

## TAUCHGANG 01 · EINSTEIGER-VARIANTE

# Das Fundament.

Claude Code, CLAUDE.md und Obsidian. Einmal aufgesetzt — und deine KI kennt dich, deine Kunden, deinen Stil.

Dauer: **ca. 2 Stunden**    Niveau: **Einsteiger**

Voraussetzung: **Mac oder Windows · Claude Pro (20 \$/Monat)**

Einsteiger

**Video:** Der Modul-Trailer zu diesem Tauchgang ist Teil der Online-Lektion. Schau ihn dir an unter [sebastian-sperber.de/tauchgang/lektion-01](https://sebastian-sperber.de/tauchgang/lektion-01).

## Für wen ist dieser Text?

**Für dich, wenn du *noch nie im Terminal* warst und bisher nur mit ChatGPT im Browser gearbeitet hast. Du musst nichts vorher können. Jeder Fachbegriff wird erklärt, sobald er das erste Mal fällt.**

**Ergebnis:** Du hast eine KI, die dich kennt. Sie weiß, wer du bist, was du anbietest, wie du arbeitest. Sie vergisst das nicht zwischen den Sitzungen.

## Worum es in diesem Tauchgang geht

Die meisten, die KI ausprobieren, machen denselben Fehler: Sie öffnen ein Chat-Fenster, tippen 30 Sekunden lang, und bekommen ein mittelmäßiges Ergebnis. Dann sagen sie: „KI ist überschätzt.“

Das Problem ist nicht die KI. Das Problem ist der Kontext. Die KI hat keinen.

In diesem Tauchgang baust du drei Dinge, die zusammenarbeiten:

1. **Claude Code** — ein Programm, das auf deinem Rechner läuft und mit Claude spricht. Anders als der Browser-Chat kann es Dateien auf deinem Computer lesen und schreiben.
2. **CLAUDE.md** — eine Text-Datei mit deinen Regeln. Claude Code liest sie bei jedem Start. Einmal schreiben, immer gültig.

3. **Obsidian** — ein Notiz-Programm, das Notizen als einfache Text-Dateien speichert. Dein Business-Wissen lebt dort. Claude Code kann darauf zugreifen.

Ergebnis: Eine KI, die deinen Stundensatz, deine Kunden und deine Arbeitsweise kennt — nicht weil du es ihr jedes Mal erklärst, sondern weil es in Dateien steht.

---

## Lektion 1.1 — Claude Code installieren

### DIE BEGRIFFE, DIE DU BRAUCHST

- **Terminal** — ein Programm, das auf jedem Mac und Windows-Rechner installiert ist. Du gibst dort Befehle als Text ein, statt sie zu klicken. Entwickler arbeiten damit, weil es schneller ist als jede Bedienoberfläche.
- **Befehl** — eine Zeile Text, die du im Terminal eingibst und mit Enter abschickst. Der Computer führt aus, was du geschrieben hast.
- **Node.js** — eine Software, die im Hintergrund läuft und andere Programme ausführt. Claude Code braucht sie, um überhaupt zu starten.
- **npm** — ein „App-Store für Entwickler-Programme“, der zusammen mit Node.js mitinstalliert wird.
- **Claude Pro / Max / Team** — die bezahlten Abos von claude.ai. Ab **Claude Pro (20 \$/Monat)** kannst du Claude Code nutzen.

### Schritt 1 • Terminal öffnen

**Mac:** `Cmd + Leertaste` drücken → „Terminal“ tippen → Enter.

**Windows:** Windows-Taste drücken → „PowerShell“ tippen → Enter.

Ein schwarzes Fenster geht auf. Das ist dein Terminal. Du siehst eine Zeile mit deinem Benutzernamen und einem Zeichen wie `$` oder `%` — das ist die Stelle, an der du Befehle eingibst.

### Schritt 2 • Prüfen, ob Node.js vorhanden ist

Tippe im Terminal:

```
node -v
```

Enter.

