SEMINAR HANDOUT · EIRC OF ICAI · 31 OCTOBER 2026

NSE Trader Alerts with Claude

Install, set up and use an alerts-only intraday toolkit for NSE stocks: scanning, news, entry timing, stop and target planning, position sizing in rupees, and alerts on your phone through Telegram.

Windows | Python | Telegram alerts | Nifty 500 | Alerts only, never trades

CA Ankit Agarwal

Tax and Investment Advisors · Tax and Investment Partners (ARN-337969) · Siliguri

For education only. Not investment advice, and not a SEBI-registered research or advisory service.

Contents

1. What this toolkit is, and is not
2. How the pieces fit together
3. What you need before you start
4. Step 1: Install Python
5. Step 2: Put the pack in place
6. Step 3: Set your trading rules
7. Step 4: Phone alerts through Telegram
8. Step 5: First live test
9. Step 6: Make it run by itself
10. The strategy: opening-range breakout
11. The morning shortlist and follow-ups
12. A trading day, and what you do
13. The alerts you will receive
14. Problems we hit, and the fixes
15. Upkeep, limits and responsible use
16. Command reference

1. What this toolkit is, and is not

It is a set of eight small Python programs ("skills") that run on your own Windows laptop. They read free NSE price data and Indian market news, apply clear rules, and send alerts to your phone. They can also be driven by chatting with Claude Code.

It is adapted from Jake Schincariol's open-source **day-trader-skill** (MIT licence, github.com/Jakeschincariol/day-trader-skill), which was built for US markets and could place orders. This version was changed for NSE and stripped of every part that could trade.

It does

- Scan the Nifty 500 for breakout setups
- Track news on every stock in your list
- Plan stop, target and size in rupees, on NSE tick sizes
- Replay the odds on the stock's last 60 days
- Send stop, target, square-off and news alerts to Telegram
- Push a ranked morning shortlist at 9:50

It does not

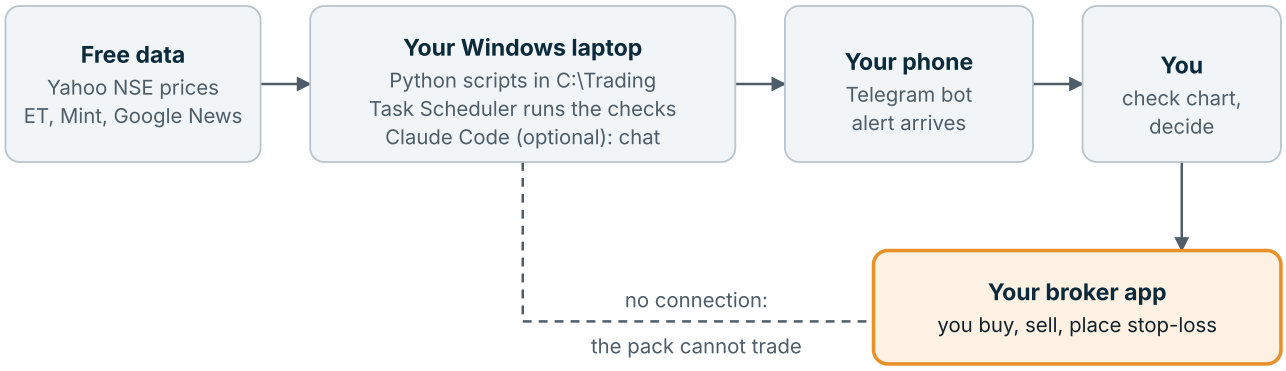
- Connect to any broker or place any order
- Know which trades you actually took
- Read NSE/BSE exchange announcements
- Give live, exchange-grade prices (Yahoo data can lag)
- Predict the market, or replace your own judgement

Read this first

SEBI's July 2024 study found that over 70% of individual intraday equity traders lost money in FY2023. Use this as a discipline and research tool. Watch the alerts for some weeks before risking money, and always place your own stop-loss order in your broker app.

2. How the pieces fit together

Your laptop fetches data and does the maths; your phone receives alerts; only you trade.



