🧩 Project Title: Mutual Fund P&L Estimator (Personal Tracker)

🧠 Problem Statement

Platforms like Groww or Zerodha Coin delay daily P&L updates until late night, even though market prices are available soon after market close (3:30 PM).
For investors who want instant insights, this delay hides how their mutual fund actually performed during the trading day.

The goal is to build a **personal tool** that estimates a mutual fund's **current-day P&L** right after market close — based on the fund's holdings and live stock data — **without waiting for the official NAV.**

🎯 Objective

Develop a **Python-based system** that:

- Takes a **mutual fund's monthly holdings Excel file** (as published by AMCs).
- Accepts user inputs: **investment amount** and **date of investment.**
- Fetches **live stock prices** for each holding using public APIs (like yfinance).
- Calculates:
  - The fund's **estimated current NAV**
  - The user's **current portfolio value**
  - **Profit/Loss (%) and ₹** for the day and overall since investment

---

⚙️ Scope (Version 1 — Lumpsum Investment Only)

✅ Inputs

- Excel file of the fund's holdings (from AMC)
- Investment amount (₹)
- Investment date

🔍 Process

1. Parse Excel file and extract columns:
   - ISIN
   - Security Name
   - Quantity or Market/Fair Value (in Lacs)
   - % to NAV or Portfolio Weight
2. Use **yfinance** to fetch the **current price** for each holding's stock.
3. Recalculate the fund's **estimated NAV** by applying today's stock price changes to the portfolio weights.
4. Fetch **NAV at investment date** (using public mutual fund APIs, e.g., AMFI or Groww's data endpoints).
5. Compute and display:
   - Current estimated NAV
   - P&L since investment

    - Daily change estimate

⚙️ Tech Stack

Layer	Tool	Why
Backend	Python + FastAPI	Fast, lightweight API for calculations and data retrieval
Data Processing	Pandas + yfinance	Easy Excel parsing + live price fetching
Database (Phase 1)	MongoDB	Flexible schema for early dev; perfect for changing Excel formats
Database (Phase 2)	PostgreSQL	For stable, structured version with accurate numeric types
Frontend (optional)	React	For displaying daily P&L and visual charts (optional later)

🧱 Database Plan

🔹 Phase 1 (Now — Early Development)

Use **MongoDB** only to:

- Cache parsed Excel data
- Store user’s investment inputs
- Optionally store recent NAV estimates

👉 This allows fast iteration while you perfect parsing and P&L logic.

🔹 Phase 2 (Later — Stable Version)

Once your logic is finalized, **migrate to PostgreSQL** for:

- Reliable numeric calculations
- Easier time-based queries (daily NAV history)
- Cleaner schema enforcement

Since it’s a single-user personal project, database complexity remains minimal — but PostgreSQL will give better *data integrity and reproducibility.*

🪜 Future Scope (SIP Mode — Phase 2)

When extending to SIP:

- Accept multiple investment dates and amounts
- Calculate **weighted average cost basis**
- Aggregate daily NAV-based values for **true SIP P&L tracking**

Provision for SIP can be kept ready by designing your schema like this:

```
{
  "investment_type": "lumpsum" or "sip",
  "transactions": [
    {"date": "2023-06-15", "amount": 10000},
    {"date": "2023-07-15", "amount": 10000}
  ]
}
```

💬 Prompt for Implementation (to share with your friend)

We’re building a **personal Mutual Fund P&L Estimator** for a *single mutual fund*, not a full portfolio or multi-user app.
The tool should take the mutual fund’s holdings Excel file, the invested amount, and the date of investment — then calculate today’s estimated P&L by fetching live stock prices via yfinance.
We’ll first implement it for **lumpsum investment**, but design the backend so it can later support **SIPs**.
For now, let’s use **MongoDB** to store parsed data and computed NAVs for flexibility. Once our logic is stable, we’ll migrate to **PostgreSQL** for structured, precise storage.
The main focus is accurate early-day P&L calculation before Groww or AMC sites update their NAVs.