LESSON 11

main.py

```
import requests
r = requests.get(url)
data = r.json()
```

Py Automation

Learn it all @MyWebUniversity



A FREE MYWEBUNIVERSITY COURSE

PYTHON SCRIPTING & OOP

LESSON 11

Scripting and Automation

argparse, subprocess, os and shutil, requests for web APIs, and logging

Command-Line Tools with argparse

cleanup.py

```
import argparse
p = argparse.ArgumentParser(description="Delete old log files")
p.add_argument("folder")
p.add_argument("--days", type=int, default=30, help="age in days")
p.add_argument("--dry-run", action="store_true")
args = p.parse_args()
print(args.folder, args.days, args.dry_run)
```

- argparse turns a script into a real command-line tool, with options, defaults, types and an …
- Your scripts become easy for others to use without reading the code.

01

Working with Files and Folders

cleanup.py

```
import time, shutil
from pathlib import Path
cutoff = time.time() - args.days * 86400
for f in Path(args.folder).glob("*.log"):
    if f.stat().st_mtime < cutoff:
        print("delete", f)
        if not args.dry_run: f.unlink()
shutil.make_archive("logs-backup", "zip", args.folder)
```

- shutil copies, moves and archives files, and pathlib finds them. Together they replace many …
- Here we delete logs older than thirty days, with a dry-run option for safety.

02

Running Commands with subprocess

run.py

```
import subprocess
r = subprocess.run(["df", "-h", "/"], capture_output=True, text=True, check=True)
print(r.stdout)

r = subprocess.run(["ping", "-c", "1", "8.8.8.8"], capture_output=True)
print("online" if r.returncode == 0 else "offline")
```

- subprocess dot run runs any program and captures its output. Pass the command as a list, …
- Avoid shell equals True with user input, to prevent command injection.

03

Web APIs with requests

api.py

```
import requests
r = requests.get("https://api.github.com/repos/python/cpython", timeout=10)
r.raise_for_status()
repo = r.json()
print(repo["full_name"], repo["stargazers_count"])

requests.post(webhook_url, json={"text": "backup done"}, timeout=10)
```

- The requests package makes HTTP easy: requests dot get fetches a URL, and dot json turns …
- Use it to read weather, currency rates, GitHub data, or to post a message to your team chat.

04

logging Instead of print

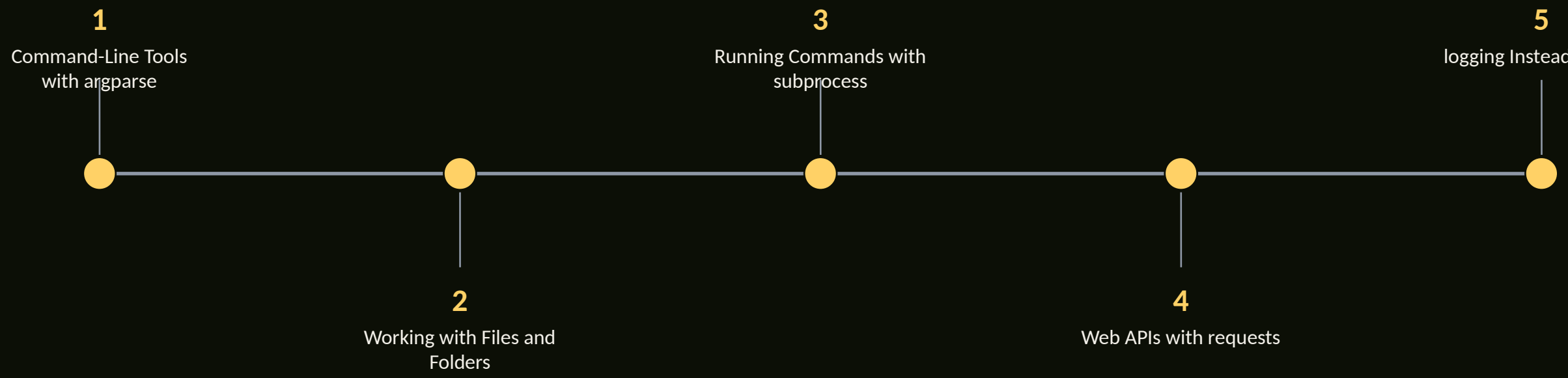
log.py

```
import logging
logging.basicConfig(filename="cleanup.log", level=logging.INFO,
                    format="%(asctime)s %(levelname)s %(message)s")
log = logging.getLogger("cleanup")
log.info("started, folder=%s", args.folder)
log.warning("disk almost full")
log.error("could not delete %s", f)
```

- For scripts that run unattended, use the logging module. It adds time and level to every …
- Levels DEBUG, INFO, WARNING and ERROR let you show more detail while developing and less in …

05

Lesson Map


1
Command-Line Tools with argparse
2
Working with Files and Folders
3
Running Commands with subprocess
4
Web APIs with requests
5
logging Instead of print

Sources and Further Reading

cleanup.py

```python
import argparse
p = argparse.ArgumentParser(description="Delete old log files")
p.add_argument("folder")
p.add_argument("--days", type=int, default=30, help="age in days")
p.add_argument("--dry-run", action="store_true")
args = p.parse_args()
print(args.folder, args.days, args.dry_run)
```

PYTHON SCRIPTING & OOP · 11

run.py

```python
import subprocess
r = subprocess.run(["df", "-h", "/"], capture_output=True, text=True, check=True)
print(r.stdout)

r = subprocess.run(["ping", "-c", "1", "8.8.8.8"], capture_output=True)
print("online" if r.returncode == 0 else "offline")
```

Thank you for watching

MyWebUniversity.com

Free technical education since 2000 — English and Dari

@MyWebUniversity · @MyWebUniversity-Dari