REPO

GRATIS-GUIDE



Welches Repo läuft in *welcher KI?*

Dein Check zum Video: was in den „10 Repos“-Listen wirklich steckt, wie du die drei Arten in zwei Minuten auseinanderhältst, welche Art in Claude, Codex, ChatGPT, GitHub Copilot und Microsoft 365 Copilot läuft, wie du sie dort einrichtest, worauf du vor dem Installieren schaust und was die Google-Suchdaten über den Hype verraten.

Kevin Coo

cooexistenz.de

Danke fürs Folgen ✓ · Du hast kommentiert: REPO ✓

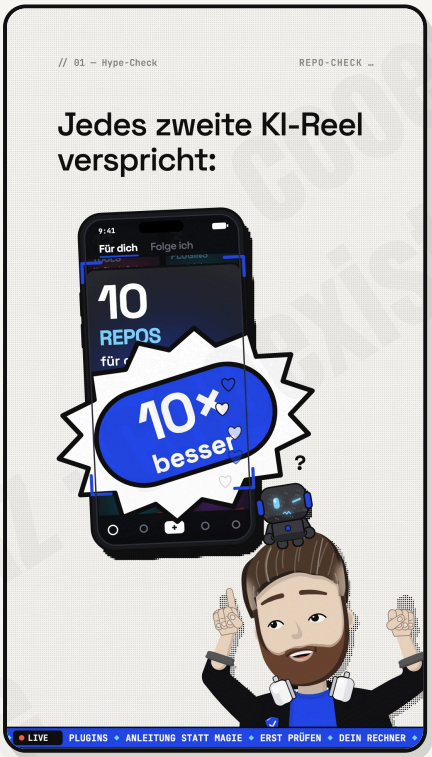
© 2026 cooexistenz · Kevin Coo · cooexistenz.de · Persönliches Exemplar, Weitergabe und Verkauf nicht gestattet · ID CX-20261102-C22A8F58

// 01

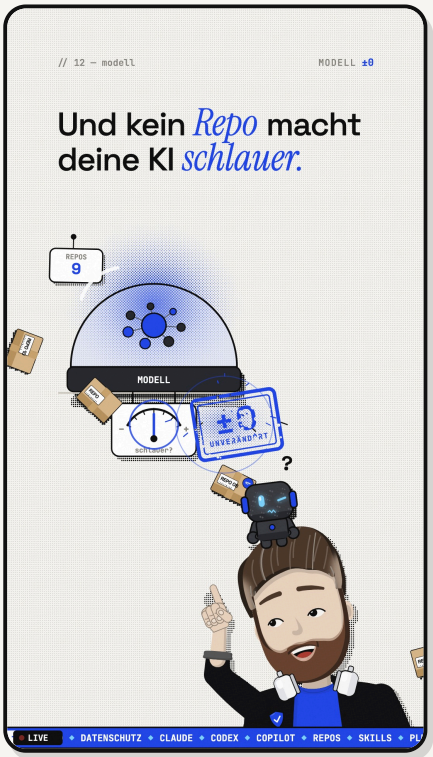
Auf einen Blick

FRAGE	KURZ GESAGT
Was ist ein Repo?	Ein Projektordner auf GitHub. Darin kann Code liegen, eine Anleitung für deine KI oder beides
Was steckt in den Listen?	Drei Arten: Skills (Anleitungen), Plugins (Werkzeuge mit Befehlen) und eigene Programme
Skills	Laufen fast überall: Claude, Codex, ChatGPT, GitHub Copilot und in Agents von Microsoft 365 Copilot
Plugins	Ihr volles Können nur in Werkzeugen, die auf deinem Rechner arbeiten dürfen, vor allem Claude Code und Codex
Programme	Laufen für sich. Nützlich, aber sie machen deine KI nicht besser
Wird meine KI schlauer?	Nein. Das Modell bleibt gleich. Es bekommt nur eine Anleitung oder ein Werkzeug dazu
Die zwei Fragen	Läuft das bei mir? Löst es mein Problem?