The eight skills

Skill	What it does
trader-scanner	Builds your universe from NSE lists (Nifty 50/100/200/500 or all EQ stocks) and scans it for opening-range breakouts or daily setups
trader-news	Headlines per stock, or market-wide for all your stocks, flagging SEBI action, circuits, pledges, block deals and fraud
trader-entry	Watches one stock bar by bar: WAITING, TRIGGERED or EXPIRED
trader-exit	Sets stop, target and square-off time before you buy, rounded to NSE tick sizes
trader-timemachine	Replays the rest of the session 10,000 times from the stock's own last 60 days, and re-tests the rule on each real day
trader-budget	Shares that fit your maximum loss, with brokerage, STT, NSE, SEBI, stamp duty and GST
trader-alerts	Stop, target, % move, time and opening-bell alerts, each ringing once, pushed to Telegram
trader-rank	Runs the whole morning funnel and pushes the top 5 to Telegram; follow-ups until there are 5

3. What you need before you start

Item	Notes
Windows 10 or 11 laptop	Must be on, logged in and awake 9:00 to 15:30 on trading days
Internet connection	Free Yahoo and news feeds; no API key or account needed
Telegram on your phone	For the private alert bot
The pack: `nse-trader-alerts.zip`	Shared with participants after the session
Claude paid plan (optional)	Only for Claude Code, to run everything by chatting
About one hour	For the one-time setup below

STEP 1 · 5 MINUTES

Install Python

1. Go to **python.org/downloads** and click **Download Python 3.x**.
2. Run the installer. On the first screen, **tick "Add python.exe to PATH"**, then click **Install Now**. This is the most important click of the whole setup.
3. Press the Windows key, type `cmd` and press Enter. A black Command Prompt window opens.
4. Check the installation:

```
python --version
```

You should see something like `Python 3.13.15`. On Windows the command is `python`, not `python3`.

STEP 2 · 5 MINUTES

Put the pack in place

Everything lives in `C:\Trading`. The pack goes in `C:\Trading\nse-trader-alerts`; your own settings and alerts go in `C:\Trading\trader`, so updating the pack never touches them.

```
REM 1. make the folder
mkdir C:\Trading
REM 2. unpack the zip (change the path to where your browser saved it)
tar -xf "D:\Downloads\nse-trader-alerts.zip" -C C:\Trading
REM 3. your settings folder and file
mkdir C:\Trading\trader
copy C:\Trading\nse-trader-alerts\trader-budget\config.example.json C:\Trading\trader\config.json
REM 4. check everything works (offline tests)
cd /d C:\Trading
python -m unittest discover -s nse-trader-alerts\tests
```

The last command should end with `Ran 79 tests` and `OK`. The test count also tells you which version you have.

Where is my Downloads folder?

Many PCs keep Downloads on the D: drive. If `tar` says "Failed to open", find the file with `dir "D:\Downloads\nse-trader-alerts*" /O-D`. A second download is often saved as `nse-trader-alerts (1).zip`; use the exact name shown at the top of the list.

STEP 3 · 5 MINUTES

Set your trading rules

```
notepad C:\Trading\trader\config.json
```

Change only the numbers. Keep every quote mark and the comma at the end of each line, then File > Save.

Setting	Example	Meaning
`capital_inr`	200000	Money used for intraday trading (no commas inside the number)
`max_loss_per_trade_inr`	2000	Most one trade may lose, charges included. 1% of capital is a sound default
`max_daily_loss_inr`	4000	The sizer warns when one stop-out would exceed this
`max_position_pct`	100	Most of your capital in one stock (50 allows two positions at once). Never above 100: no leverage
`no_entries_after`	"15:00"	No new entry signals after this IST time
`flat_by`	"15:10"	Square-off reminder, ahead of the broker's auto square-off
`charges`	Zerodha rates	Change to your broker's contract-note rates

Check the file reads correctly with a practice calculation:

```
python -X utf8 nse-trader-alerts\trader-budget\size.py --entry 1434.25 --stop 1421.40
```

The "Your rules" line should show your capital and maximum loss. "is not valid JSON" means a comma or quote was deleted.

Why 1% a trade?

