

Day-Trader-Skill

Elf Skills für Charts, Handelspläne und 10.000 mögliche Verläufe. Dein Einstieg ohne Broker und ohne Echtgeld.



Für Claude Code · Original-Skills auf Englisch

1. Arbeitsumgebung vorbereiten

Claude Code installieren, anmelden und einen leeren Projektordner öffnen. Benötigt: Python 3.8 oder neuer. Im Terminal prüfen:

`claude --version` und `py --version` (Windows) oder `python3 --version` (macOS/Linux).

2. Paket installieren

Starte `claude` im Ordner. Diese Zeilen nacheinander in Claude Code eingeben:

```
/plugin marketplace add Jakeschincariol/day-trader-skill  
/plugin install day-trader-skill@day-trader-skill
```

3. Einrichtung prüfen

Neue Sitzung starten. Unter `/` nach `day-trader-skill` suchen. Beispiel:
`/day-trader-skill:trader-scanner`

Nutze danach den Einrichtungsauftrag auf der Begleitseite. Er prüft Python und Skills und legt die Beispielkonfiguration mit `paper=true` an. Es wird kein Broker verbunden.

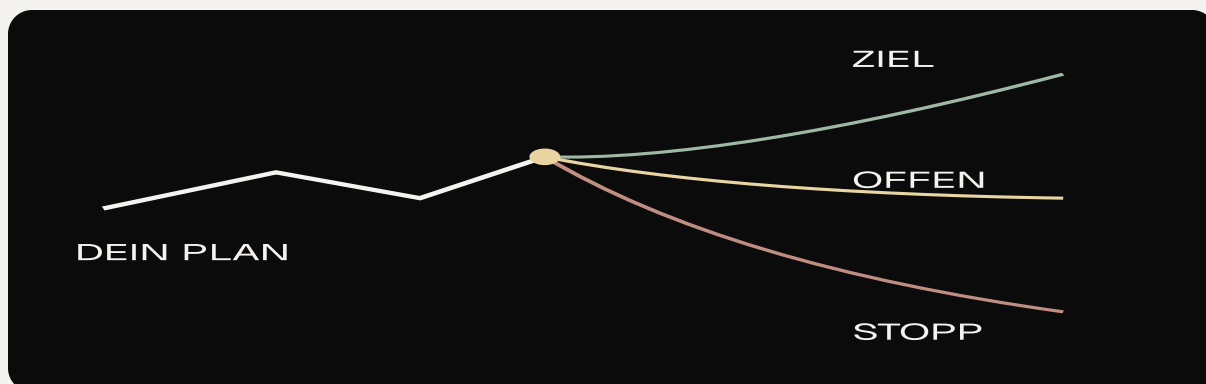
[Einrichtungsauftrag kopieren →](#)

Der Skill-Code ist kostenlos. Claude-Zugang und mögliche Dienste werden separat abgerechnet. Ein Link im normalen Claude-Chat installiert nicht automatisch die Python-Werkzeuge.

[Original-Repository](#) · [Plugin-Hilfe](#)

Die Zeitmaschine. Was die Zahlen sagen.

10.000 Verläufe aus vergangenen Kursbewegungen zeigen mögliche Ausgänge, keinen sicheren Gewinn.



Historisches Beispiel: TSLA

Testdaten vom 19.10.2026. Einstieg 364,18 USD, Stopp 358,32 USD, Ziel 370,04 USD. Seed 19; die vorherigen 59 Sitzungen bilden die Datengrundlage. Keine Kaufempfehlung.

29,4 %

Ziel zuerst

25,5 %

Stopp zuerst

45,1 %

Bis Schluss offen

Der getrennte Rücktest der Regel ergab 33 Auslösungen: 13-mal Ziel zuerst, 15-mal Stopp zuerst, 5-mal keines. Simulation und Rücktest getrennt lesen.

Dein erster Test

Der Testauftrag auf der Begleitseite nutzt die mitgelieferten Dateien und prüft diese Zahlen. Ohne ausgeführten Lauf darf Claude kein Ergebnis behaupten.

[Zeitmaschinen-Test kopieren →](#)

Neue Nachrichten und Kurslücken können anders verlaufen als die historischen Daten. Ein Stopp ist ein Plan, keine garantierte Verlustgrenze.

Vom Chart zum Plan.

Jeder Schritt liest die Ergebnisse des vorherigen.

Scanner → News → Einstieg → Ausstieg

Regel prüfen, Nachrichten lesen und einen konkreten Plan festhalten.

Zeitmaschine → Budget → Risiko

Verläufe simulieren, Stückzahl rechnen und eigene Grenzen prüfen.

Broker

Nur mit separater Verbindung, passendem Paper-Konto und konkreter Freigabe.

Drei weitere Aufgaben

Alerts: Preis- und Zeitmarken bei laufendem Werkzeug prüfen.

Analyst: Unternehmensberichte auswerten.

Pelosi: veröffentlichte Kongress-Handelsmeldungen lesen.

SEC-Abfragen brauchen Name und E-Mail als `sec_user_agent` in der Konfiguration.

Wenn etwas nicht funktioniert

Kein Aufruf? Plugin unter `/plugin` aktivieren und eine neue Sitzung starten.

Kein python3? Unter Windows den vorhandenen Aufruf `py` verwenden.

Keine Kurse? Historische Testdaten verwenden. Fehlende Charts und Datenstand offen ausweisen.

[Alle elf Aufrufe und Scanner-Test öffnen →](#)

Geprüft: 315 Offline-Tests bestanden, Quellstand 3d0f4ef. Kein Claude-Neuinstallationstest und keine Order. Technische Anleitung, keine Anlageberatung.

Komm gern in meine Gratis-Community.

[Gratis-Community beitreten →](#)