🧭 WEB SCRAPING MASTER PLAN

(Freelance-Ready | Module-1 ka sirf Scraping Part)

> **Goal (1 line):** Client bole: *"Yeh data chahiye"* → tum bole: *"Kab tak aur kaun-se format me?"*

🧠 MENTAL MODEL (VERY IMPORTANT)

Web scraping = **page copy karna nahi** Web scraping = **data pipeline reverse-engineer karna**

So hum **stages** me seekhenge, jaise industry me hota hai.

🟧 STAGE 0 — WEBSITE X-RAY (NON-NEGOTIABLE)

🎯 WHY (kyun pehle?)

- 80% scraping **code likhne se pehle** hoti hai
- Agar yeh nahi aata → har scraper fragile hoga

🧠 WHAT YOU'RE LEARNING

- Data aata **kahaan se**?
- HTML ≠ real data
- Kab browser chahiye, kab nahi

🛠️ TOOLS

- Chrome DevTools (Elements + Network)

🔍 WHAT TO PRACTICE

- Elements tab: DOM samajhna
- Network tab:
  - `Doc` vs `XHR` vs `Fetch`
  - JSON response dhoondhna

- Status codes dekhna (200/403/429)

☑ DONE WHEN

> Tum bina Python likhe bol sako: **“Is site ka data yahan se aa raha hai”**

▨ STAGE 1 — STATIC SCRAPING (TU YAHAN HAI ☑)

🎯 WHY

- Freelance ka **60–70% kaam** yahin se aata hai
- Blogs, listings, directories, news, catalogs

🧠 WHAT YOU’RE LEARNING

- HTML → structured data
- Pagination logic
- Clean extraction

🛠 TOOLS

- `requests`
- `BeautifulSoup`
- `urljoin`

📌 MUST-KNOW CONCEPTS

- GET vs POST
- Headers (User-Agent = survival)
- DOM traversal
- Selector fragility (kya tootega?)

🩹 PRACTICE TASKS

- Multi-page scraping ☑
- Relative → absolute URLs ☑
- Dict → list → JSON ☑

⚠ BACHE HUE (SMALL BUT IMPORTANT)

- HTTP status handling
- try/except

- basic logging
- selector robustness

👉 **1–2 din max**

🟨 STAGE 2 — ASYNC SCRAPING (FREELANCE BOOSTER)

🎯 WHY

Client bole:

> “10,000 pages scrape karo”

Sync code bole:

> “2 ghante lagenge”

Async code bole:

> “5 minute”

🧠 WHAT YOU’RE LEARNING

- Speed = money
- Rate limit ka respect
- Partial failures handle karna

🛠️ TOOLS

- `asyncio`
- `aiohttp`
- `ClientSession`
- `Semaphore`

🧪 PRACTICE TASK

- 20–50 pages concurrently
- Retry failed pages
- Limit requests/sec

☑️ DONE WHEN

> Tum confidently bol sako: **“requests slow kyun hota hai”**

🟧 STAGE 3 — ANTI-SCRAPING AWARENESS (HIDDEN KILLER)

🎯 WHY

Scraper **chal raha tha**, achanak:

- 403
- empty HTML
- random blocks

🧠 WHAT YOU'RE LEARNING

- Sites kaise detect karti hain bots
- Soft block vs hard block

🛠️ CONCEPTS

- Rate-based bans
- Header fingerprinting
- Cookies & sessions

🚫 IMPORTANT

❌ CAPTCHA solve nahi ❌ Illegal hacks nahi

☑ Sirf **diagnosis**

🟨 STAGE 4 — DYNAMIC SCRAPING (HEAVY ARTILLERY)

🎯 WHY

JS-heavy sites:

- Amazon-like
- Infinite scroll
- Data JS se load

🧠 WHAT YOU'RE LEARNING

- Browser automation
- Waits ka importance

🛠 TOOLS

- **Playwright (preferred)**
- Selenium (secondary)

🩹 PRACTICE TASK

- Scroll + wait
- Price scrape
- Network-idle logic

🔥 ADVANCED (OPTIONAL)

Hybrid scraping:

- Browser se cookies
- Async se data

🟧 STAGE 5 — SCRAPING AS A SYSTEM (FREELANCE-READY)

🎯 WHY

Client ka real sawaal:

> “Agar script crash ho jaye to?”

🧠 WHAT YOU’RE LEARNING

- Script ≠ system
- Resume, retry, validate

🛠 CONCEPTS

- Folder structure
- Config-driven scraping
- Logging
- Resume logic

🧰 TOOLS

- `logging`
- `tenacity`
- Pandas (basic)

🧩 WHERE PYDANTIC FITS (IMPORTANT)

❌ HTML parsing phase — **NO** ☑ System / validation phase — **YES**

Pydantic comes when:

- Data DB / API me jaa raha ho
- Contract chahiye

⏱ REALISTIC FREELANCE TIMELINE

Stage	Time
Stage 0	2 days
Stage 1	4–5 days
Stage 2	3 days
Stage 3	1–2 days
Stage 4	3–4 days
Stage 5	2 days

👉 **~2–3 weeks = freelance-ready scraper**