At ₹2,000 on ₹2 lakh, ten losses in a row still leave about 90% of capital. At 5% a trade, two losses cost 10%. The sizer always takes the smaller of "shares your risk allows" and "shares your capital allows", so on tight-stop large caps the capital cap often binds first.

STEP 4 · 10 MINUTES

Phone alerts through Telegram

You create your own private Telegram "bot", a robot account that only messages you.

Part A: create the bot (phone)

1. In Telegram, search **BotFather**, open the one with the blue tick, tap **Start**.
2. Send `/newbot`. Give a name (e.g. *My NSE Alerts*), then a username ending in `bot` that nobody else has.
3. BotFather replies "Done! Congratulations..." with a **token**: a long code with a colon in the middle. Treat it like a password. To see it again: `/mybots` > your bot > **API Token**.

Part B: start your bot and find your chat id

1. Search for **your bot's username**, open it, tap **Start** and send `hi`. A bot cannot message anyone who has not messaged it first.
2. Find your chat id: open **@userinfobot**, tap Start, and note the number after **Id:**.

Part C: put both in the config (laptop)

Copy the token on the laptop through **web.telegram.org** rather than typing it. In `config.json`, fill the `notify` block:

```
"notify": {
  "ntfy_topic": "",
  "ntfy_server": "https://ntfy.sh",
  "telegram_bot_token": "7123456789:AAH...your token...",
  "telegram_chat_id": "987654321"
}
```

Part D: test

```
cd /d C:\Trading
python nse-trader-alerts\trader-alerts\alerts.py notify-test
```

Success: the window shows `notify: pushed test to Telegram` and your phone receives "phone alerts work".

If the test fails

This one-line check reads the token from your config and shows which bot it belongs to and the chat id Telegram sees. Run it from C:\Trading:

```
python -c "import json,urllib.request as u;t=json.load(open('trader/config.json'))['notify']['telegram_bot_token'].strip();print(u.urlopen('https://api.telegram.org/bot'+t+'/getMe').read().decode());print(u.urlopen('https://api.telegram.org/bot'+t+'/getUpdates').read().decode())"
```

Line 1 names your bot. Line 2 contains `"chat":{"id":NUMBER`; that number is your chat id. If line 2 is `"result":[]`, send `hi` to the bot named in line 1 and run it again.

STEP 5 · 10 MINUTES, DURING MARKET HOURS

First live test

Run each from `C:\Trading`:

Command	Good result
`python nse-trader-alerts\trader-alerts\alerts.py clock`	NSE OPEN, session 09:15 to 15:30 IST
`python nse-trader-alerts\trader-scanner\scan.py universe`	"Kept 496 EQ-series equities", writes `trader\universe.txt`
`python nse-trader-alerts\trader-entry\entry.py check RELIANCE`	Today's opening range and WAITING, TRIGGERED or EXPIRED
`python nse-trader-alerts\trader-scanner\scan.py orb`	A list of breakouts (2 to 4 minutes)
`python -X utf8 nse-trader-alerts\trader-news\news.py --market --flagged`	Today's flagged headlines naming your stocks

If `scan.py universe` is refused (NSE sometimes blocks scripts), download the Nifty 500 constituents CSV from niftyindices.com in your browser and run `scan.py universe --file ind_nifty500list.csv`.

STEP 6 · 10 MINUTES

Make it run by itself

Windows Task Scheduler runs four checks on weekdays. Create a small batch file for each with these `echo` lines, pasted one at a time (right-click pastes in Command Prompt). Note the space before the last `>>`: without it Windows drops the final "1" and the batch file breaks.

```
REM price alerts
echo @echo off> C:\Trading\run_alerts.bat
echo cd /d C:\Trading>> C:\Trading\run_alerts.bat
echo python -X utf8 C:\Trading\nse-trader-alerts\trader-alerts\alerts.py check --notify ^>^> C:\T
rading\trader\alerts-log.txt 2^>^&1 >> C:\Trading\run_alerts.bat
REM market-wide news
echo @echo off> C:\Trading\run_news.bat
echo cd /d C:\Trading>> C:\Trading\run_news.bat
echo python -X utf8 C:\Trading\nse-trader-alerts\trader-news\news.py --market --flagged --notify
^>^> C:\Trading\trader\news-log.txt 2^>^&1 >> C:\Trading\run_news.bat
REM morning shortlist and follow-ups
echo @echo off> C:\Trading\run_rank.bat
echo cd /d C:\Trading>> C:\Trading\run_rank.bat
echo python -X utf8 C:\Trading\nse-trader-alerts\trader-rank\rank.py ^>^> C:\Trading\trader\rank-
log.txt 2^>^&1 >> C:\Trading\run_rank.bat
echo @echo off> C:\Trading\run_rank_followup.bat
echo cd /d C:\Trading>> C:\Trading\run_rank_followup.bat
echo python -X utf8 C:\Trading\nse-trader-alerts\trader-rank\rank.py --followup ^>^> C:\Trading\t
rader\rank-log.txt 2^>^&1 >> C:\Trading\run_rank_followup.bat
```