**Mögliche Antworten:**

- Eine Zahl wie `v18.17.0` oder `v20.5.1` → Node.js ist da. Wenn die erste Zahl **18 oder höher** ist, geht's weiter mit Schritt 3.
- `command not found: node` → Node.js fehlt. Gehe auf [nodejs.org](https://nodejs.org), lade die **LTS-Version** herunter und installiere sie.

**Was bedeutet `node -v` ?** „Zeig mir die installierte Version ( `v` = Version) von Node.js.“ Der Computer antwortet mit der Versionsnummer.

## Schritt 3 • Claude-Account einrichten

Browser öffnen, auf [claude.ai](https://claude.ai) gehen, registrieren, **Claude Pro** (20 \$/Monat) auswählen. Das ist nötig, damit Claude Code deinen Account erkennt.

## Schritt 4 • Claude Code installieren

Zurück ins Terminal. Diesen Befehl tippen (oder kopieren und einfügen):

```
npm install -g @anthropic-ai/claude-code
```

### Was bedeutet der Befehl?

- `npm install` — **hol dir ein Programm aus dem Internet und installiere es.**
- `-g` — **global, also für den ganzen Rechner verfügbar.**
- `@anthropic-ai/claude-code` — **der offizielle Name des Programms im npm-Store.**

Die Installation rattert 30 Sekunden bis 2 Minuten. Fertig ist sie, wenn du wieder das blinkende Zeichen am Zeilenanfang siehst.

## Schritt 5 • Einmal einloggen

```
claude
```

Claude Code öffnet den Browser und bittet dich, dich einzuloggen. Danach bist du im Terminal in einer Claude-Sitzung — du siehst oben ein `>` und kannst Nachrichten tippen. Einmal `/exit` eingeben, um zu beenden.

**Wenn `claude` nicht gefunden wird:** Terminal schließen, neu öffnen, nochmal probieren.

## Schritt 6 • Die globale **CLAUDE.md** anlegen

Das ist die wichtigste Datei in diesem Tauchgang. Sie enthält deine persönlichen Regeln. Claude Code liest sie bei jedem Start automatisch, egal in welchem Projekt du gerade arbeitest.

Speicherort: `~/ .claude/CLAUDE.md`

### Was bedeutet ~/.claude/ ?

- **~ ist die Abkürzung für deinen Benutzer-Ordner. Auf dem Mac ist das /Users/deinname/ .**
- **.claude ist ein versteckter Ordner (jeder Ordnername, der mit einem Punkt beginnt, ist auf dem Mac unsichtbar).**
- **CLAUDE.md ist die Datei selbst. Die Endung .md steht für Markdown.**

```
mkdir -p ~/.claude
open -a TextEdit ~/.claude/CLAUDE.md
```

TextEdit öffnet sich mit einer leeren Datei. Folgenden Inhalt einfügen und anpassen:

```
# Globale Regeln

## Wer ich bin
- [Dein Name], [Beruf] aus [Ort]
- Kleinunternehmer nach § 19 UStG (falls zutreffend)
- Stundensätze: [X] € (Neukunden) / [Y] € (Bestandskunden)

## Sprache
- Antworte auf Deutsch.
- Code, Git-Commits und Datei-Namen auf Englisch.

## Verhalten
- Sei direkt und knapp. Keine Floskeln.
- Lies Dateien vollständig, bevor du sie änderst.
- Frag nach, wenn du unsicher bist.

## Was ich nicht will
- Keine Zusammenfassungen am Ende deiner Antworten.
- Keine Emojis, außer ich frage explizit danach.
- Keine erfundenen Funktionen oder Pfade.
```

Cmd + S (Speichern). Datei schließen.

## Schritt 7 • Prüfen, ob Claude deine Regeln liest

```
cd ~
claude
```

**cd ~ heißt „change directory“, also „wechsle in den Benutzer-Ordner“.**

Im Claude-Prompt eingeben:

Lies meine globale CLAUDE.md und fasse in drei Sätzen zusammen, wer ich bin und wie ich arbeite.