// AUS DEM VIDEO







© 2026 cooexistenz · Kevin Coo · cooexistenz.de · Persönliches Exemplar, Weitergabe und Verkauf nicht gestattet · ID CX-20261102-C22A8F58

// 02

Was ein Repo wirklich ist

Ein **Repository**, kurz Repo, ist ein Projektordner auf GitHub. GitHub ist eine Plattform, auf der Menschen Code und Dateien ablegen, teilen und gemeinsam weiterentwickeln. Ein Repo kann ein riesiges Programm enthalten, eine einzige Textdatei oder eine Sammlung von Anleitungen. Von außen sehen alle gleich aus: ein Name, eine Beschreibung, ein paar Sterne.

Genau deshalb sind die „10 Repos für dein Claude“-Listen so verwirrend. Sie stellen Dinge nebeneinander, die technisch nichts miteinander zu tun haben. Ein Repo voller Anleitungen und ein Repo mit einem Stimmen-Klon landen auf derselben Liste, als wären sie dasselbe.

Was die Listen fast nie sagen:

- **In welcher KI es läuft.** Ein Repo, das in Claude Code glänzt, macht im normalen Chat-Fenster oft nur einen Bruchteil.
- **Was es auf deinem Rechner tut.** Manche Repos sind reiner Text. Andere führen Befehle mit deinen Rechten aus.
- **Dass die KI nicht schlauer wird.** Kein Repo verändert das trainierte Modell. Es kommt nur etwas dazu: eine Anleitung, die Claude liest, wenn sie passt, oder ein Werkzeug, das es benutzen darf.



DER MENTOR SAGT

Die Listen sind nicht falsch. Ihnen fehlen nur zwei Fragen: Läuft das bei dir? Und löst es dein Problem?

// 03

Die drei Arten in zwei Minuten erkennen

Öffne das Repo auf GitHub und schau auf die Dateiliste. Du musst keinen Code lesen können, die Dateinamen verraten fast alles.

ART	DU ERKENNST SIE AN	WAS SIE TUT
Skill	einem Ordner mit einer Datei `SKILL.md`, oft nur Markdown-Dateien	Eine Anleitung: wie die KI eine Aufgabe erledigen soll. Name und Beschreibung oben, Schritte darunter
Plugin	Ordnern wie `skills/`, `hooks/`, `bin/`, einer Datei `.mcp.json` oder einem Plugin-Manifest	Ein Paket aus Skills plus Werkzeugen: Befehle, Hooks, Verbindungen zu anderen Diensten
Programm	`package.json`, `requirements.txt`, einer Installationsanleitung, einer eigenen Oberfläche	Eine eigenständige App, die du selbst startest. Sie braucht keine KI-Anbindung

© 2026 cooexistenz · Kevin Coo · cooexistenz.de · Persönliches Exemplar, Weitergabe und Verkauf nicht gestattet · ID CX-20261102-C22A8F58

Ein Skill ist der offene Standard „Agent Skills“. Anthropic hat ihn entwickelt und als offenen Standard veröffentlicht. Pflicht sind nur zwei Felder: `name` (klein geschrieben, mit Bindestrichen, gleich wie der Ordnername) und `description` (was der Skill tut und wann er gebraucht wird). Optional kommen Ordner wie `scripts/`, `references/` oder `assets/` dazu. Sobald ein Skill einen `scripts/`-Ordner hat, enthält er ausführbaren Code. Dann gilt er nicht mehr als „nur Text“.

Merksatz: Text liest die KI. Code führt sie aus. Schau nach, was davon im Repo liegt.

// 04

Welches Repo läuft in welcher KI?

Das ist der Kern deines Checks. Stand 1. November 2026, abgeglichen mit den offiziellen Seiten der Anbieter.

KI	SKILL (ANLEITUNG)	PLUGIN (WERKZEUG)	PROGRAMM
Claude Code	ja, voll	ja, voll: Skills, Befehle, Hooks, Agents, MCP	läuft für sich
claude.ai Chat	ja, als ZIP-Upload, Code-Ausführung muss an sein	ja, aber im Chat laden nur die Skills; Agents, Hooks und lokale Server werden ignoriert	läuft für sich
Codex (CLI, IDE)	ja, Ordner `.agents/skills`	ja, in der Codex CLI über `/plugins`; die IDE-Erweiterung kann laut Doku keine Plugins	läuft für sich
ChatGPT	ja, eigenständig in der Desktop-App; Skills in Plugins auch im Web und mobil	ja, Plugins mit Skills und MCP-Servern	läuft für sich
GitHub Copilot	ja: Cloud-Agent, Code Review, CLI, Copilot-App, Agent-Modus in VS Code, Visual Studio, JetBrains	über Skills und MCP-Server	läuft für sich
Microsoft 365 Copilot	ja, in eigenen Agents im Agent Builder, mit Copilot-Lizenz	nein, als Plugin nicht; Skript-Dateien laufen dort in einer abgeschotteten Umgebung	läuft für sich