Check a file with `type C:\Trading\run_news.bat`: its last line must end in `2>&1`. Then schedule all four:

```
schtasks /create /tn "NSE Alerts" /tr "C:\Trading\run_alerts.bat" /sc weekly /d MON,TUE,WED,THU,F
RI /st 09:15 /ri 3 /du 0006:20
schtasks /create /tn "NSE News" /tr "C:\Trading\run_news.bat" /sc weekly /d MON,TUE,WED,THU,FRI /
st 09:00 /ri 15 /du 0006:30
schtasks /create /tn "NSE Shortlist" /tr "C:\Trading\run_rank.bat" /sc weekly /d MON,TUE,WED,THU,
FRI /st 09:50
schtasks /create /tn "NSE Shortlist Followup" /tr "C:\Trading\run_rank_followup.bat" /sc weekly /
d MON,TUE,WED,THU,FRI /st 10:00 /ri 10 /du 0005:00
schtasks /query /fo table | findstr NSE
```

Task	Runs	Sends
NSE Alerts	Every 3 min, 9:15 to 15:35	Stop, target and 15:10 square-off for trades you added
NSE News	Every 15 min, 9:00 to 15:30	New flagged headlines on your stocks
NSE Shortlist	9:50	Top 5 breakouts with levels, size and odds
NSE Shortlist Followup	Every 10 min, 10:00 to 15:00	Each later breakout until the day has 5

Test straight away with `schtasks /run /tn "NSE News"`, then `type C:\Trading\trader\news-log.txt`. Pause with `schtasks /change /tn "NSE News" /disable`; remove with `schtasks /delete /tn "NSE News" /f`.

Keep the laptop awake

Settings > System > Power & sleep: "When plugged in, PC goes to sleep after" = Never. A locked screen is fine; signed out, asleep or shut down is not.

10. The strategy: opening-range breakout (ORB), long only

The first 15 minutes after 9:15 is when overnight news and early orders settle into a range. A close above that range shows buyers in control; a fall back into it means the idea was wrong, and you exit with a small, pre-decided loss.

#	Rule	Detail	Reliance, 28 Oct 2026
1	**Opening range**	High and low of the 9:15, 9:20 and 9:25 five-minute bars	High ₹1,194.80, low ₹1,187.00
2	**Entry**	The first 5-minute bar that *closes* above the range high (a wick does not count). Not after 15:00. Long only	9:40 bar closed ₹1,199.60, trigger at 9:45 AM
3	**Stop-loss**	Midpoint of the range, rounded down to NSE's tick for that price. Alternatives: range low, a %, or an ATR multiple	(1,194.80 + 1,187.00) ÷ 2 = ₹1,190.90; risk ₹8.70 a share = 1R
4	**Target**	Entry + 2 × risk (2R). At 2R, being right 1 time in 3 breaks even before charges	1,199.60 + 17.40 = ₹1,217.00
5	**Time exit**	Square off at 15:10 if neither stop nor target is hit	Price fell to ₹1,167: stopped out, −1R
6	**Size**	Shares = max loss ÷ risk a share, trimmed so the stop-out loss including charges fits	Fixed rupee risk whatever the stop distance

NSE tick sizes used for stops and targets (from 15 April 2025)

Price	up to ₹250	₹250 to 1,000	₹1,000 to 5,000	₹5,000 to 10,000	₹10,000 to 20,000	above ₹20,000
Tick	₹0.01	₹0.05	₹0.10	₹0.50	₹1.00	₹5.00

