perplexity

Generate a script that runs 'extract_networkConfig_localHost.py', 'get_localHost_sysInfo.py', and 'activeApp_and_networkTraffic_localHost.py' on every two minutes.

Here’s a Python script that schedules the execution of three scripts (`extract_networkConfig_localHost.py`, `get_localHost_sysInfo.py`, and `activeApp_and_networkTraffic_localHost.py`) every two minutes using the `schedule` library.

Python Script: Scheduler for Running Scripts Every Two Minutes

```
import schedule
import time
import subprocess

# Define functions to run each script
def run_extract_network_config():
    print("Running extract_networkConfig_localHost.py...")
    subprocess.run(["python", "extract_networkConfig_localHost.py"])

def run_get_sys_info():
    print("Running get_localHost_sysInfo.py...")
    subprocess.run(["python", "get_localHost_sysInfo.py"])

def run_active_app_and_traffic():
    print("Running activeApp_and_networkTraffic_localHost.py...")
    subprocess.run(["python", "activeApp_and_networkTraffic_localHost.py"])

# Schedule the scripts to run every 2 minutes
schedule.every(2).minutes.do(run_extract_network_config)
schedule.every(2).minutes.do(run_get_sys_info)
schedule.every(2).minutes.do(run_active_app_and_traffic)

print("Scheduler started. Scripts will run every 2 minutes.")

# Keep the scheduler running
while True:
    schedule.run_pending()
    time.sleep(1)
```

How It Works

1. **Script Execution**:
    - The `subprocess.run()` function is used to execute Python scripts (`extract_networkConfig_localHost.py`, `get_localHost_sysInfo.py`, and `activeApp_and_networkTraffic_localHost.py`).
    - Each script is executed independently.
2. **Scheduling**:
    - The `schedule.every(2).minutes.do()` method schedules each function to run every 2 minutes.
3. **Continuous Execution**:
    - The `while True` loop ensures the scheduler continuously checks for pending tasks and executes them when their time arrives.
4. **Output**:
    - Prints messages to indicate when each script is being executed for better visibility.

Requirements

1. Install the `schedule` library if not already installed:

    ```
    pip install schedule
    ```

2. Place all three scripts (`extract_networkConfig_localHost.py`, `get_localHost_sysInfo.py`, and `activeApp_and_networkTraffic_localHost.py`) in the same directory as this scheduler script or adjust file paths accordingly.
3. Run this scheduler script:

    ```
    python scheduler_script.py
    ```

Example Output

When you run the scheduler, you'll see messages like:

```
Scheduler started. Scripts will run every 2 minutes.
Running extract_networkConfig_localHost.py...
Running get_localHost_sysInfo.py...
Running activeApp_and_networkTraffic_localHost.py...
```

This output will repeat every 2 minutes as the scripts are executed.

Customizations

1. **Change Interval**:
    - Adjust the interval by modifying `schedule.every(2).minutes.do()` to a different value (e.g., `.seconds`, `.hours`, `.days`).
2. **Logging**:
    - Add logging functionality to save execution timestamps and results to a file for tracking purposes.
3. **Error Handling**:
    - Wrap each `subprocess.run()` call in a `try-except` block to handle errors gracefully.

Advantages of Using This Script

- Simple and readable scheduling with the `schedule` library.
- No need for external tools like cron or task schedulers.
- Easily customizable for different intervals or additional tasks.

This script ensures all three Python scripts are executed periodically every 2 minutes!

⁂