo.com	UP
support.yotpo.com	UP
es.wikipedia.org	UP
eu.wikipedia.org	UP