Wenn Claude deinen Namen, deinen Stundensatz und deine Regeln korrekt wiedergibt: **Die Basis läuft.**

## ***Lektion 1.2 — Obsidian als Wissensbasis***

### ***Warum Obsidian und nicht Notion?***

Notion, Evernote und ähnliche Programme speichern deine Notizen in ihrer Cloud. Vorteile: komfortabel, überall verfügbar, schnell. Nachteile:

- Die Notizen liegen nicht bei dir, sondern bei einer Firma in den USA.
- Wenn die Firma Preise erhöht oder den Dienst einstellt, bist du abhängig.
- Claude Code kann **nicht** auf Notion-Inhalte zugreifen.

Obsidian macht es anders:

- Notizen sind **ganz normale Text-Dateien** auf deinem Rechner (Dateiendung `.md`).
- **DSGVO-freundlich**, weil nichts automatisch in die Cloud geht.
- **Claude Code kann sie lesen, schreiben, verknüpfen.** Weil es einfach Dateien sind.
- **Kostenlos für den Privatgebrauch.** Für Sync reicht iCloud.

### ***Schritt 1 • Obsidian installieren***

Browser: [obsidian.md](https://obsidian.md) → Download → installieren. (Homebrew: `brew install --cask obsidian`)

### ***Schritt 2 • Einen Ordner als „Vault“ anlegen***

Obsidian speichert seine Notizen in einem Ordner, den es **Vault** nennt (englisch für „Tresor“).

Im Finder: In **Dokumente** gehen → Neuen Ordner anlegen: `MeinBusiness` (ohne Umlaute, ohne Leerzeichen am Anfang).

### ***Schritt 3 • Unterordner nach Schema anlegen***

```
cd ~/Documents/MeinBusiness
mkdir 00-daily 00-inbox raw wiki 01-calls 02-proposals 03-marketing 04-crm 05-content 07-templates
mkdir 04-crm/kunden
```

Die Logik hinter den Namen:

| ORDNER         | INHALT                                                                        |
|----------------|-------------------------------------------------------------------------------|
| 00-daily/      | Eine Datei pro Tag: 2026-04-18.md . Was ist erledigt, was offen, was wichtig. |
| 00-inbox/      | Schneller Zwischenspeicher für halbgare Ideen.                                |
| raw/           | Rohmaterial: Artikel, Screenshots, Transkripte. Ungeordnet.                   |
| wiki/          | Die saubere Wissensbasis. Claude pflegt sie aus raw/ .                        |
| 01-calls/      | Vorbereitung und Notizen zu Kundengesprächen.                                 |
| 02-proposals/  | Angebote, die du rausgeschickt hast.                                          |
| 03-marketing/  | Content-Ideen, Social-Posts-Drafts.                                           |
| 04-crm/kunden/ | Eine Datei pro Kunde. Dein eigenes CRM.                                       |
| 05-content/    | Skripte, Blog-Artikel, Newsletter.                                            |
| 07-templates/  | Wiederverwendbare Vorlagen.                                                   |
| 08-personal/   | Alles, was nicht zum Business gehört.                                         |

**Warum die Ziffern am Anfang?** Dateisysteme sortieren alphabetisch. Mit 00- steht der tägliche Ordner oben, mit 08- steht das Persönliche unten.

## ***Schritt 4 • Obsidian mit dem Ordner verbinden***

Obsidian starten → „Open folder as vault“ → Ordner MeinBusiness auswählen.

## ***Schritt 5 • Business-CLAUDE.md schreiben***

In der globalen CLAUDE.md steht, wer **du** bist. In der Business-CLAUDE.md steht, was **dein Business** ist.

