# Vergleich: Claude Code (Anthropic) vs. Codex CLI (OpenAI)

## Vergleichende Tabelle

| **Merkmal** | **Claude Code (Anthropic)** | **Codex CLI (OpenAI)** |
| --- | --- | --- |
| **API-Preis (pro 1.000 Tokens)** | *Claude* bietet mehrere Modelle: z.B. Haiku 3.5 für günstige Tasks (≈ $0,0008 / $0,004 input/output pro 1k)[[1]](https://www.anthropic.com/pricing#:~:text=%2A%20), Sonnet 4 als Allrounder (≈ $0,003 / $0,015)[[2][3]](https://www.anthropic.com/pricing#:~:text=), Opus 4.1 für höchste Qualität (≈ $0,015 / $0,075)[[4]](https://www.anthropic.com/pricing#:~:text=Most%20intelligent%20model%20for%20complex,tasks)[[5]](https://www.anthropic.com/pricing#:~:text=Write%20%2418). <br>*(Hinweis: Preise in $ pro Million Tokens.)* | *OpenAI*’s Codex-CLI nutzt primär GPT-Modelle. Standard-Modell **codex-mini-latest** (tuned o4-mini) kostet ~$0,0015/$0,006[[6]](https://help.openai.com/en/articles/11381614-codex-cli-and-sign-in-with-chatgpt#:~:text=,6%20%2F%20M%20output%20tokens). Neuere Modelle: **GPT-5** (input $0,00125, output $0,01)[[7]](https://openai.com/api/pricing/#:~:text=Input%3A%20%241), **GPT-5 mini** (input $0,00025, output $0,002)[[8]](https://openai.com/api/pricing/#:~:text=Input%3A%20%240) usw. Auch **GPT-4.1** kann per API genutzt werden (feiner Code-Modell, input $0,003, output $0,012[[9]](https://openai.com/index/gpt-4-1/#:~:text=rewriting%20an%20entire%20file,latency%20of%20full%20file%20rewrites)). |
| **Verfügbare Modelle** | Claude Code nutzt Anthropics Claude-Familie: **Claude Opus 4.1, Opus 4, Sonnet 4, Sonnet 3.7, Haiku 3.5**[[10]](https://support.anthropic.com/en/articles/11940350-claude-code-model-configuration#:~:text=%2A%20Opus%204.1%2C%20%60claude) (je nach Account-Stufe; Pro-Plan z.B. ohne Opus). Alle sind speziell für vielfältige Aufgaben trainiert. | Codex CLI ist ein Agent, der *jede* OpenAI-Modelle aufrufen kann[[11]](https://help.openai.com/en/articles/11096431-openai-codex-cli-getting-started#:~:text=By%20default%20Codex%20targets%20GPT,thought%20summaries%20in%20the%20API). Standard ist GPT-5, unterstützt werden aber **GPT-5 mini/nano, GPT-4.1, GPT-4o (Turbo), GPT-4, GPT-3.5 (inkl. o1/o3)** u.v.m. (über -m Flag auswählbar). Der CLI-Code ist Open-Source[[12]](https://openai.com/index/introducing-o3-and-o4-mini/#:~:text=We%E2%80%99re%20also%20sharing%20a%20new,1%20%E2%81%A0). |
| **Kontextlänge (max)** | Standard-**Claude 4** (z.B. Sonnet 4): konfigurierbarer Kontext bis **200.000 Tokens**[[13]](https://docs.anthropic.com/en/docs/build-with-claude/context-windows#:~:text=,generating%20new%20output%20from%20Claude). Neu: Sonnet 4 *Beta* unterstützt bis **1.000.000 Tokens** (gegen Aufpreis)[[14]](https://every.to/vibe-check/vibe-check-claude-sonnet-4-now-has-a-1-million-token-context-window#:~:text=Today%2C%20Anthropic%20is%20releasing%20a,Potter%20books%20in%20each%20prompt). Ältere Claude-3-Modelle (Haiku) haben geringere Limits (~100k–200k). | **GPT-5 (Text+Vision)** hat bis **400.000 Tokens** Kontext (davon max. 128k Output)[[15]](https://openai.com/gpt-5/#:~:text=GPT,opens%20in%20a%20new%20window). GPT-4.1 unterstützt bis **32.768 Tokens**[[16]](https://openai.com/index/gpt-4-1/#:~:text=prompting%20guide%20%E2%81%A0%20,latency%20of%20full%20file%20rewrites). Ältere GPT-3.5/4-Modelle haben 16k–128k (z.B. GPT-4o 32k, GPT-4.5 wohl 128k). Codex CLI kann alle diese nutzen. |
