

// NEXURAL AUTOMATION

12-Week Automation Curriculum

A week-by-week plan to go from manual trader to fully-automated system. Each week has clear deliverables. Check them off as you go.

Week 1: Markets Are Auctions

Goal: Build the mental model: price is the output of an auction, not a random walk.

- Summarize auction theory in 1 page of your own words
- Identify 3 value areas on a current chart
- Calculate VWAP $\pm$ 1 std for today's ES session
- Journal: what did buyers vs sellers control today?

Resources: The Asymmetric Investor Ch. 1-3 · Volume Profile: The Insider's Guide

Week 2: The Anatomy of a Strategy

Goal: Decompose any strategy into: signal $\rightarrow$ filter $\rightarrow$ size $\rightarrow$ entry $\rightarrow$ exit $\rightarrow$ risk.

- Write the pseudocode for one strategy you currently trade
- Identify every decision point and when it can fail
- Define your minimum acceptable R:R (should be $\geq 2:1$)
- Write a strategy spec doc (1 page, structured format)

Resources: Nexural /playbooks · The Asymmetric Investor Ch. 10-11

Week 3: From TradingView to Code

Goal: Get your first alert triggering a real action outside TradingView.

- Write a TradingView alert with JSON payload
- Set up a webhook receiver (Node or Python)
- Log every received alert to a file
- Verify the payload arrives within 500ms of the signal

Resources: TradingView Pine Script Docs · github.com/JasonTeixeira/Vibe-Trading

Week 4: Python Basics for Traders / Trading Basics for Coders

Goal: Fill the gap in your weakest skill set — trading logic or Python.

- Complete 3 data manipulation exercises with pandas
- Load a CSV of trade data and calculate expectancy

- Write a function that computes position size given risk %
- Plot 20-day SMA on historical price data

Resources: [Python for Finance \(Yves Hilpisch\)](#) · [pandas docs](#)

Week 5: Reading Order-Flow Data

Goal: Understand delta, footprint, cumulative delta, and how to use them as filters.

- Identify 2 delta divergence examples on a past chart
- Write a function that flags delta exhaustion
- Back-test a simple delta-filtered entry on 20 trades
- Compare results with and without the delta filter

Resources: [Nexural Indicators /features](#) · [The Asymmetric Investor Ch. 8](#)

Week 6: Your First Backtest

Goal: Run a rigorous backtest with realistic assumptions and no look-ahead bias.

- Define in-sample and out-of-sample periods before starting
- Code the strategy in Python/Backtrader/QuantConnect
- Apply realistic slippage (0.5-1 tick) and commission model
- Calculate: net P&L, max drawdown, Sharpe ratio, profit factor

Resources: [github.com/JasonTeixeira/SageQuant](#) · [QuantConnect Lean Docs](#)

Week 7: Risk Math

Goal: Build the risk layer: R:R gate, position sizing, daily loss limit, expectancy.

- Code a `pre_trade_risk_check()` function (see [/automation Risk](#) section)
- Implement Kelly Criterion with 0.25 fraction cap
- Calculate your strategy's expectancy per trade
- Simulate 100 trades at your strategy's win rate — plot equity curve

Resources: [github.com/JasonTeixeira/Nexural_Automation](#) · [The Asymmetric Investor Ch. 12](#)

Week 8: The Broker API

Goal: Connect to a paper trading account and submit your first programmatic order.

- Set up paper trading on Alpaca OR IBKR TWS OR NT8 Simulation
- Submit a limit order via API and confirm execution
- Test order rejection (e.g., submit outside market hours)
- Implement idempotency keys on all order submissions

Resources: [Alpaca Docs](#) · [IBKR API Guide](#) · [github.com/JasonTeixeira/Trade-Engine](#)

Week 9: Webhooks and Order Routing

Goal: Build the full TradingView → webhook → broker pipeline end-to-end.

- Webhook receiver deduplicates by client_order_id
- Payload validation rejects malformed alerts
- Error handling sends alert on rejected order
- Round-trip test: signal → order → fill → log (under 2 seconds)

Resources: github.com/JasonTeixeira/Nexural_Automation · Express/FastAPI webhook patterns

Week 10: Telemetry and Journaling

Goal: Build the observability layer: structured logs, P&L; dashboard, trade journal.

- All events emitted as structured JSON logs
- P&L; updates in real-time (websocket or polling $\leq 5s$)
- Trade journal records: entry, exit, R:R, notes, screenshot
- Alerting: email/SMS on daily loss limit approach (80% threshold)

Resources: [Nexural /journal](#) · Prometheus + Grafana basics

Week 11: Kill Switches and Circuit Breakers

Goal: Harden the system: kill switch, deadman, drawdown breaker, all tested.

- Kill switch tested: flat-all executes within 5 seconds
- Deadman switch tested: flattens if heartbeat absent $> 60s$
- Drawdown circuit breaker pauses signals at -5% drawdown
- All three documented in your incident response playbook

Resources: See [/automation Risk Layer](#) section · Incident Response PDF

Week 12: Going Live with Real Capital

Goal: Structured go-live with minimum viable capital and full monitoring in place.

- Complete all 27 items on the Pre-Flight Checklist PDF
- Trade 2 weeks on paper in live-hours conditions (not just historical)
- Start live with $\leq 25\%$ of intended capital for the first 2 weeks
- Review daily P&L; and compare to backtest expectations
- Commit to no strategy changes for the first 30 live trading days

Resources: Pre-Flight Checklist PDF · nexural.io/automation

Interactive version with progress tracking: nexural.io/automation
