✅ Python's `__pycache__` vs Browser Cache — Analogy

🔶 Browser Cache:

- Stores things like images, scripts, CSS files from websites.
- So next time you open the same page, your browser **doesn't download everything again**.
- It just **loads the cached data** — faster!

🔶 Python `__pycache__`:

- Stores **compiled bytecode** (i.e., `.pyc` files) from your `.py` files.
- So the next time you run or import that `.py` file, Python **doesn't need to recompile it**.
- It **uses the cached** `.pyc` file — faster!

🔄 Both Save Time by Avoiding Repeated Work

Feature	Browser Cache	Python `__pycache__`
Stores	HTML, CSS, JS, images	Compiled `.pyc` files (bytecode)
Purpose	Avoid re-downloading	Avoid re-compiling
Speed Benefit	Faster page load	Faster script execution/import
Can delete?	Yes	Yes
Auto rebuild?	Yes (when visiting site again)	Yes (when running the code again)

So yes — `__pycache__` **is like a smart memory system**, just like browser cache, that avoids doing the same slow steps every time.