

The perfect gift for readers and writers. [Give the gift of Medium](#)



Stackademic

 Member-only story

11 Python Scripts to Automate Your Daily Tasks

A must-have for every tech guy



Abdur Rahman

[Follow](#)

4 min read · 3 days ago

 297

 4



...



Image Generated using Sora

I'm willing to bet you've been coding in Python long enough to feel confident... until you open your terminal on a Monday morning and think, "*There has to be a better way to automate this mess.*"

Good news: there is.

After 4+ years of building automations, debugging disasters I caused myself, and writing scripts that somehow make me look like a cyber-wizard to non-Python friends, I've hand-picked **11 extremely practical and rare Python scripts** that will make your daily workflow smoother than a freshly installed Linux distro.

Let's get moving.

1. Auto-Detect Wi-Fi Stealers (Know Who's Using Your Internet)

Yes, Python can expose that one neighbor who “accidentally” guessed your Wi-Fi password.

```
import subprocess

def get_connected_devices():
    result = subprocess.check_output("arp -a", shell=True).decode()
    devices = [line for line in result.split('\n') if "dynamic" in line.lower()]
    for d in devices:
        print(d)

get_connected_devices()
```



Why it's gold:

If you work remotely, slow internet is a productivity killer.

This gives you *instant visibility* into all MAC/IPs using your network.

2. Auto-Mute Your Laptop When You're on a Call (Save Yourself the Embarrassment)

This one is criminally underrated.

```
import subprocess
import time

def on_call():
    result = subprocess.check_output("lsof -i | grep zoom", shell=True).decode()
    return bool(result.strip())

def mute_system():
    subprocess.call(["amixer", "-D", "pulse", "sset", "Master", "mute"])

while True:
    if on_call():
        mute_system()
    time.sleep(2)
```



Why it matters:

Your keyboard smashing during meetings? Gone.

Your boss thinking you're typing a novel? Gone.

3. Clipboard Auto-Translator (Copy Text → Auto-Translated Text Appears)

Copy anything. Translation appears automatically.

```
import pyperclip
from googletrans import Translator
import time

translator = Translator()
last = ""

while True:
    text = pyperclip.paste()
    if text != last:
        last = text
        translated = translator.translate(text, dest='en').text
        print("Translated:", translated)
    time.sleep(0.5)
```

When it saves your day:

Whenever you get messages from clients in languages you pretend to understand.

4. Auto-Backup Any File the Moment You Save It

Live-backup like VS Code... but for *any* folder.

```
import time
import shutil
from watchdog.observers import Observer
from watchdog.events import FileSystemEventHandler

class Handler(FileSystemEventHandler):
    def on_modified(self, event):
        if event.is_directory: return
        shutil.copy(event.src_path, f"{event.src_path}.bak")
        print("Backup created:", event.src_path)

observer = Observer()
observer.schedule(Handler(), path="your/folder/path", recursive=True)
observer.start()
```

```
try:  
    while True: time.sleep(1)  
except KeyboardInterrupt:  
    observer.stop()
```

Why it's a secret weapon:

You will *never* lose code again because you forgot Git exists.

5. Automated Browser Profile Switcher (Open Chrome With Different Identities)

For testers, marketers, or that one friend running 5 side hustles.

```
import subprocess  
  
def open_profile(profile_name):  
    subprocess.Popen([  
        "google-chrome",  
        f"--profile-directory={profile_name}"  
    ])  
  
open_profile("Profile 2")
```

Use case:

Debug cookies, test logins, manage accounts.

Zero friction.

6. Auto-Label and Sort Screenshots (Your Desktop Will Finally Look Normal)

```
import os  
import shutil  
from datetime import datetime  
  
src = "/path/to/screenshots"  
  
for f in os.listdir(src):  
    if f.lower().endswith(".png"):  
        date = datetime.fromtimestamp(  
            os.path.getmtime(os.path.join(src, f))  
        ).strftime("%Y-%m-%d")
```

```
folder = os.path.join(src, date)
os.makedirs(folder, exist_ok=True)
shutil.move(os.path.join(src, f), os.path.join(folder, f))
print(f"Moved {f} → {folder}")
```

Why it's underrated:

Screenshots multiply like rabbits.

This script neuters them.

7. Auto-Summarize Any PDF (Perfect for Research & Meetings)

Summarized PDFs = life upgrade.

```
import PyPDF2
from transformers import pipeline

summarizer = pipeline("summarization")

def summarize_pdf(file):
    reader = PyPDF2.PdfReader(file)
    text = "".join(page.extract_text() for page in reader.pages)
    summary = summarizer(text[:3000])[0]['summary_text']
    return summary

print(summarize_pdf("report.pdf"))
```

Why it shines:

A 52-page PDF becomes a 5-line email.

8. Auto-Clean Your Codebase (Remove Dead Imports & Unused Variables)

```
import subprocess

def clean_code():
    subprocess.call(["autoflake", "--in-place", "--remove-unused-variables", "-r"])
    subprocess.call(["isort", "."])
    subprocess.call(["black", "."])
```

```
clean_code()
```

For senior devs:

A clean repo is a happy repo.

This script keeps your Python looking like poetry.

9. Daily System Health Report in Your Email

Every morning → CPU, memory, disk, battery status.

```
import psutil
import smtplib

def report():
    text = f"""
CPU: {psutil.cpu_percent()}%
RAM: {psutil.virtual_memory().percent}%
Disk: {psutil.disk_usage('/').percent}%
Battery: {psutil.sensors_battery().percent}%
"""

    return text

def send_mail(body):
    server = smtplib.SMTP('smtp.gmail.com', 587)
    server.starttls()
    server.login("your@gmail.com", "password")
    server.sendmail("your@gmail.com", "your@gmail.com", body)
    server.quit()

send_mail(report())
```

Why engineers love it:

Peace of mind delivered to your inbox.

