**Assignment Overview**

You have been provided with three CSV files for calendar year 2023:

spot\_with\_signals\_2023.csv

options\_data\_2023.csv

**Your tasks:**

Combine & Enhance Signals

Use our in‑house signal plus any of these indicators on the spot data:

Example Indicators:

(MACD,RSI,ADX,SuperTrend ,Bollinger Bands, EMA crossover, Stochastic, ATR)

Build a composite signal, documenting your voting/weighting logic clearly.

**Options Strategy**

Buy signal: Sell an ATM PUT option.

Sell signal: Sell an ATM CALL option.

Use the nearest expiry in options\_2023.csv for each spot timestamp.

Spot ML Model

Train any regression/classification model (e.g. Linear/Logistic regression, XGBoost, LSTM, clustering) on the spot data + composite signal.

Document your train/test split and evaluation.  
FIND THE BEST COMBINATION.

**Backtest**

Starting capital: ₹200,000

Trade Stop‑loss: 1.5%

Trade Take‑profit: 3% (or close at EOD if not hit)

Force exit: 15:15 local time

Ignore margin interest, transaction costs, and slippage.

**Output**:

equity\_curve.png

drawdown.png

metrics.csv (Sharpe, max drawdown, total return)

trades.csv (entry time, exit time, instrument, strike, premium, P&L)

(example output file is given)

Code & Documentation

Provide modular, well‑commented Python code with this kind of structure:

strategy-backtest/

├── data/

│ ├── spot\_with\_signals\_2023.csv

│ ├── options\_data\_2023.csv

├── indicators.py

├── signal\_engine.py

├── model.py

├── backtest.py

├── utils.py

├── requirements.txt

└── README.md

README.md must cover:

Environment setup & dependencies

Script usage examples

Description of indicators, model, backtest params

How to interpret results

Deliverables:

Full code repo (ZIP or GitHub link)

README.md with instructions

**Results folder:**

equity\_curve.png

drawdown.png

metrics.csv

trades.csv

A 1–2 page write-up summarizing your approach and key insights.