Reading the time machine: the Reliance example

7.2%	40.4%	52.4%	+0.19R
target first	stop first	neither by 15:10	history average, 27 real fires

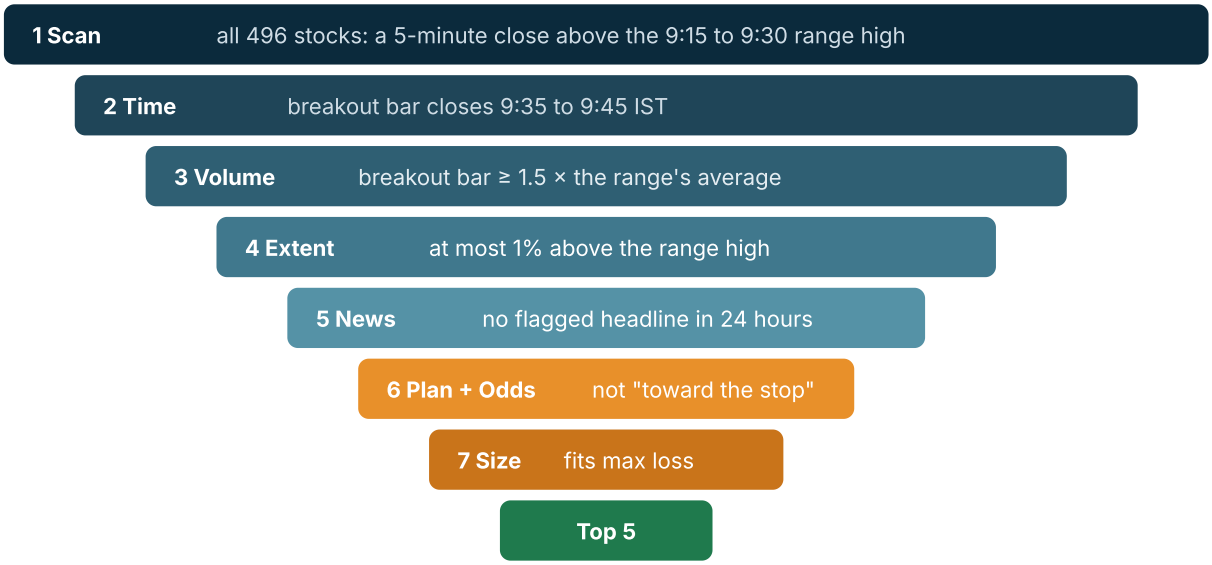
10,000 replays built from Reliance's last 59 sessions said "strongly toward the stop". The history line (the same rule on each of those real days) was slightly positive: 8 targets, 12 stops, 7 time exits. The two views disagreeing is a reminder that this is a guide, not a forecast. The real outcome that day was a stop-out.

Why the strategy can still work

Many breakouts fail. The edge comes from small fixed losses (−1R) against wins twice as large (+2R), plus filters that remove weak setups: early breakouts, strong volume, no unusual news, a supportive chart and market.

11. The morning shortlist and follow-ups

At 9:50 the ranker turns 496 stocks into at most five, each with entry, stop, target, shares and odds. You do the chart judgement.



Each step removes stocks; ranked by the time machine's edge, then volume. Every threshold can be changed under "rank" in config.json.

Follow-ups: if the 9:50 list has fewer than five, the ranker checks every 10 minutes until 15:00. Each new breakout that passes the same rules, and whose current price is still above the range high and within 1% of it, is pushed on its own ("NSE next breakout (3 of 5)"). Each stock is sent at most once a day. Later breakouts tend to be weaker; set `"followup_until": "11:30"` to stop earlier.