10. Auto-Detect Slow Python Code & Suggest Fixes

This one is *chef's kiss*.

```
import ast
import time
```

```

def time_block(code):
    start = time.time()
    exec(code)
    end = time.time()
    return end - start

def analyze(code):
    tree = ast.parse(code)
    long_lines = [n.lineno for n in tree.body if isinstance(n, ast.For)]
    print("Possible slow loops at:", long_lines)

with open("your_script.py") as f:
    code = f.read()

print("Execution Time:", time_block(code))
analyze(code)

```

Exactly the kind of script senior engineers secretly use.

11. Auto-Sync Notes Across Devices Using a Local Folder + Git

A fully private syncing system. No Notion. No Google Docs. No cloud spying.

```

import subprocess
import time

def sync():
    subprocess.call(["git", "add", "."])
    subprocess.call(["git", "commit", "-m", "auto-sync"])
    subprocess.call(["git", "push"])

while True:
    sync()
    time.sleep(300)

```

Why it's brilliant:

Your notes sync every 5 minutes.

You control the repo. No vendor locking, no fees.

Debug Smarter, Faster! Grab your Python Debugging Guide – Click here to download!

99 Python Debugging Tips — A Practical Guide for Developers

Debug Smarter, Not Harder. Bugs are inevitable, wasted hours chasing them don't have to be...

abdurrahman12.gumroad.com

If you enjoyed reading, be sure to give it 50 CLAPS! Follow and don't miss out on any of my future posts — subscribe to my profile for must-read blog updates!

Thanks for reading!

Python

Programming

Data Science

Artificial Intelligence

Technology



Published in Stackademic

69K followers · Last published 1 hour ago

Follow

Stackademic is a learning hub for programmers, devs, coders, and engineers. Our goal is to democratize free coding education for the world.



Written by Abdur Rahman

20K followers · 11 following

Follow

1M+ Views online ✨ Entrepreneur | AI Lover | Developer | Writer.

Responses (4)



What are your thoughts?



Roman
10 hours ago

...

These Snippets are lifesaver for any python developers.

Save Re-usable & Special code snippets at Programmingly.dev for free.

Search across community Snippets, Share & Embed in blogs.

2 [Reply](#)



Ouakrat Avi
7 hours ago

...

I see a python script that need to be executed manually!

How does the automation works ?

You keep the python scrpit run all day ?

[Reply](#)



DURGA SHANKAR AKODIA
9 hours ago

...

Project: Akodia Computational Verification

System: Akodia Sieve (Mod 180 Unified)

Author: Durga Shankar Akodia

Target: Odd Perfect Numbers where $n = 9 \bmod 180$

.....

```
import time
import multiprocessing
import os
import math
import numpy as np
from... more
```

[Reply](#)

[See all responses](#)

More from Abdur Rahman and Stackademic



In Python in Plain English by Abdur Rahman

9 Python Libraries That Make You Look Like a Data Scientist

Even if you're secretly just googling syntax.

★ Dec 7 ⚡ 804 🗣 4



In Stackademic by Mark Henry

The Top 7 MCP Servers Every Dev Needs to Know

How to Make Vibe Coding Not Suck

★ Nov 15 ⚡ 642 🗣 17



In Stackademic by Habibwahid

Stop Writing CRUD APIs: 5 Advanced Spring Boot Patterns...

Basic CRUD gets you hired. These advanced patterns get you promoted. Learn what code...

★ Nov 11 ⚡ 252 🗣 13



In Python in Plain English by Abdur Rahman

6 Python Libraries That Turn Text Into Intelligence

NLP doesn't have to be complicated.

★ Dec 5 ⚡ 411 🗣 4



[See all from Abdur Rahman](#)

[See all from Stackademic](#)

Recommended from Medium



 In AI & Analytics Diaries by Analyst Uttam

Power BI vs Python Dashboards

Last Tuesday, I watched a data scientist spend forty-five minutes explaining to our CFO why...

2d ago  631  26



...



 In Python in Plain English by Abdur Rahman

8 Python Concepts I Finally Understood After Teaching Them

Nothing makes you learn faster than explaining.

 6d ago  412  3



...



 In Data Science Coll... by Marina Wyss - Gratitude...

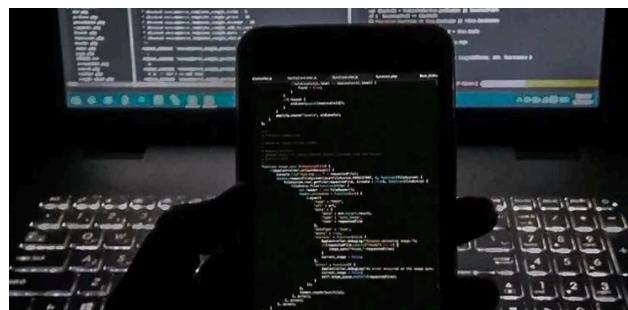
AI Agents: Complete Course

From beginner to intermediate to production.

 Dec 6  1.4K  47



...



 In T3CH by TalhaUlfat

12 Must-Try Python Libraries Every Dev Will Use in 2026 (Don't Get Le...

Subtitle: Future proof your coding stack with the tools shaping automation in the next...

 Dec 7  76



...

AI TOOLS TIER LIST FOR 2026



In Artificial Corner by The PyCoach

The Best AI Tools for 2026

If you're going to learn a new AI tool, make sure it's one of these

Dec 1 2.1K 82



In Towards AI by MohamedAbdelmenem

The AI That Codes So Fast, Developers Are Begging It to Slow...

Eight days after Anthropic released the most intelligent AI coding model ever, Reddit...

Dec 4 489 29

[See more recommendations](#)