In Obsidian oben links auf **File** → **New note**. Name: CLAUDE.md (exakt so, Großbuchstaben wichtig).  
Inhalt:

```
# Business-Kontext

## Wer ich bin
- [Name], [Berufsbezeichnung] seit [Jahr]

## Hauptziel [Jahr]
- [Ein messbares Ziel.]

## Dienstleistungen & Preise
- [Leistung]: [Preis]
- Stundensatz Neukunden: [X] € · Bestand: [Y] €

## Workflows
- Neue Anfrage → Datei in `04-crm/kunden/<id>-<name>.md` anlegen.
- Kundengespräch → `01-calls/YYYY-MM-DD-<name>.md`.
- Angebot → `02-proposals/YYYY-MM-DD-<kunde>-<projekt>.md`.

## Was ich nicht mache
- [Grenze, z. B.: keine Projekte unter 200 €]
```

Speichern. **Was du gerade gebaut hast:** Eine Mini-Wissensbasis über dein eigenes Geschäftsmodell. Jede Claude-Sitzung, die in diesem Vault läuft, liest das automatisch mit.

---

## Lektion 1.3 — Claude und Obsidian zusammenführen

### Schritt 1 • Claude im Vault starten

```
cd ~/Documents/MeinBusiness
claude
```

Claude Code liest jetzt **zwei CLAUDE.md-Dateien automatisch**:

1. Die globale aus `~/Claude/CLAUDE.md` (dein Profil).
2. Die Business-CLAUDE.md im aktuellen Ordner (dein Geschäftsmodell).

Beides zusammen ergibt einen Kontext, den Claude sonst nie hätte.

### Schritt 2 • Das `raw/` - `wiki/` -System

Das Prinzip stammt von Andrej Karpathy, einem KI-Forscher. Einfache Idee:

- `raw/` — dorthin legst du alles Unverdaute: Artikel, Screenshots, Transkripte. Keine Sortierung.
- `wiki/` — dort entsteht die saubere Form. Jeder Artikel ein Thema.

Claude macht die Arbeit zwischen `raw/` und `wiki/`. Du gibst ihm einen Prompt wie diesen:

Lies alle Dateien in `raw/`, die seit dem letzten Eintrag in `wiki/log.md` dazugekommen sind.

Für jede Quelle:

1. Lege oder erweitere einen Wiki-Artikel in `wiki/<thema>.md`.
2. Fasse die wichtigsten Punkte zusammen, Quelle vermerken.
3. Verknüpfe mit `[[wiki-links]]` zu verwandten Themen.
4. Aktualisiere `wiki/index.md` als Inhaltsverzeichnis.
5. Trage die heutige Aktion in `wiki/log.md` ein.

**Was sind `[[wiki-links]]` ? Eine Besonderheit von Obsidian. Schreibst du irgendwo `[[DSGVO-Newsletter]]`, baut Obsidian automatisch einen klickbaren Link zur Datei `DSGVO-Newsletter.md`.**

## Schritt 3 • Ein Kunde als Beispiel

Lege im Ordner `04-crm/kunden/` eine erste Datei an: `001-mustername.md`.

```
# 001 Mustername

## Kontakt
- E-Mail: ...
- Telefon: ...

## Status
- Seit: 2024-03
- Typ: Bestandskundin (65 €/h)
- Letzter Kontakt: 2026-04-10

## Gesprächshistorie
- 2026-04-10 - Call: Neue Social-Posts, 30 Min
- 2026-03-22 - Mail: Rechnung März bezahlt

## Aktive Projekte
- Website-Refresh (Deadline 2026-05-15)

## Nächste Schritte
- [ ] 5 Instagram-Posts für Mai bis 2026-04-30
- [ ] Rechnung April rausschicken
```

## Schritt 4 • Tägliches Log

Lege in `00-daily/` eine Datei mit dem heutigen Datum an: `2026-04-18.md`. Kein festes Format, keine Pflicht zur Vollständigkeit. Drei Minuten am Tag reichen.

