

Auf einen Blick

TEIL I Coding mit KI-Unterstützung

1	Einführung	17
2	Pair Programming	63
3	Debugging	107
4	Refactoring	127
5	Software testen	145
6	Software dokumentieren	181
7	Datenbanken	199
8	Scripting und Systemadministration	227

TEIL II Sprachmodelle lokal ausführen, fortgeschrittene KI-Tools

9	Sprachmodelle lokal ausführen	261
10	Code automatisiert verarbeiten	295
11	Level-3-Tools: OpenHands und Aider	325
12	Retrieval Augmented Generation (RAG) und Text-to-SQL	351
13	Risiken und Ausblick	385

Inhalt

Vorwort	9
---------------	---

TEIL I Coding mit KI-Unterstützung

1 Einführung	17
<hr/>	
1.1 Coding per Chat	17
1.2 Die Kunst des Prompting	27
1.3 Code-Assistenten	35
1.4 Chat oder Assistent?	47
1.5 Grundlagen von Large Language Models (LLMs)	48
2 Pair Programming	63
<hr/>	
2.1 Code in Funktionen strukturieren	64
2.2 Beispiel: IBAN-Validierung	74
2.3 Objektorientiert programmieren	81
2.4 Beispiel: Quiz	90
2.5 Beispiel: Sudoku lösen	96
3 Debugging	107
<hr/>	
3.1 Web-Applikationen	108
3.2 App-Entwicklung	117
3.3 Entwicklung am Raspberry Pi	120
3.4 Visual Studio und VS Code	123
3.5 Fazit	125
4 Refactoring	127
<hr/>	
4.1 Einführung ins Refactoring	128
4.2 Refactoring mit KI-Werkzeugen	128

4.3	Best Practices	141
4.4	Fazit	143
5	Software testen	145
5.1	Erzeugen von Testdaten	146
5.2	KI-gestützte Testautomatisierung	150
5.3	Testgetriebene Entwicklung mit KI	155
5.4	Arbeiten mit Abhängigkeiten	162
5.5	Optimieren von Tests	168
5.6	E2E-Tests	175
5.7	Fazit	179
6	Software dokumentieren	181
6.1	Inline-Dokumentation	183
6.2	Funktions- und Klassendokumentation	186
6.3	API-Dokumentation	190
6.4	Veraltete Dokumentation	193
6.5	Fazit	196
7	Datenbanken	199
7.1	Datenbankdesign	199
7.2	SQL-Kommandos	210
7.3	Administration	218
7.4	Client-Programmierung	225
8	Scripting und Systemadministration	227
8.1	Scripting	228
8.2	Beispiel: Python-Script in PHP-Code umwandeln	235
8.3	Reguläre Muster	237
8.4	Systemadministration	241
8.5	Beispiel: wget-Script plus tmpfs-Konfiguration	249
8.6	GitHub Copilot und ChatGPT im Terminal aufrufen	254

TEIL II Sprachmodelle lokal ausführen, fortgeschrittene KI-Tools

9	Sprachmodelle lokal ausführen	261
9.1	Die Qual der LLM-Wahl	262
9.2	GPT4All	267
9.3	Ollama	271
9.4	Open WebUI für Ollama	276
9.5	Continue	284
9.6	Ollama-API	287
9.7	Tabby	290
9.8	Fazit	293
10	Code automatisiert verarbeiten	295
10.1	OpenAI-API	296
10.2	Ollama-API	308
10.3	Groq-API	311
10.4	Beispiel: Code automatisiert kommentieren	312
10.5	Beispiel: von Python 2 zu Python 3	321
11	Level-3-Tools: OpenHands und Aider	325
11.1	OpenHands	327
11.2	OpenHands in der Praxis	333
11.3	Aider	336
11.4	Aider in der Praxis	341
12	Retrieval Augmented Generation (RAG) und Text-to-SQL	351
12.1	Schnellstart RAG	352
12.2	Das viel-falter-Projekt	356
12.3	Dokumente laden	357
12.4	Index erzeugen	359
12.5	Vector-Store-Datenbanken	363
12.6	RAG-Abfragen	366
12.7	Text-to-SQL	371

13.1 Probleme und Einschränkungen beim Einsatz von KI-Tools	385
13.2 Beispielhaftes KI-Versagen	389
13.3 Ethische Fragen	400
13.4 Schlussfolgerungen und Ausblick	404
Index	409