© 2026 cooexistenz · Kevin Coo · cooexistenz.de · Persönliches Exemplar, Weitergabe und Verkauf nicht gestattet · ID CX-20261102-C22A8F58

Drei Dinge fallen auf:

1. **Skills sind die tragbarste Art.** Derselbe Ordner mit `SKILL.md` funktioniert in fast allen großen KI-Werkzeugen. Wenn du dir etwas Eigenes baust, dann einen Skill.
2. **Plugins sind an ihr Werkzeug gebunden.** Ein Claude-Code-Plugin, das Befehle auf deinem Rechner ausführt, kann im Browser-Chat genau das nicht. Dort bleibt meist nur die Anleitung übrig.
3. **Programme gehören nicht in diese Listen.** Ein Stimmen-Klon oder eine Hintergrund-Bibliothek kann großartig sein. Mit deiner KI hat sie nichts zu tun.



DER MENTOR SAGT

Ein Blatt Papier passt durch jeden Briefschlitz. Ein Gerät mit Stecker braucht eine Steckdose. So einfach ist der Unterschied.

// 05

So richtest du einen Skill ein

Für alle Wege gilt: Der Skill ist ein Ordner, im Ordner liegt `SKILL.md`, und der Ordnername entspricht dem `name`-Feld. Lade nur Skills, die du vorher angesehen hast (siehe „Der 5-Minuten-Check“).

Claude Code

Lege den Ordner an einer dieser Stellen ab:

```
~/.claude/skills/<skill-name>/SKILL.md
.claude/skills/<skill-name>/SKILL.md
```

Die erste Stelle gilt für alle deine Projekte auf diesem Rechner, die zweite nur für das aktuelle Projekt. Claude nutzt den Skill automatisch, wenn er passt. Direkt aufrufen kannst du ihn mit `/skill-name`, eine Übersicht zeigt `/skills`. Neu: Skills, die du in claude.ai hochlädst, synchronisiert Claude Code in den Ordner `~/.claude/skills/synced/`, wenn du mit deinem Konto angemeldet bist.

claude.ai (Web und Desktop-App)

1. Code-Ausführung in den Einstellungen einschalten (ohne sie laufen keine Skills).
2. **Customize › Skills** öffnen, auf „+“ klicken, dann „Create skill“ und „Upload a skill“.
3. Eine ZIP-Datei hochladen, in der der Skill-Ordner liegt.
4. Im Chat `/` tippen: Das Menü zeigt deine Skills.

© 2026 cooexistenz · Kevin Coo · cooexistenz.de · Persönliches Exemplar, Weitergabe und Verkauf nicht gestattet · ID CX-20261102-C22A8F58

Codex und ChatGPT

Codex sucht Skills im Ordner `.agents/skills` (im Projekt, vom aktuellen Ordner bis zur Wurzel des Repos) und in `$HOME/.agents/skills` (für alle Projekte). In der Codex CLI und der IDE-Erweiterung rufst du einen Skill mit `/skills` auf oder erwähnst ihn mit `$`. In ChatGPT wählst du einen Skill mit `@`. Eigenständige Skills laufen laut OpenAI in der ChatGPT-Desktop-App, der Codex CLI und der IDE-Erweiterung. Skills, die in einem Plugin stecken, gibt es auch in ChatGPT im Web und mobil.

GitHub Copilot

Für ein Projekt legst du Skills in einen dieser Ordner im Repo: `.github/skills`, `.claude/skills` oder `.agents/skills`. Für dich persönlich in `~/.copilot/skills` oder `~/.agents/skills`. In der Copilot CLI zeigt `/skills list` alle Skills. Neu ist der Befehl `gh skill` (öffentliche Vorschau, GitHub CLI 2.90.0 oder neuer):

```
gh skill search TOPIC
gh skill preview OWNER/REPOSITORY SKILL
gh skill install OWNER/REPOSITORY SKILL
```

GitHub weist selbst darauf hin, dass diese Skills nicht von GitHub geprüft sind.

