

Data Structures & Processing Flow

Input Data

C Transaction

○ symbol: string

○ strike_price: int

○ expiry_date: date

○ trade_date: date

○ entry_price: float

○ market_lot: int

● option_type: string (CE/PE)

● lots: int (signed)

Initialize from transactions

Processing Structures

C PositionLayers

● combo_key: (symbol, opt_type, strike, expiry)

● long: deque[(price, qty)]

● short: deque[(price, qty)]

C OpenPositions

○ net_lots: int

○ avg_entry_price: float

○ market_lot: int

● combo_key: (symbol, opt_type, strike, expiry)

Calculate daily PnL

Daily PnL Data

C RealisedEntry

○ contract: [symbol, opt_type, strike, expiry, "CLOSED_" + position_type]

○ lots: int

○ pnl: float

○ debug_info: {entry_price_closed, exit_price, market_lot, pnl_calculation}

C UnrealisedEntry

○ contract: [symbol, opt_type, strike, expiry, position_type]

○ lots: int

○ pnl: float

○ debug_info: {entry_price, closing_price, market_lot, pnl_calculation}

C DailyPnL

○ date: string

○ unrealised[]: UnrealisedEntry[]

○ realised[]: RealisedEntry[]

○ total_unrealised_pnl: float

○ total_realized_pnl: float

○ cumulative_total_realized_pnl: float

Aggregate results

Output Data

C MonthSummary

○ month: string

○ month_name: string

○ strategy_type: string

○ num_positions: int

○ total_realized_pnl: float

○ positions[]: Position[]

○ daily_pnl[]: DailyPnL[]