| **Tools/SDKs** | Offizielle **Anthropic SDKs** (Python, TypeScript) für Claude Code. Es gibt auch Multi-Agent-Plattformen wie **Claude-Flow** (Open-Source-CLI, „Swarm/Hive-Mind“-Orchestrierung)[[17]](https://github.com/ruvnet/claude-flow#:~:text=Overview). Claude Code selbst bietet Funktionen wie Tool-Aufrufe (z.B. Code-Ausführung, Websuche). | Codex CLI ist ein eigenständiges Node/NPM-Tool (@openai/codex CLI)[[12]](https://openai.com/index/introducing-o3-and-o4-mini/#:~:text=We%E2%80%99re%20also%20sharing%20a%20new,1%20%E2%81%A0). Es wird lokal ausgeführt und kann mit ChatGPT-Login verbunden werden. Durch die **Responses API** können auch *Ergebniszusammenfassungen* („chain-of-thought“) genutzt werden[[11]](https://help.openai.com/en/articles/11096431-openai-codex-cli-getting-started#:~:text=By%20default%20Codex%20targets%20GPT,thought%20summaries%20in%20the%20API). Tools wie lokales Dateisystem, Screenshots oder Websuche sind geplant/eingebaut. |
| **Vor-/Nachteile (Python/Flutter)** | **Vorteile:** Sehr gute Verständnis- und Generierungsfähigkeiten über viele Sprachen (u.a. Python) hinweg[[18]](https://medium.com/@averageguymedianow/claude-code-in-your-workflow-the-good-the-bad-and-the-ugly-304e4f5a9dc9#:~:text=Claude%20is%20an%20AI%20assistant,simple%20completion%20tools%2C%20Claude%20can), inkl. erklärender Kommentare und Refactoring. Starke Sicherheit/Failsafes (Claude zögert eher). **Nachteile:** Kontextlimit (außer Sonnet-Beta 1M), höhere Kosten bei Top-Modellen (Opus). Claude-Code-Agent ist neu, GUI-Integration begrenzt. | **Vorteile:** Zugriff auf **GPT-5** – laut OpenAI “bestes Modell für Codierung”[[19]](https://openai.com/gpt-5/#:~:text=GPT%E2%80%915%20produces%20high,%E2%80%98verbosity%E2%80%99%20parameter%20in%20the%20API). In internen Tests erzeugt GPT-4.1 herausragende Frontend/UI-Code (80 % der Zeit besser als GPT-4o)[[20]](https://openai.com/index/gpt-4-1/#:~:text=GPT%E2%80%914,of%20the%20time), GPT-5 dürfte das übertreffen. Preis/Leistung kann günstiger sein (GPT-5 mini sehr billig für einfache Tasks). Codex CLI ist flexibel und Open-Source. **Nachteile:** Abhängigkeit von OpenAI-Ökosystem (erfordert ChatGPT-Login). CLI-Modus evtl. ungewohnt. |
| **Besonderheiten/Einschränkungen** | Claude verfügt über „Extended Thinking“, Prompt-Caching u.v.m. (z.B. Websuche-Tool, Python-Ausführung)[[21]](https://www.anthropic.com/pricing#:~:text=%2A%20). Größte Stärke: *selbstorganisierende Ketten von Tools*. Enterprise-Kunden bekommen Priority-Service und SSO. Claude Code selbst ist noch jung (GA Mai 2025). | GPT-5/CLI unterstützt **Multi-Modales Input** (Screenshots, Skizzen) in zukünftigen Versionen[[12]](https://openai.com/index/introducing-o3-and-o4-mini/#:~:text=We%E2%80%99re%20also%20sharing%20a%20new,1%20%E2%81%A0). Codex CLI ist vollständig quelloffen und erhält exklusive Förderung (Grants). Standardmodell codex-mini-latest ist stark auf Interaktivität optimiert[[6]](https://help.openai.com/en/articles/11381614-codex-cli-and-sign-in-with-chatgpt#:~:text=,6%20%2F%20M%20output%20tokens). GPT-5 in Codex CLI wird über ChatGPT-Zugang verfügbar; kostenlose API-Keys können limitierte Zugriffe haben. |