Microsoft 365 Copilot

Hier gehört ein Skill zu einem eigenen Agent. Du brauchst eine Microsoft-365-Copilot-Lizenz oder Zugang über nutzungsabhängige Abrechnung.

1. Im Agent Builder einen Agent öffnen oder neu anlegen.
2. Den Tab **Configure** öffnen und **Skills** aufklappen.
3. **Add** wählen und das vollständige ZIP hochladen (nicht nur die `SKILL.md`).
4. Den Inhalt prüfen und im Tab **Try it** testen.

Grenzen laut Microsoft: höchstens 8 Skills pro Agent, ZIP bis 50 MB. Skripte laufen dort in einer abgeschotteten Umgebung ohne Internet. Ob die Funktion schon allgemein verfügbar ist oder noch als Vorschau gilt, sagen die Seiten nicht ausdrücklich. [prüfen] in deinem Mandanten, ob sie freigeschaltet ist.

© 2026 cooexistenz · Kevin Coo · cooexistenz.de · Persönliches Exemplar, Weitergabe und Verkauf nicht gestattet · ID CX-20261102-C22A8F58

// 06

Plugins: wo sie ihr volles Können zeigen

Ein Plugin ist ein Paket. Es kann Skills enthalten, dazu Befehle, Hooks (Aktionen, die bei bestimmten Ereignissen automatisch laufen), eigene Agents und Verbindungen über MCP. MCP ist ein offener Standard, mit dem eine KI an andere Dienste andockt, ein bisschen wie ein USB-C-Anschluss für KI-Apps.

Claude Code kann alles davon. Der offizielle Marktplatz von Anthropic ist schon eingerichtet. So installierst du ein Plugin in einer Sitzung:

```
/plugin
/plugin install <plugin-name>@claude-plugins-official
/plugin marketplace add <owner/repo>
```

`/plugin` ohne Namen öffnet die Übersicht zum Stöbern. Mit `/plugin marketplace add` fügst du einen weiteren Marktplatz aus einem GitHub-Repo hinzu. Ausschalten und entfernen geht über `/plugin disable` und `/plugin uninstall`.

claude.ai kann Plugins inzwischen auch hinzufügen, unter **Customize › Plugins**. Im Chat laden davon aber nur die Skills. Agents, Hooks, lokale MCP-Server und Einstellungen werden ignoriert, und ein Plugin mit `bin/`-Ordner lässt sich gar nicht installieren. Genau das meint der Satz im Video: Im normalen Chat läuft meist nur die Anleitung.

Codex CLI: `/plugins` öffnet den Plugin-Browser, danach eine neue Sitzung starten. **ChatGPT** zeigt Plugins im Web und am Desktop unter „Plugins“, mobil nutzt du die, die schon in deinem Konto sind.



DER MENTOR SAGT

Wenn ein Plugin Befehle auf deinem Rechner ausführt, ist es Software. Behandle es auch so.

© 2026 cooexistenz · Kevin Coo · cooexistenz.de · Persönliches Exemplar, Weitergabe und Verkauf nicht gestattet · ID CX-20261102-C22A8F58

// 07

Der 5-Minuten-Check vor jeder Installation

Anthropic sagt es deutlich: Nutze Skills nur aus vertrauenswürdigen Quellen, also selbst gebaute oder von Anthropic. Prüfe alle Dateien, die dabei sind, und behandle die Installation wie die Installation von Software. Für Plugins in Claude Code steht in der Doku sogar, dass sie beliebigen Code mit deinen Benutzerrechten ausführen können.

So prüfst du ein Repo, bevor es auf deinem Rechner landet:

1. **Dateiliste ansehen.** Auf github.com im Repo durch die Ordner klicken. Nur `SKILL.md` und weitere `.md`-Dateien? Dann ist es Text. `scripts/`, `bin/`, `hooks/hooks.json` oder `.mcp.json`? Dann läuft Code.
2. **SKILL.md lesen.** Passt die Beschreibung zu dem, was du willst? Steht irgendwo, dass Daten an fremde Adressen gehen?
3. **Ohne Installation ansehen.** Wenn du die Kommandozeile nutzt, geht das auch ohne Browser:

```
gh api "repos/<owner>/<repo>/git/trees/HEAD?recursive=1" --jq '.tree[].path'
npx skills add <owner/repo> --list
gh skill preview OWNER/REPOSITORY SKILL
```

Der erste Befehl listet alle Dateien des Repos, der zweite zeigt die enthaltenen Skills, ohne sie zu installieren (braucht Node.js 22.20 oder neuer), der dritte zeigt einen Skill in der Vorschau.

