You are an expert in Python, Flask, and Quantitative Finance.

Build me a full Flask backend for a quantitative research web app.

🔧 Requirements:

1. Use `yfinance` to fetch historical data for US stocks.

2. Implement a simple \*\*multi-factor stock selection model\*\*:

- Factors: Momentum (past 6-month return), Size (market cap), Value (PE ratio)

- Normalize each factor (z-score or min-max)

- Combine with user-defined weights from POST input

- Rank stocks and pick top N (e.g. top 10)

3. Backtest:

- Buy and hold selected stocks over the backtest period

- Calculate daily portfolio value (equal weight)

- Compute key metrics: total return, annualized return, max drawdown, Sharpe ratio

- Output daily equity curve as a list of {"date": ..., "value": ...}

4. Create a Flask API with:

- POST `/backtest`: receives JSON payload:

```json

{

"tickers": ["AAPL", "MSFT", "GOOG", "TSLA"],

"start\_date": "2019-01-01",

"end\_date": "2023-12-31",

"factors": {

"momentum": 0.4,

"value": 0.3,

"size": 0.3

},

"top\_n": 10

}

```

- Returns JSON:

```json

{

"equity\_curve": [{"date": "...", "value": 100.0}, ...],

"summary": {

"total\_return": 0.55,

"annual\_return": 0.13,

"max\_drawdown": 0.22,

"sharpe\_ratio": 1.45

}

}

```

5. Structure:

- `app.py` main Flask server

- helper functions: `fetch\_data`, `score\_stocks`, `run\_backtest`

- Clear comments for each part

6. Optional: Enable CORS so it can connect with a React frontend later.

Let’s build this backend!