## Grafische Darstellung

**Grafik:** Kontextlänge vs. Token-Kosten. In der folgenden Darstellung sind einige Modell-Kombinationen eingezeichnet: Claude Haiku 3.5, Sonnet 4, Opus 4.1 sowie GPT-5/Familie (inkl. mini) und Codex-mini. Man sieht, dass GPT-5-Varianten sehr große Kontexte (400k Tokens) zu moderaten Preisen bieten[[15]](https://openai.com/gpt-5/#:~:text=GPT,opens%20in%20a%20new%20window)[[7]](https://openai.com/api/pricing/#:~:text=Input%3A%20%241), während Claude-Modelle (z.B. Opus 4.1) bei ~200k Kontext deutlich teurer sind[[4]](https://www.anthropic.com/pricing#:~:text=Most%20intelligent%20model%20for%20complex,tasks). Claude Haiku ist extrem günstig (unterstes Preisspektrum)[[1]](https://www.anthropic.com/pricing#:~:text=%2A%20). *(Eigene Darstellung, basierend auf den zitierten Preisen und Kontextdaten.)*

*Abbildung: Vergleich von maximaler Kontextlänge und kombinierten Input/Output-Kosten (pro 1k Tokens) ausgewählter Claude-Modelle (Haiku, Sonnet, Opus) versus OpenAI-Modelle (GPT-5/mini, GPT-4.1, Codex-mini)*[*[4]*](https://www.anthropic.com/pricing#:~:text=Most%20intelligent%20model%20for%20complex,tasks)[*[15]*](https://openai.com/gpt-5/#:~:text=GPT,opens%20in%20a%20new%20window)*.*

## Kostenaufschlüsselung typischer Aufgaben

* **Python-Funktion schreiben:** Beispiel-Prompt mit 20 Tokens, generierter Code ≈ 300 Tokens. <br>
* *Claude Sonnet 4:* 20 Tokens Input → ~0,00006$ (20/1000·$3) + 300 Tokens Output → ~0,0045$ (300/1000·$15) = **~0,0046 $**[[2][3]](https://www.anthropic.com/pricing#:~:text=).
* *Claude Haiku 3.5:* 20 Tokens → ~0,000016$ + 300 Tokens → ~0,0012$ = **~0,00122 $**[[1]](https://www.anthropic.com/pricing#:~:text=%2A%20).
* *Codex CLI (GPT-5):* 20 Tokens → ~0,000025$ (20/1000·$1,25) + 300 Tokens → ~0,0030$ (300/1000·$10) = **~0,00303 $**[[7]](https://openai.com/api/pricing/#:~:text=Input%3A%20%241).
* *Codex CLI (Codex-mini):* 20 Tokens → ~0,00003$ + 300 Tokens → ~0,0018$ = **~0,00183 $**[[6]](https://help.openai.com/en/articles/11381614-codex-cli-and-sign-in-with-chatgpt#:~:text=,6%20%2F%20M%20output%20tokens).
* **Flutter-UI-Komponente generieren:** Beispiel-Prompt mit 30 Tokens, Ausgabe ~500 Tokens. <br>
* *Claude Sonnet 4:* 30 Tokens → ~0,00009$ + 500 Tokens → ~0,0075$ = **~0,00759 $**.
* *Claude Haiku 3.5:* 30 Tokens → ~0,000024$ + 500 Tokens → ~0,0020$ = **~0,00202 $**.
* *Codex CLI (GPT-5):* 30 Tokens → ~0,0000375$ + 500 Tokens → ~0,0050$ = **~0,00504 $**.
* *Codex CLI (Codex-mini):* 30 Tokens → ~0,000045$ + 500 Tokens → ~0,0030$ = **~0,003045 $**.