4. **Sicherheits-Scans als Hinweis nutzen.** Auf skills.sh/audits zeigt Vercel die Ergebnisse von drei Scannern (Gen Agent Trust Hub, Socket, Snyk). Was genau geprüft wird, erklärt die Seite nicht. Die Scans ersetzen deinen Blick also nicht.
5. **Erst klein testen.** In einem Übungsprojekt mit erfundenen Daten, nie zuerst mit Kundendaten.

Agent Skills · offizielle Doku von Anthropic

was Skills sind und die Sicherheitshinweise im Original

platform.claude.com ↗

agentskills.io

der offene Standard: Aufbau eines Skills und Pflichtfelder

agentskills.io ↗

Plugin-Sicherheit in Claude Code

welche Dateien du vor einem Plugin lesen solltest

code.claude.com ↗

© 2026 cooexistenz · Kevin Coo · cooexistenz.de · Persönliches Exemplar, Weitergabe und Verkauf nicht gestattet · ID CX-20261102-C22A8F58

// 08

Was die Suchdaten über den Hype verraten

Ich habe mit DataForSEO nachgesehen, wie oft in Deutschland bei Google nach diesen Begriffen gesucht wird: Monatsdurchschnitt aus Google-Ads-Daten, Oktober 2025 bis September 2026, abgerufen am 1. November 2026. Google fasst ähnliche Begriffe zusammen, „github repo“ und „github repository“ haben deshalb dieselbe Zahl.

SUCHBEGRIFF	Ø PRO MONAT	OKT 2025 → SEP 2026
claude code	135.000	40.500 → 110.000
codex	60.500	22.200 → 90.500
github copilot	33.100	22.200 → 22.200
claude skills	8.100	4.400 → 6.600
agent skills	1.600	70 → 1.900
claude plugins	1.600	320 → 1.900
claude code installieren	1.000	140 → 1.000
codex skills	1.000	10 → 720
chatgpt skills	480	30 → 880
was ist ein repository	480	320 → 590
skill.md	260	10 → 260

Was ich daraus lese:

- **Skills sind in einem Jahr vom Nischenbegriff zum Alltagsthema geworden.** „agent skills“ stieg von 70 auf 1.900 Suchen im Monat, „codex skills“ von 10 auf 720, „chatgpt skills“ von 30 auf 880. Das passt dazu, dass inzwischen fast jeder Anbieter denselben Standard unterstützt.
- **Viele stehen noch ganz am Anfang.** „claude code installieren“ hat sich in einem Jahr versiebenfacht, „was ist ein repository“ wird jeden Monat Hunderte Male gesucht. Die Listen erreichen also viele, die noch gar nicht wissen, was ein Repo ist.
- **Es fehlt verständliches Wissen auf Deutsch.** Bei „claude skills“ stehen auf Google.de oben vor allem englische Anleitungen und Code-Schnipsel. Unter „github repos ki“ fragen Leute: „Which AI works with GitHub?“. Genau diese Frage beantwortet die Tabelle oben.

© 2026 cooexistenz · Kevin Coo · cooexistenz.de · Persönliches Exemplar, Weitergabe und Verkauf nicht gestattet · ID CX-20261102-C22A8F58

// 09

Prompts zum Kopieren

Diese Prompts funktionieren in Claude, ChatGPT oder Copilot. Ersetze die Teile in eckigen Klammern.

1 · Ein Repo einordnen lassen

```
Hier ist der Link zu einem GitHub-Repo: [Link]. Sieh dir die Dateiliste und die README an und sag mir in einfachen Worten: Ist das ein Skill, ein Plugin oder ein eigenständiges Programm? Welche Dateien führen Code aus? In welchen dieser Werkzeuge läuft es: Claude Code, claude.ai, Codex, ChatGPT, GitHub Copilot, Microsoft 365 Copilot? Begründe jede Einschätzung mit dem Dateinamen, auf den du dich stützt.
```

