

Robust Logger

Here is a concise AI prompt for generating the NinjaScript code, based on your detailed requirements:

Concise AI Prompt

"Generate a C# class for NinjaTrader 8 named RobustLogger. This class will be a reusable utility for logging strategy events to a single CSV file.

Key Requirements:

1. File Format:

- The logger will create and append to a single CSV file named StrategyLog_{Date}.csv located in the NinjaTrader 8/logs directory.
- The CSV header must be: Timestamp,EventType,EventSubtype,Level,Message,Details

2. CSV Column Formatting:

- **Timestamp:** UTC in ISO 8601 format (yyyy-MM-ddTHH:mm:ss.fffZ).
- **EventType:** A string category: CONNECTION, DATA, ORDER, or SYSTEM.
- **EventSubtype:** A specific string for the event (e.g., CONNECTED, ORDER_FILLED).
- **Level:** A string: INFO, WARN, or ERROR.
- **Message:** A brief, human-readable description.
- **Details:** A single string of semi-colon delimited key-value pairs (e.g., OrderID=123;FillPrice=5100.25;FillSize=2).

3. **Core Methods to Implement:**

- **Connection Logging:** A method to log connection status changes from `OnConnectionStatusUpdate()`.
- **Order Status Logging:** A method to log all order state changes (Submitted, Filled, Cancelled, Rejected) from `OnOrderUpdate()`.
- **Execution Logging & Latency:** A method that takes an Execution object from `OnExecutionUpdate()` and logs fill details. It should also calculate and log total order latency (from a provided `submitTime` to the execution time).
- **Data Stall Logging:** A method to be called from `OnMarketData()` that logs a WARN event if no tick has been received for a configurable number of seconds.
- **General Logging:** Generic methods for logging INFO and ERROR messages.

4. **Crucial Error Handling:**

- All file-writing operations inside the `RobustLogger` class **must** be wrapped in a try-catch block.
- A logging failure **must not** crash the strategy. If writing to the CSV fails, the logger should fall back to writing a critical error message to the main NinjaTrader Log window using `NinjaTrader.Code.Globals.Log()`.

Please provide the complete `RobustLogger.cs` file and a separate example `MyStrategy.cs` file demonstrating how to instantiate the logger and call each of its logging methods from within the appropriate NinjaScript event handlers (`OnStateChange`, `OnConnectionStatusUpdate`, `OnOrderUpdate`, `OnExecutionUpdate`, and `OnMarketData`).