CLAUDE UND CODEX

Zwei Modelle, ein Projekt

OpenAI hat ein Plugin gebaut, das in Claude Code läuft. Claude baut, Codex prüft gegen

4 Befehle
einmal einrichten

ChatGPT Free
reicht aus

32.711
Sterne auf GitHub

Eine KI, die ihren eigenen Code prüft, ist ungefähr so verlässlich wie ein Schüler, der seine eigene Klausur korrigiert.
Deshalb holt das Plugin ein zweites Modell dazu. Codex schaut sich an, was Claude gebaut hat, schickt seine Kritik zurück, und Claude arbeitet sie ein.

01 **Einrichten in vier Zeilen**

Marketplace einhängen, Plugin installieren, neu laden, Setup. Danach ist es dauerhaft da.

02 **Der Befehl für den Alltag**

Slash codex review, wenn du mit einem Stück Arbeit fertig bist. Das ist der eine Befehl, den du wirklich brauchst.

03 **Und wenn es härter sein soll**

Es gibt eine Prüfung, die nicht den Code angreift, sondern deine Entwurfsentscheidungen. Die steht auf Seite 4.

Repo openai/codex-plugin-cc, offiziell von OpenAI, geprüft am 3. September 2026. 32.711 Sterne, letzter Push 8. Juli 2026.

SCHRITT 1

Einmal einrichten

Vier Zeilen in Claude Code, danach nie wieder

IN CLAUDE CODE EINTIPPEN

```
/plugin marketplace add openai/codex-plugin-cc
/plugin install codex@openai-codex
/reload-plugins
/codex:setup
```

01

Was du brauchst

Ein ChatGPT-Konto, auch das kostenlose. In den Voraussetzungen steht wörtlich, dass ein Abo einschließlich Free reicht, alternativ ein OpenAI-Schlüssel. Dazu Node.js in Version 18.18 oder neuer.

02

Warum vier und nicht drei

Zwischen Installieren und Setup muss einmal neu geladen werden, sonst kennt Claude die Befehle noch nicht. Das wird oft unterschlagen.

03

Was das Setup macht

Es prüft, ob die Codex-Kommandozeile da ist und ob du angemeldet bist. Danach stehen dir alle Slash-Befehle zur Verfügung.

Voraussetzungen laut README: ChatGPT subscription (incl. Free) or OpenAI API key, Node.js 18.18+.

DER ALLTAG

Slash
codex review

Der eine Befehl, den du wirklich brauchst

NACH JEDEM FERTIGEN STÜCK

```
/codex:review
```

01 Wann du ihn tippst

Nicht nach jeder Zeile, sondern wenn ein Stück Arbeit fertig ist. Eine Funktion, eine Seite, ein Ablauf. Vorher lohnt es nicht.

02 Was passiert

Codex liest sich den geänderten Code durch und schickt eine Liste zurück, also was fehlt, was schiefgehen kann und was zu viel ist. Claude bekommt die Liste direkt und arbeitet sie ab.

03 Warum das mehr bringt als nachfragen

Wenn du Claude selbst bittest, seinen Code zu prüfen, bewertet dasselbe Modell dieselbe Entscheidung noch einmal. Codex hat den Entwurf nicht gemacht und sieht deshalb andere Sachen.

Die Nutzung zählt auf deine Codex-Limits. Das Plugin geht über deine lokale Codex-Installation und deine bestehende Anmeldung.

WENN ES HART SEIN SOLL

Die zwei starken Befehle

Einer greift deine Entscheidungen an, der andere übernimmt ganz

KRITIK AN DEN ENTWURFSENTSCHEIDUNGEN

```
/codex:adversarial-review
```

01

Was das anders macht

Die normale Prüfung schaut auf den Code. Diese hier geht an die Entscheidung dahinter, also warum du es überhaupt so gebaut hast. Nimm sie, bevor du auf einer Architektur weiterbaust.

EINE GANZE AUFGABE ABGEBEN

```
/codex:rescue
```

02

Wofür du das nimmst

Wenn Claude sich festgefahren hat. Der Befehl gibt die Aufgabe komplett an Codex ab, statt nur eine Meinung einzuholen.

03

Der ehrliche Hinweis

Beide kosten mehr Durchläufe und zählen auf deine Codex-Limits. Für den Alltag reicht die normale Prüfung.

Befehlsliste aus dem Repo openai/codex-plugin-cc, geprüft am 3. September 2026.

WENN ES LÄNGER DAUERT

Aufgaben im Blick

Vier kleine Befehle für alles, was im Hintergrund läuft

/codex:status	zeigt, was gerade läuft und wie weit es ist
/codex:result	holt das Ergebnis, wenn ein Durchlauf fertig ist
/codex:cancel	bricht einen Durchlauf ab, der zu lange braucht
/codex:transfer	übergibt eine laufende Aufgabe an die andere Seite

01

Warum es die überhaupt gibt

Eine Prüfung über viele Dateien dauert. Statt zu warten arbeitest du weiter und holst dir das Ergebnis, wenn es da ist.

02

Der praktische Ablauf

Prüfung starten, weiterbauen, mit status nachsehen, mit result einsammeln. Wenn es hängt, abbrechen und kleiner nochmal starten.

Alle Befehle aus dem offiziellen Repo, geprüft am 3. September 2026.

NÄCHSTER SCHRITT

Direkt umsetzen

Fang mit einem Stück Arbeit an, nicht mit dem ganzen Projekt

01 **Die vier Zeilen von Seite 2 eintippen**

Zwei Minuten. Wenn slash codex setup ohne Fehler durchläuft, steht alles.

02 **Ein fertiges Stück prüfen lassen**

Nimm etwas, das du gerade gebaut hast, und tipp slash codex review. Lies die Kritik einmal selbst durch, bevor Claude sie einbaut.

03 **Eine Sache gegenprüfen**

Schau dir einen Punkt aus der Kritik genau an und entscheide, ob du ihm zustimmst. Codex liegt nicht immer richtig, und das ist auch nicht der Punkt. Der Punkt ist die zweite Meinung.

Die Nutzung zählt auf deine Codex-Limits.
Beim kostenlosen Konto sind die schnell erreicht. Deshalb gezielt prüfen lassen und nicht nach jeder Zeile.

Mehr Anleitungen in der Short AI Bibliothek unter shortmedia.agency.