2 · Sicherheits-Check vor der Installation

```
Prüfe diese SKILL.md und alle Dateien, die ich dir unten einfüge, wie ein vorsichtiger IT-Sicherheitsberater. Liste auf: welche Befehle ausgeführt werden, ob Daten an externe Adressen gehen, ob Zugangsdaten oder Dateien außerhalb des Projekts gelesen werden. Gib am Ende eine Ampel (grün, gelb, rot) und sag mir, was ich vor einer Installation klären sollte. [Dateien einfügen]
```

3 · Passt das zu meinem Problem?

```
Mein Problem im Alltag ist: [beschreiben]. Ich nutze [Werkzeug, z. B. claude.ai im Browser]. Hier ist eine Liste von Repos aus einem Video: [Liste]. Sortiere sie in drei Gruppen: passt zu meinem Problem und läuft in meinem Werkzeug / passt, läuft aber nicht bei mir / passt nicht. Erkläre jede Einordnung in einem Satz.
```

4 · Einen eigenen Skill bauen statt zu suchen

```
Hilf mir, einen eigenen Skill nach dem offenen Standard Agent Skills zu bauen. Die Aufgabe: [Aufgabe, z. B. Angebote nach unserer Vorlage schreiben]. Stell mir zuerst bis zu acht Fragen zu Ablauf, Tonfall und Ausnahmen. Schreib danach eine SKILL.md mit den Feldern name und description und einer nummerierten Anleitung. Kein Code, nur Text.
```

5 · Einen Skill für ein anderes Werkzeug umziehen

```
Hier ist ein Skill-Ordner, den ich in Claude Code nutze: [Inhalt der SKILL.md]. Ich möchte ihn auch in [GitHub Copilot / Codex / Microsoft 365 Copilot] nutzen. Sag mir, in welchen Ordner er dort gehört oder wie ich ihn hochlade, welche Felder ich anpassen muss und was dort nicht funktioniert. Nenne nur Dinge, die du sicher weißt, und markiere Unsicheres.
```

© 2026 cooexistenz · Kevin Coo · cooexistenz.de · Persönliches Exemplar, Weitergabe und Verkauf nicht gestattet · ID CX-20261102-C22A8F58

6 · Die Liste aus dem Video entzaubern

```
Ein Video verspricht: [Titel, z. B. „10 Repos, die dein Claude zehnmal besser machen“]. Hier sind die Repos: [Liste]. Prüfe für jedes: Was ist es wirklich? Ändert es das Modell oder gibt es nur eine Anleitung oder ein Werkzeug dazu? Welche Behauptung im Titel ist übertrieben? Antworte als Tabelle mit den Spalten Repo, Art, Was es wirklich tut, Übertreibung.
```

// 10

Wer was davon hat

WER	TYPISCHES PROBLEM	WAS DU KONKRET MACHST
Büro und Verwaltung	Immer gleiche Texte und Abläufe	Einen eigenen Skill für Vorlagen, Protokolle oder Antworten bauen und in Claude oder Microsoft 365 Copilot nutzen
Teams mit Copilot	Jeder promptet anders	Einen Agent im Agent Builder mit zwei, drei Team-Skills ausstatten
Selbstständige	Keine Zeit, zehn Repos zu testen	Mit Prompt 3 die Liste auf ein Repo für das echte Problem eindampfen
Entwicklerinnen und Entwickler	Die KI legt zu früh los oder baut zu viel um	Gezielt ein Plugin in Claude Code oder Codex testen, im Übungsprojekt
Content-Creator	Hype-Listen verbreiten, ohne sie zu prüfen	Mit Prompt 6 die Liste prüfen und ehrlich einordnen
IT und Datenschutz	Mitarbeitende installieren ungeprüfte Erweiterungen	Den 5-Minuten-Check als feste Regel einführen

© 2026 cooexistenz · Kevin Coo · cooexistenz.de · Persönliches Exemplar, Weitergabe und Verkauf nicht gestattet · ID CX-20261102-C22A8F58