*(Die obigen Werte basieren auf den in der Tabelle zitierten Preisen; gerundet auf 4 Stellen.)*

## Bewertung und Empfehlung

**Claude Code (Anthropic)** punktet mit starken Agentenfunktionen (automatisches Refactoring, mehrstufige Queries, Code-Execution-Tool) und kann – insbesondere mit Sonnet 4 – riesige Kontexte (bis zu 1 M Token) verarbeiten[[14]](https://every.to/vibe-check/vibe-check-claude-sonnet-4-now-has-a-1-million-token-context-window#:~:text=Today%2C%20Anthropic%20is%20releasing%20a,Potter%20books%20in%20each%20prompt). Für sehr umfangreiche Code-Basen oder komplexe mehrstufige Workflows ist dies einzigartig. Allerdings sind die aktuell besten Claude-Modelle teuer (Opus 4.1 kostet 5× mehr als GPT-5)[[4]](https://www.anthropic.com/pricing#:~:text=Most%20intelligent%20model%20for%20complex,tasks)[[7]](https://openai.com/api/pricing/#:~:text=Input%3A%20%241). Die Qualität der Code-Generierung ist hoch, aber in internen OpenAI-Tests wurde gezeigt, dass GPT-4.1/5 noch überlegen sind (z.B. bei Frontend-UI-Code schlägt GPT-4.1 GPT-4o mit großem Vorsprung)[[20]](https://openai.com/index/gpt-4-1/#:~:text=GPT%E2%80%914,of%20the%20time). **Codex CLI (OpenAI)** liefert schnellen Zugriff auf die neuesten GPT-Modelle. GPT-5 ist laut OpenAI „das beste Modell für Codierung und agentische Aufgaben“[[19]](https://openai.com/gpt-5/#:~:text=GPT%E2%80%915%20produces%20high,%E2%80%98verbosity%E2%80%99%20parameter%20in%20the%20API), bietet 400k Kontext bei moderaten Kosten[[15]](https://openai.com/gpt-5/#:~:text=GPT,opens%20in%20a%20new%20window) und glänzt in API-Tests mit besonders gutem UI- und Mehrsprachencode. Der Agent-CLI-Workflow ist schlank und flexibel, benötigt aber OpenAI-Zugang (ChatGPT-Login).

**Empfehlung:** Für reine Python- und Flutter-Entwicklung bei guter Kosten-/Leistungs-Anforderung ist **Codex CLI (mit GPT-5)** in der Regel die bessere Wahl. Es erzeugt sehr hochwertigen Code (inklusive UI-Generierung) und ist deutlich günstiger pro Token als Claude Opus[[7]](https://openai.com/api/pricing/#:~:text=Input%3A%20%241)[[20]](https://openai.com/index/gpt-4-1/#:~:text=GPT%E2%80%914,of%20the%20time). Bei Budget-Restriktionen kann man auf die billigeren Varianten (GPT-5 mini, Claude Haiku) ausweichen. Nur wenn außergewöhnlich lange Kontexte oder Anthropic-spezifische Workflows (etwa mehrere Claude-Agenten via Claude-Flow[[17]](https://github.com/ruvnet/claude-flow#:~:text=Overview)) nötig sind, lohnt sich **Claude Code**. Insgesamt liegt für typische Aufgaben – unter Berücksichtigung von Preis, Leistung und Support für Python/Flutter – **Codex CLI (OpenAI)** vorn.

**Quellen:** Offizielle Preislisten und Dokumentationen von Anthropic und OpenAI[[4]](https://www.anthropic.com/pricing#:~:text=Most%20intelligent%20model%20for%20complex,tasks)[[7]](https://openai.com/api/pricing/#:~:text=Input%3A%20%241)[[15]](https://openai.com/gpt-5/#:~:text=GPT,opens%20in%20a%20new%20window)[[20]](https://openai.com/index/gpt-4-1/#:~:text=GPT%E2%80%914,of%20the%20time)[[6]](https://help.openai.com/en/articles/11381614-codex-cli-and-sign-in-with-chatgpt#:~:text=,6%20%2F%20M%20output%20tokens).

