

CODEX VS. CLAUDE CODE: WELCHES CODING-AGENT PASST ZU DIR?

Du schwankst zwischen Codex und Claude Code? Dieser Guide zeigt dir die Unterschiede bei Setup, Arbeitsweise, Integrationen und typischen Coding-Workflows und gibt dir eine klare Entscheidungshilfe.

2 PROMPTS

TOOL: CODEX + CLAUDE CODE

STAND 2026-09-07

DU BRAUCHST EINEN CODING-AGENTEN, NICHT NOCH EINEN CHAT

Beide Tools sitzen direkt neben deinem Code und können Aufgaben über natürliche Sprache bearbeiten. **Codex** kommt von OpenAI und läuft als Coding-Agent lokal auf deinem Rechner; **Claude Code** kommt von Anthropic und ist ein agentisches Coding-Tool, das deinen Codebestand versteht und Git-Workflows unterstützt.

Der Unterschied liegt deshalb weniger in der Frage, ob beide programmieren können, sondern darin, wie du deinen Arbeitsalltag aufbauen willst.

WAS DU BRAUCHST

Ein bestehendes Projektverzeichnis, Git für saubere Änderungen und eine aktuelle Installation von Codex oder Claude Code. Für die Anmeldung kannst du Codex mit ChatGPT oder einem API-Key nutzen. Claude Code unterstützt unter anderem Anthropic Console, Claude Pro/Max und Enterprise-Integrationen.

CODEX VS. CLAUDE CODE AUF EINEN BLICK

	CODEX	CLAUDE CODE
Anbieter	OpenAI	Anthropic
Kernumgebung	Lokaler Coding-Agent; CLI plus IDE-, Desktop- und Web-Varianten	Agentisches Coding-Tool; Terminal, IDE und GitHub
MCP	MCP-Server können konfiguriert werden	MCP-Server können eingebunden werden
Arbeitsweise	Agentische Coding- und Automatisierungs-Workflows	Direkte Repository- und CLI-Workflows
Ökosystem	OpenAI / ChatGPT	Anthropic / Claude

01

CODEX EINRICHTEN

Wenn du Codex ausprobieren willst oder bereits ChatGPT für deine Entwicklungsarbeit nutzt.

Installiere Codex und starte es direkt in deinem Projektverzeichnis. Auf macOS oder Linux kannst du den offiziellen Installer verwenden; alternativ sind npm und Homebrew verfügbar. Danach startest du mit **codex**.

Für die Anmeldung kannst du dich mit ChatGPT anmelden oder Codex mit einem API-Key betreiben.

[Offizielle Codex-Dokumentation und Repository](#)

ERSTER PROMPT

```
Analysiere dieses Repository. Erkläre mir zuerst die Architektur, nenne die wichtigsten  
Einstiegspunkte und schlage danach drei konkrete Verbesserungen vor. Ändere noch nichts.
```

01

CLAUDE CODE EINRICHTEN

Wenn du Claude Code ausprobieren willst oder bereits Claude als Coding-Partner nutzt.

Installiere Claude Code nach der offiziellen Anleitung, wechsele in dein Projektverzeichnis und starte **claude**. Anthropic dokumentiert mehrere Installationswege und unterstützt unter anderem macOS, Linux und Windows-Setups.

Zur Authentifizierung stehen unter anderem Anthropic Console, Claude Pro/Max sowie Enterprise-Integrationen zur Verfügung.

[Offizielle Claude-Code-Setup-Anleitung](#)

ERSTER PROMPT

```
Analysiere dieses Repository. Erkläre mir zuerst die Architektur, nenne die wichtigsten  
Einstiegspunkte und schlage danach drei konkrete Verbesserungen vor. Ändere noch nichts.
```

WO CODEX DIE BESSERE WAHL IST

Nutze Codex, wenn du deine Coding-Arbeit eng mit dem OpenAI-Ökosystem verbinden willst oder agentische Aufgaben und Automatisierung in einem Workflow bündeln möchtest. Die aktuelle Codex-Dokumentation nennt unter anderem das Verstehen großer Codebasen, langfristige Ziele, Security-Scans und Code-Reviews als Anwendungsfälle.

Codex ist außerdem interessant, wenn du nicht nur in der CLI arbeiten willst: Die offizielle Dokumentation beschreibt auch IDE-, Desktop- und Web-Varianten.