// 11

Grenzen und Risiken

- **Code mit deinen Rechten.** Plugins und Skills mit Skripten können auf deinem Rechner Dateien lesen und Befehle ausführen. Die Claude-Code-Doku sagt das ausdrücklich.
- **Anleitungen können manipuliert sein.** Auch reiner Text kann die KI zu Dingen bringen, die du nicht willst, zum Beispiel Daten an eine fremde Adresse zu schicken. Darum gilt: lesen vor laden.
- **Repos ändern sich.** Was heute harmlos ist, kann nach einem Update anders aussehen. Ein Skill, der Inhalte von fremden Adressen nachlädt, ist besonders heikel.
- **Datenschutz.** Teste nie mit echten Kunden- oder Personaldaten. Im Büro gilt zusätzlich, was deine IT, dein Datenschutzbeauftragter oder der Betriebsrat freigegeben haben.
- **Lizenzen.** Nicht jedes Repo darf beliebig weiterverwendet oder verkauft werden. Steht keine Lizenz dabei, frag den Urheber.
- **Zu viel auf einmal.** Viele Skills mit ähnlichen Beschreibungen machen es der KI schwer, den richtigen zu wählen. Lieber wenige, die einen klaren Job haben.
- **Stand der Dinge.** Menüs, Befehle und Pläne ändern sich schnell. Die Angaben hier haben ein Datum. Prüf die verlinkten Seiten, bevor du etwas im Team ausrollst.

Skills in Claude nutzen · Hilfe-Center

Upload, Pläne, Code-Ausführung

support.claude.com ↗

Skills bauen · OpenAI

Ordner und Aufruf in Codex und ChatGPT

learn.chatgpt.com ↗

Agent Skills · GitHub Docs

Ordner und Oberflächen in GitHub Copilot

docs.github.com ↗

Skills im Agent Builder · Microsoft Learn

Hochladen, Lizenz, Grenzen in Microsoft 365 Copilot

learn.microsoft.com ↗

© 2026 cooexistenz · Kevin Coo · cooexistenz.de · Persönliches Exemplar, Weitergabe und Verkauf nicht gestattet · ID CX-20261102-C22A8F58

// 12

Deine erste Stunde

- [ ] Schreib dein Problem in einem Satz auf, bevor du irgendeine Liste öffnest.
- [ ] Notier, welches Werkzeug du wirklich nutzt: Claude im Browser, Claude Code, Codex, ChatGPT, Copilot oder Microsoft 365 Copilot.
- [ ] Nimm aus der Liste höchstens ein Repo, das zu diesem Problem passt.
- [ ] Ordne es ein: Skill, Plugin oder Programm (Tabelle „Die drei Arten“).
- [ ] Schau in der großen Tabelle nach, ob diese Art in deinem Werkzeug läuft.
- [ ] Mach den 5-Minuten-Check: Dateiliste, SKILL.md, Code-Dateien.
- [ ] Lass Prompt 2 die Dateien prüfen, wenn du unsicher bist.
- [ ] Teste es in einem Übungsprojekt mit erfundenen Daten.
- [ ] Entscheide nach einer Woche: behalten, anpassen oder wieder entfernen.

// 13

Mein Tipp

Bevor du das nächste „10 Repos“-Video speicherst, stell dir die zwei Fragen aus dem Video: Läuft das bei mir? Löst es mein Problem? In den meisten Fällen bleibt dann ein einziges Repo übrig, oft ein Skill. Und sehr oft ist der beste Skill der, den du dir in einer halben Stunde selbst schreibst: mit deiner Vorlage, deinem Tonfall und deinen Regeln. Den kannst du dann in fast jede KI mitnehmen.

Im nächsten Reel zeige ich dir, wie ich für jedes Projekt die passenden Repos und Skills finde. Folg mir, damit du Teil 2 nicht verpasst.

Alle Links, Befehle und Angaben habe ich am 1. November 2026 auf den offiziellen Seiten der Anbieter geprüft. Suchdaten: DataForSEO (Google Ads, Deutschland), abgerufen am 1. November 2026.

© 2026 cooexistenz · Kevin Coo · cooexistenz.de · Persönliches Exemplar, Weitergabe und Verkauf nicht gestattet · ID CX-20261102-C22A8F58



DER MENTOR SAGT

Weniger Repos, die passen, schlagen zehn, die du nie benutzt.

— Kevin Coo

Mehr Anleitungen mit Stand und Quelle, in einfachen Worten und mit Datenschutz im Blick: auf cooexistenz.de.

cooexistenz.de ↗

Kevin Coo · cooexistenz.de

REPO

© 2026 cooexistenz · Kevin Coo · cooexistenz.de · Persönliches Exemplar, Weitergabe und Verkauf nicht gestattet · ID CX-20261102-C22A8F58