

CHAPTER 08

INPUT/OUTPUT & OS INTERACTIONS

[ACADEMY.CYBERKARTA.COM](https://academy.cyberkarta.com)

Checkpoint

- *Done*
 - membuat program dengan logika kompleks
 - memakai library eksternal
 - menggunakan function
- *What's next?*
 - program realistik: menerima input dari eksternal
 - program realistik: meneruskan output ke program/sistem lain
 - koordinasi antarprogram: integration, pipeline, dll
- Untuk mencapainya, perlu pemahaman terkait program input, output, dan interaksi dalam OS

Bagaimana Program Menerima Input?

Metode	Deskripsi	Implementasi
input interaktif	user input ketika program berjalan	<code>input("masukkan username: ")</code>
arguments	ditulis berbarengan ketika menjalankan program	<code>sys.argv[1]</code> using <code>import sys</code>
baca file	baca data dari file lain	<code>open("log.txt").read()</code>
env variable	sistem punya variable environment khusus, biasa untuk setup config	<code>os.getenv("API_KEY")</code>

Bagaimana Program Menerima Output?

Metode	Deskripsi	Implementasi
console	tampilkan output di shell/terminal, hilang jika shell mati/logoff	<code>print("upload success!")</code>
tulis di file	tulis output sebagai file, sifat data persisting	<code>open("out.txt", "w").write(data)</code>
return value	mengembalikan value untuk dipakai code / sistem lain	<code>return result</code>

Interaksi OS

(Terbuka untuk eksplorasi mandiri)

Operasi	Implementasi
Tampilkan daftar file dalam direktori	<code>os.listdir(".")</code>
Jalankan perintah (seperti user jalankan perintah di shell)	<code>os.system("ls -l")</code> or <code>subprocess.run()</code>
Cek apakah file “config.txt” ada	<code>os.path.exists("config.txt")</code>
Cek direktori saat ini	<code>os.getcwd()</code>
Pindah direktori	<code>os.chdir("/home/user")</code>

Interaksi Antarprogram

Metode	Use Case	Implementasi
Import sebagai module	Reuse code	<code>import utils</code>
Subprocess	Menjalankan script lain dari script yang running saat itu	<code>subprocess.run(["python3", "tool.py"])</code>
Socket/network	komunikasi program via network	<code>socket.socket()</code> (advanced)

- **Pipeline: Ekstraksi log + reporting**
 - Buat 2 file:
 - “[extractor.py](#)” untuk mengambil data sensitif log
 - “[reporter.py](#)” untuk mengirim data ke sistem eksternal via HTTP
 - Disediakan file log “[service_v1.log](#)”
 - “[extractor.py](#)” menjadikan “[service_v1.log](#)” sebagai input, lalu output dikirim ke “[reporter.py](#)”
 - Jangan lupa DRIVE framework
 - **⚠ Kode hanya untuk keperluan edukasi. Jangan dijalankan di mesin yang tidak kamu miliki *authorization*-nya.**