DER EHRLICHE TEIL

Es gibt hier keinen sinnvollen Gewinner für jedes Projekt. Wenn du bereits tief in einem der beiden Ökosysteme arbeitest, ist die Integration oft wichtiger als ein pauschaler Modellvergleich. Für die eigentliche Entscheidung solltest du deshalb denselben echten Task in beiden Tools laufen lassen: Repository verstehen, Änderung planen, Code schreiben, Tests ausführen und den Diff prüfen.

WO CLAUDE CODE DIE BESSERE WAHL IST

Nutze Claude Code, wenn du einen stark repository- und CLI-zentrierten Agenten willst. Claude Code bietet eine CLI mit interaktivem Modus, nicht-interaktiver Ausgabe, Session-Fortsetzung, MCP-Konfiguration und steuerbaren Berechtigungen.

Wenn dein Workflow stark über MCP-Server, CLI-Automatisierung oder GitHub läuft, ist Claude Code besonders interessant.

01

MACH DEN 30-MINUTEN-VERGLEICH

Wenn du nicht nach Bauchgefühl entscheiden willst.

Nimm einen echten Task aus deinem aktuellen Projekt. Gib beiden Tools denselben Auftrag und vergleiche nicht nur den fertigen Code.

1. Repository analysieren lassen.
2. Änderungsplan erstellen lassen.
3. Implementierung durchführen lassen.
4. Tests und Linting ausführen lassen.
5. Diff und Erklärungen prüfen.

Bewerte danach vier Dinge: Qualität der Änderungen, Anzahl deiner Korrekturen, Geschwindigkeit bis zum brauchbaren Ergebnis und wie gut der Agent deinen gewünschten Workflow respektiert.

HÄUFIGE FRAGEN

KANN ICH BEIDE TOOLS IM SELBEN PROJEKT VERWENDEN?

Ja. Beide arbeiten mit lokalen Projektverzeichnissen. Du kannst denselben Git-Workflow nutzen und die Tools für unterschiedliche Aufgaben einsetzen.

UNTERSTÜTZEN BEIDE MCP?

Ja. Beide können mit MCP-Servern arbeiten. Anthropic dokumentiert MCP ausdrücklich für Claude Code; OpenAI dokumentiert MCP-Server für Codex CLI.

WELCHES TOOL SOLLTE ICH ZUERST TESTEN?

Nimm das Tool, dessen Ökosystem du bereits verwendest. Danach teste denselben echten Task im anderen Tool. Das liefert dir mehr als ein allgemeiner Benchmark.

WENN DEIN VERGLEICH SCHIEFGEHT

Vergleiche nicht zwei unterschiedliche Aufgaben und bewerte danach das Tool. Verwende dasselbe Repository, denselben Branch, dieselben Tests und möglichst denselben Prompt. Prüfe außerdem den finalen Git-Diff selbst, statt nur die Erklärung des Agenten zu bewerten.

MEINE ENTSCHEIDUNGSHILFE

Codex: Nimm es zuerst, wenn du OpenAI/ChatGPT bereits als Zentrum deines Entwickler-Workflows nutzt und einen Agenten für umfangreiche, agentische Coding- und Automatisierungsaufgaben suchst.

Claude Code: Nimm es zuerst, wenn dein Alltag stark repositoryzentriert ist und du eine direkt steuerbare CLI mit Git-, MCP- und Automatisierungs-Workflows willst.

Wenn du unentschlossen bist: Installiere beide. Der beste Test ist kein Benchmark, sondern dein eigenes Repository.

HINWEIS

Dieser Guide dient zu Informations- und Bildungszwecken. Alle Inhalte spiegeln den Stand bei Erstellung wider — Oberflächen, Funktionen und Preise der genannten Tools können sich ändern. Prüfe Ergebnisse von KI-Tools immer selbst nach. @GPTMarlon übernimmt keine Haftung für Entscheidungen auf Grundlage dieser Inhalte. Dieser Guide stellt keine Anlage-, Steuer-, Rechts- oder Gesundheitsberatung dar. Lass Ergebnisse bei Bedarf fachlich prüfen.

Dieser Guide stellt keine Anlage-, Steuer-, Rechts- oder Gesundheitsberatung dar. Lass Ergebnisse bei Bedarf fachlich prüfen.