**Placement Empowerment Program**

***Cloud Computing and DevOps Centre***

**Write a Python Script to Monitor an Application:**

Create a Python script that sends periodic HTTP requests to your application and alerts you if it’s down.

Name: Sivakumar C Department: AML



**Introduction**

Monitoring the availability of your application is crucial to ensure uninterrupted service for users. By implementing a Python script that periodically checks your application's status, you can promptly detect and address any downtime issues. This script sends HTTP requests to the application at regular intervals and alerts you if it becomes unresponsive.

**Objectives**

* **Automate Application Monitoring:** Develop a Python script that automatically sends HTTP requests to your application at regular intervals.
* **Detect Downtime Promptly:** Configure the script to recognize when the application is unresponsive or returns an error status code.
* **Implement Alert Mechanism:** Set up notifications (e.g., console alerts, emails) to inform you immediately if the application is down.

**Importance**

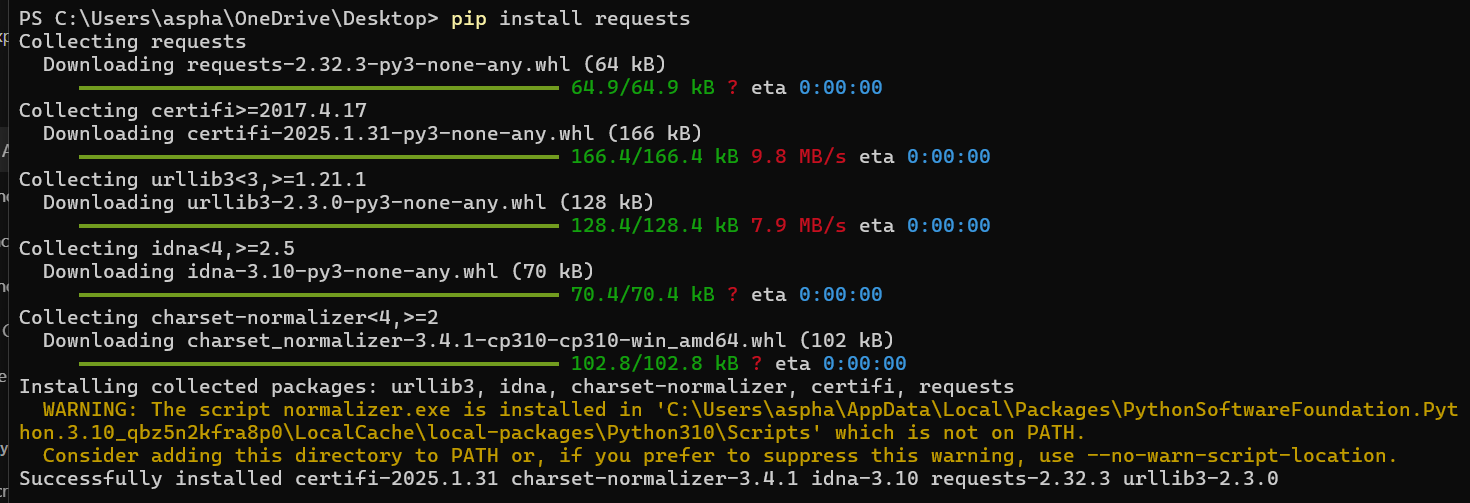
* **Ensures High Availability**: Continuous monitoring helps maintain the application's uptime, enhancing user satisfaction and trust.
* **Facilitates Rapid Response**: Immediate alerts enable quick troubleshooting and resolution of issues, minimizing potential revenue loss.
* **Supports Proactive Maintenance**: Regular monitoring can identify patterns or recurring issues, allowing for proactive system improvements.

**STEPS:**

**STEP 1:** **Install the Required Library**

Ensure the requests library is installed to handle HTTP requests. Install it using pip:

**pip install requests**

****

**STEP 2: Develop the Monitoring Script**

Create a Python script that sends HTTP GET requests to your application's URL and checks the response status code. If the response is not 200, the script will print an alert message.

Python Script:

**import requests**

**import time**

**# URL of your application**

**url = 'https://your-application-url.com'**

**# Monitoring interval in seconds**

**interval = 60 # Check every 60 seconds**

**while True:**

**try:**

**response = requests.get(url, timeout=10) # 10-second timeout**

**if response.status\_code == 200:**

**print(f'[{time.strftime("%Y-%m-%d %H:%M:%S")}] Application is up.')**

**else:**

**print(f'[{time.strftime("%Y-%m-%d %H:%M:%S")}] Alert: Application returned status code {response.status\_code}!')**

**except requests.RequestException as e:**

**print(f'[{time.strftime("%Y-%m-%d %H:%M:%S")}] Error: Could not reach the application. Exception: {e}')**

**time.sleep(interval)**

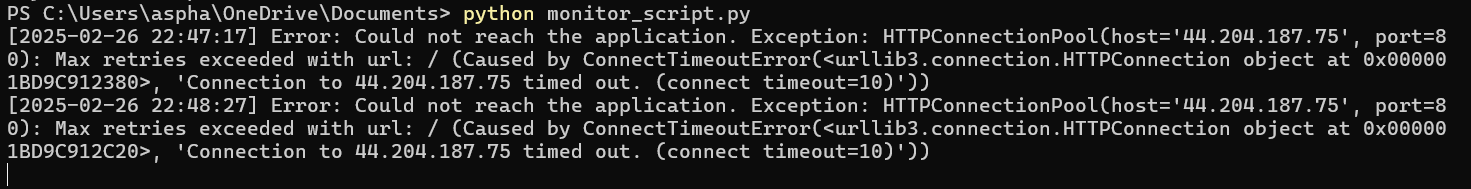
**STEP 3: Run the Script**

Save the script as monitor.py and run it using:

**python monitor.py**

This will continuously check your application's status every 60 seconds.



In case , the application is not running or failed , it might show  
  
Press Ctrl+C to stop the execution of the program.