Windows Task Scheduler with PowerShell - DevOps Automation Guide

==========

Goal:

Automate Task Scheduler using PowerShell to run a Python script (backup.py) daily at 11:00 PM on a Windows VM in Azure.

Prerequisites:

- Python is installed on the VM.
- You have a Python script (e.g., C:\Users\YourUserName\Desktop\backup.py).
- PowerShell is running with Administrator rights.

Step 1: Define Variables in PowerShell

```
$TaskName = "NightlyFileBackup"
$ScriptPath = "C:\Users\YourUserName\Desktop\backup.py"
$PythonPath = "C:\Users\YourUserName\AppData\Local\Microsoft\WindowsApps\PythonSoftwareFoundation.Python.3.11_qbz5n2kfra8p0\python.exe"
$StartInPath = "C:\Users\YourUserName\Desktop"
```

Step 2: Create the Action

$Action = New-ScheduledTaskAction -Execute $PythonPath -Argument "`"$ScriptPath`"" -WorkingDirectory $StartInPath

Step 3: Create the Trigger (Run Daily at 11 PM)

$Trigger = New-ScheduledTaskTrigger -Daily -At 23:00

Step 4: Set Task Principal (User Context)

$User = "$env:USERNAME"

$Principal = New-ScheduledTaskPrincipal -UserId $User -LogonType Interactive -RunLevel Highest

Step 5: Register the Task

Register-ScheduledTask -TaskName $TaskName -Action $Action -Trigger $Trigger -Principal $Principal

Step 6: Test the Task

Start-ScheduledTask -TaskName $TaskName

Extra: View or Remove Task

View details

Get-ScheduledTask -TaskName $TaskName | Format-List

Remove task

```
Unregister-ScheduledTask -TaskName $TaskName -Confirm:$false
```

Real DevOps Use Case in Azure VM:

Example: Automate log collectors or DB backups that run daily in the background using Task Scheduler without external tools like NSSM.

Benefits:

- No need for third-party service managers.
- Simple and native to Windows.
- Easy to modify or remove.