Example message (illustrative numbers)

```
NSE morning shortlist Thu 29 Oct 09:50 IST: 2 of 34 breakouts passed.

1. TATASTEEL (Tata Steel Ltd.)  breakout 9:40 AM at ₹152.35 (0.45% over the range, volume 2.1x)
   Entry ₹152.35  Stop ₹151.20  Target ₹154.65  Risk a share ₹1.15
   Shares 820 (₹1,24,927.00, max loss ₹2,000.00 incl. charges)
   Odds: Leans toward the target (target first 38%, stop first 22%, neither 40%)
```

Setting under "rank"	Default	Setting	Default
`window_start` / `window_end`	09:35 / 09:45	`top`	5
`min_volume_x`	1.5	`stop` / `target`	or-mid / 2R
`max_pct_over`	1.0	`followup_until`	15:00

12. A trading day, and what you do

Time (IST)	What arrives or happens	Your action
9:00 onwards	Flagged news on your stocks, every 15 minutes	Note stocks to avoid today
9:50	Morning shortlist: up to 5 stocks with levels	Check each chart on TradingView; decide
10:00 to 15:00	"Next breakout (n of 5)" until there are five	Same: check, decide
When you buy	Nothing automatic: the pack does not know	Place the stop-loss in your broker app, then add the alerts (below)
Any time	Stop or target hit for trades you added	Confirm your broker order filled, or exit / book profit
15:10	Square-off reminder	Close anything still open

Recording a trade you took

The pack has no link to your broker, so stop and target alerts exist only if you add them.

```
REM planned with the pack, bought near the trigger price
python -X utf8 nse-trader-alerts\trader-alerts\alerts.py add --from-plan
REM bought at a different price: re-plan first with your real fill
python -X utf8 nse-trader-alerts\trader-exit\exit.py plan --entry 1201.50 --write
python -X utf8 nse-trader-alerts\trader-alerts\alerts.py add --from-plan
REM a trade you found yourself: your own levels
python -X utf8 nse-trader-alerts\trader-alerts\alerts.py add --symbol TATASTEEL --stop 148.20
python -X utf8 nse-trader-alerts\trader-alerts\alerts.py add --symbol TATASTEEL --target 155.00
python -X utf8 nse-trader-alerts\trader-alerts\alerts.py add --symbol TATASTEEL --at 15:10 --labe
l "square off intraday"
REM see what is watched
python -X utf8 nse-trader-alerts\trader-alerts\alerts.py list
```

Situation	What to do
Two or more trades	Add alerts for each stock; each gets its own
Exited early by hand	Nothing; remaining alerts may ring today; they stop tomorrow
Moved stop up (trailing)	`alerts.py add --symbol X --stop NEW` ; the higher one rings first
Holding overnight	Use the `--symbol` form (no date, watched daily); skip 15:10
Clear everything	`del C:\Trading\trader\alerts.json`

13. The alerts you will receive

Alert	Example	Action
Stop hit	"RELIANCE is ₹1,167.90: at or below your stop ₹1,190.90... Nothing was sold for you."	If your broker SL filled, nothing; otherwise exit now
Target hit	"... at or above your target ₹1,217.00. Book profit in your broker app if you want to."	Book profit, or trail the stop
Square off	"Square off intraday: it is 15:10 IST..."	Close open intraday positions
Shortlist	"NSE morning shortlist ... 2 of 34 breakouts passed"	Check charts, decide
News	"VISL: Vedanta Iron and Steel tumbles 5% to hit lower circuit [circuit]"	Read the story; avoid or rethink that stock

Alerts can lag a few minutes

Prices come from Yahoo's free feed and checks run every 3 minutes, so in a fast fall the price can be well past your stop when the alert arrives. Each alert rings once. The broker stop-loss order is your real protection.