[[1]](https://www.anthropic.com/pricing" \l ":~:text=%2A%20) [[2]](https://www.anthropic.com/pricing#:~:text=) [[3]](https://www.anthropic.com/pricing#:~:text=) [[4]](https://www.anthropic.com/pricing#:~:text=Most%20intelligent%20model%20for%20complex,tasks) [[5]](https://www.anthropic.com/pricing#:~:text=Write%20%2418) [[21]](https://www.anthropic.com/pricing#:~:text=%2A%20) Pricing \ Anthropic

<https://www.anthropic.com/pricing>

[[6]](https://help.openai.com/en/articles/11381614-codex-cli-and-sign-in-with-chatgpt#:~:text=,6%20%2F%20M%20output%20tokens) Codex CLI and Sign in with ChatGPT | OpenAI Help Center

<https://help.openai.com/en/articles/11381614-codex-cli-and-sign-in-with-chatgpt>

[[7]](https://openai.com/api/pricing/#:~:text=Input%3A%20%241) [[8]](https://openai.com/api/pricing/#:~:text=Input%3A%20%240) Pricing | OpenAI

<https://openai.com/api/pricing/>

[[9]](https://openai.com/index/gpt-4-1/#:~:text=rewriting%20an%20entire%20file,latency%20of%20full%20file%20rewrites) [[16]](https://openai.com/index/gpt-4-1/#:~:text=prompting%20guide%20%E2%81%A0%20,latency%20of%20full%20file%20rewrites) [[20]](https://openai.com/index/gpt-4-1/#:~:text=GPT%E2%80%914,of%20the%20time) Introducing GPT-4.1 in the API | OpenAI

<https://openai.com/index/gpt-4-1/>

[[10]](https://support.anthropic.com/en/articles/11940350-claude-code-model-configuration#:~:text=%2A%20Opus%204.1%2C%20%60claude) Claude Code Model Configuration | Anthropic Help Center

<https://support.anthropic.com/en/articles/11940350-claude-code-model-configuration>

[[11]](https://help.openai.com/en/articles/11096431-openai-codex-cli-getting-started#:~:text=By%20default%20Codex%20targets%20GPT,thought%20summaries%20in%20the%20API) OpenAI Codex CLI – Getting Started | OpenAI Help Center

<https://help.openai.com/en/articles/11096431-openai-codex-cli-getting-started>

[[12]](https://openai.com/index/introducing-o3-and-o4-mini/#:~:text=We%E2%80%99re%20also%20sharing%20a%20new,1%20%E2%81%A0) Introducing OpenAI o3 and o4-mini | OpenAI

<https://openai.com/index/introducing-o3-and-o4-mini/>

[[13]](https://docs.anthropic.com/en/docs/build-with-claude/context-windows#:~:text=,generating%20new%20output%20from%20Claude) Context windows - Anthropic

<https://docs.anthropic.com/en/docs/build-with-claude/context-windows>

[[14]](https://every.to/vibe-check/vibe-check-claude-sonnet-4-now-has-a-1-million-token-context-window#:~:text=Today%2C%20Anthropic%20is%20releasing%20a,Potter%20books%20in%20each%20prompt) Vibe Check: Claude Sonnet 4 Now Has a 1-million Token Context Window

<https://every.to/vibe-check/vibe-check-claude-sonnet-4-now-has-a-1-million-token-context-window>

[[15]](https://openai.com/gpt-5/#:~:text=GPT,opens%20in%20a%20new%20window) [[19]](https://openai.com/gpt-5/#:~:text=GPT%E2%80%915%20produces%20high,%E2%80%98verbosity%E2%80%99%20parameter%20in%20the%20API) GPT-5 is here | OpenAI

<https://openai.com/gpt-5/>

[[17]](https://github.com/ruvnet/claude-flow#:~:text=Overview) GitHub - ruvnet/claude-flow: Claude-Flow v2.0.0 Alpha represents a leap in AI-powered development orchestration. Built from the ground up with enterprise-grade architecture, advanced swarm intelligence, and seamless Claude Code integration.

<https://github.com/ruvnet/claude-flow>

[[18]](https://medium.com/@averageguymedianow/claude-code-in-your-workflow-the-good-the-bad-and-the-ugly-304e4f5a9dc9#:~:text=Claude%20is%20an%20AI%20assistant,simple%20completion%20tools%2C%20Claude%20can) Claude Code in Your Workflow: The Good, The Bad, and The Ugly | by Averageguymedianow | Jul, 2025 | Medium

<https://medium.com/@averageguymedianow/claude-code-in-your-workflow-the-good-the-bad-and-the-ugly-304e4f5a9dc9>