Sysadmin Tool

Building a command-line system tool like this will not only strengthen your knowledge of Python's standard library modules (`os`, `platform`, `subprocess`, `shutil`, `glob`, `sys`, `argparse`), but also simulate real-world system administration scripting.

🔧 CW: `sysadmin_tool.py`

✅ Features to implement:

1. **System Monitor**

   - `system-monitor`

   - Optional `--verbose` to display OS info, CPU, memory, etc.

   - Uses: `platform`, `subprocess`, `os`

2. **Backup Directory**

   - `backup-dir <source> <destination>`

   - Copies all contents, preserves structure

   - Uses: `os`, `shutil`, `glob`

3. **Delete Empty Directories**

   - `delete-empty <target-dir>`

   - Optional `--recursive` for depth search

   - Uses: `os`, `os.walk`

4. **List Python Files**

   - `list-py <directory>`

   - Lists all `.py` files (using `glob`)

- Uses: `glob`, `os`

5. **Sys Info Summary**

   - `sys-info`

   - Combines `platform`, `os`, `sys` info

   - Uses: `platform`, `sys`, `os`

🧪 Practice Areas per Module

Module	Usage Examples
`os`	Directory traversal, checking/creating/removing files
`platform`	OS, architecture, system-related metadata
`subprocess`	Shell commands (e.g., CPU stats using `top`, `tasklist`)
`shutil`	Copying, archiving, disk usage
`glob`	File pattern matching (`*.py`, `*/*/`)
`sys`	Access to script arguments, platform info
`argparse`	Full-featured CLI parser for the tool

🖥️ CLI Usage Examples

```
./sysadmin_tool system-monitor --verbose
./sysadmin_tool backup-dir ./myfolder /mnt/backup
./sysadmin_tool delete-empty /var/log
./sysadmin_tool delete-empty /home/user/projects --recursive
./sysadmin_tool list-py /home/user/scripts
./sysadmin_tool sys-info
```