## Schritt 5 • Der Praxis-Test

```
cd ~/Documents/MeinBusiness  
claude
```

Prompt:

```
Lies alle Dateien in 04-crm/kunden/ und die letzten 7 Tage in 00-daily/.  
Beantworte: Welche Kunden brauchen diese Woche Aufmerksamkeit?  
Begründe jede Empfehlung mit einem konkreten Punkt aus den Dateien.
```

Wenn Claude mit konkreten Namen und Gründen antwortet — **dein Business hat ein Gedächtnis gewonnen**. Du musst nie wieder aus dem Kopf rekonstruieren, wo du bei welchem Kunden stehst.

---

## Checkliste für Tauchgang 01

- Claude Code ist installiert, `claude` startet ohne Fehler.
- Globale `CLAUDE.md` enthält dein Profil.
- Obsidian-Vault `MeinBusiness` mit Ordnerstruktur.
- Business- `CLAUDE.md` im Vault, mit deinen Leistungen und Preisen.
- Mindestens ein Kunde als Datei in `04-crm/kunden/`.
- Erstes Daily-Log in `00-daily/`.
- Claude beantwortet die Test-Frage mit echten Namen und Daten.

**Das ist das Fundament.** Die nächsten drei Tauchgänge setzen komplett darauf auf — wenn die Basis steht, geht alles Weitere schneller.

---

## Wenn etwas nicht klappt

- **command not found: claude** → Terminal schließen, neu öffnen, nochmal probieren.
- **Permission denied bei npm install -g** → Mit `sudo` davor: `sudo npm install -g @anthropic-ai/claude-code`.
- **Claude ignoriert deine CLAUDE.md** → Liegt die Datei wirklich unter `~/claude/CLAUDE.md`? Der Befehl `cat ~/claude/CLAUDE.md` zeigt dir den aktuellen Inhalt.
- **Obsidian findet den Vault nicht** → Beim Start „Open folder as vault“ und den richtigen Pfad auswählen.

---

## ***Links aus diesem Tauchgang***

- Claude: [claude.ai](https://claude.ai)
- Node.js LTS: [nodejs.org](https://nodejs.org)
- Obsidian: [obsidian.md](https://obsidian.md)

## BONUS

# ***Bonus: 8 sofort-einsetzbare Geschenk-PDFs***

Fertige Vorlagen und Prompt-Sets, die du direkt im Agentur-Alltag einsetzen kannst. PDFs herunterladen und abheften.

**01**

**5 KI-Prompts für Agenturen**

PDF · SOFORT EINSETZBAR

**02**

**Content-Brief Baukasten**

PDF · TEMPLATE

**03**

**Angebots-Vorlage**

PDF · § 19 USTG-READY

**04**

**Meeting-Protokoll**

PDF · STRUKTUR-PROMPT

**05**

**Mein KI-Stack**

PDF · TOOL-AUSWAHL

**06**

**30-Tage-Plan**

PDF · EINFÜHRUNGS-PLAN

**07**

**Style-Guide Baukasten**

PDF · MARKENTON-TEMPLATE

**08**

**KI-Audit Fragebogen**

PDF · FÜR KUNDENGESPRÄCHE

# ***Nächster Tauchgang: Die Werkzeuge.***

Slash-Commands, Skills, Subagents.

Einen Prompt einmal schreiben — tausendmal nutzen.

[sebastian-sperber.de/tauchgang](https://sebastian-sperber.de/tauchgang)

## ***Kurz-Anleitungen zu diesem Tauchgang***

Vier 45-Sekunden-Videos — jeweils ein konkreter Schritt zum Nachmachen.

**01**

### ***Claude Code installieren***

Node, npm, Claude Pro, Login

**02**

### ***CLAUDE.md schreiben***

Personal Memory File auf Root

**03**

### ***Obsidian-Vault aufsetzen***

iCloud-Pfad, Struktur,  
Kontext

**04**

### ***Memory-System aktivieren***

4 Speicher-Typen, Recall

---

© 2026 Sebastian Sperber

Mediengestalter · Art Director · Röthenbach a.d. Pegnitz  
Kleinunternehmer nach § 19 UStG · keine Mehrwertsteuer

[sebastian-sperber.de/tauchgang](https://sebastian-sperber.de/tauchgang)