14. Problems we hit during setup, and the fixes

What you see	Cause	Fix
`'python' is not recognized`	PATH box not ticked	Reinstall Python with "Add python.exe to PATH"; open a new window
`tar: Failed to open ...zip`	Wrong folder or name	`dir "D:\Downloads\nse-trader-alerts*" /O-D` and use that name
Tests show an older count than expected	Old zip unpacked again	Unpack the newest file; delete old zips afterwards
No "Congratulations" from BotFather	Username taken or not ending in `bot`	Answer its last question; `/mybots` shows the token
`Telegram failed ... HTTP Error 400`	Bot not started, or wrong chat id	Send `hi` to your bot; re-check the chat id
getUpdates in the browser: `404 Not Found`	Address mistyped	Use the one-line Python check in Step 4
`"result":[]`	Bot has had no message	Send `hi` to the bot named by getMe
`no rules in trader/alerts.json`	No trade added yet, or wrong folder	`cd /d C:\Trading` ; add alerts for a trade
Batch file ends in `2>&`	Missing space before the last `>>`	Recreate with the Step 6 lines
`UnicodeEncodeError ... '\u20b9'` in a log	Windows log encoding has no ₹	Use `python -X utf8` in batch files (fixed in the pack too)
News task sends nothing in a test	Headlines already marked as seen	`del C:\Trading\trader\state\news.json` , run again
Odd characters for ₹ in `type`	Console display only	None; the file and Telegram are correct
`is not valid JSON`	A comma or quote deleted in config	Compare with `config.example.json`
`NSE did not hand over the list`	NSE blocks scripts at times	Download the CSV in a browser; `scan.py universe --file`
`Yahoo did not answer` / HTTP 429	Free feed rate limit	Wait a minute and retry

15. Upkeep, limits and responsible use

How often	What
Daily	Laptop on and awake 9:00 to 15:30; add alerts for any trade you take
Monthly	Refresh the universe: `scan.py universe` (Nifty 500 changes twice a year)
Each December	Add next year's NSE holiday list to the tools
When the pack is updated	Unpack over `C:\Trading`; your `trader` folder is untouched; re-run the tests

- **Not advice.** A rule-based shortlist is not a recommendation. Members advising clients should consider SEBI's Investment Adviser and Research Analyst regulations before sharing signals.
- **Data.** Yahoo and news feeds are free and unofficial; check the live price in your broker app. Exchange filings are not covered.
- **Simulation.** The time machine uses the stock's last 60 days. It cannot see results, news, block deals or circuits coming.
- **Charges and tax.** Defaults match a discount broker's published intraday rates (checked 27 Oct 2026); your contract note is the real figure. Intraday equity results are speculative business income.
- **Not modelled.** Muhurat trading and other special sessions.

16. Command reference

Always start in `C:\Trading` (`cd /d C:\Trading`). Prefix `python -X utf8 nse-trader-alerts\` to each path below.

Command	Does
`trader-alerts\alerts.py clock`	NSE open or closed, next open
`trader-scanner\scan.py universe [--index nifty100]`	Build the stock list
`trader-scanner\scan.py daily`	Pre-market daily setups
`trader-scanner\scan.py orb [--at 10:00]`	Opening-range breakouts so far
`trader-news\news.py --tickers RELIANCE INFY`	News for named stocks
`trader-news\news.py --market --flagged`	Serious headlines for your whole list
`trader-entry\entry.py check SYMBOL --write`	Has the setup triggered?
`trader-entry\entry.py watch SYMBOL --notify`	Wait and push the trigger
`trader-exit\exit.py plan [--entry PRICE] --write`	Stop, target, square-off
`trader-timemachine\timemachine.py run`	Odds from the last 60 days
`trader-budget\size.py`	Shares, rupee risk, charges
`trader-alerts\alerts.py add --from-plan`	Stop, target and 15:10 alerts
`trader-alerts\alerts.py add --symbol X --stop S`	Your own alert levels
`trader-alerts\alerts.py list` / `check --notify`	Show rules / check now
`trader-alerts\alerts.py notify-test`	Test the phone link
`trader-rank\rank.py` / `--followup`	Morning shortlist / later breakouts

Optional: drive it by chatting with Claude Code

With a paid Claude plan, install Claude Code from PowerShell (`irm https://claude.ai/install.ps1 | iex`), start it in `C:\Trading` with `claude` , and add the pack as a local plugin (`/plugin marketplace add C:\Trading\nse-trader-alerts` , then `/plugin install nse-trader-alerts@nse-trader-alerts`). Then ask in plain words: "Scan Nifty 500 for breakouts", "Plan the exit for RELIANCE and size it", "Set the alerts". Tell it once to use `python` rather than `python3` on Windows.

Adapted from day-trader-skill by Jake Schincariol (MIT licence). NSE holiday list as published for 2026; tick sizes per NSE's revision effective 15 April 2025; SEBI intraday study, July 2024. Prepared October 2026.