

Quick Start for Text Work with Cline

Version 1; 02.07.2026, created with the help of AI

1. Introduction: What is Cline?

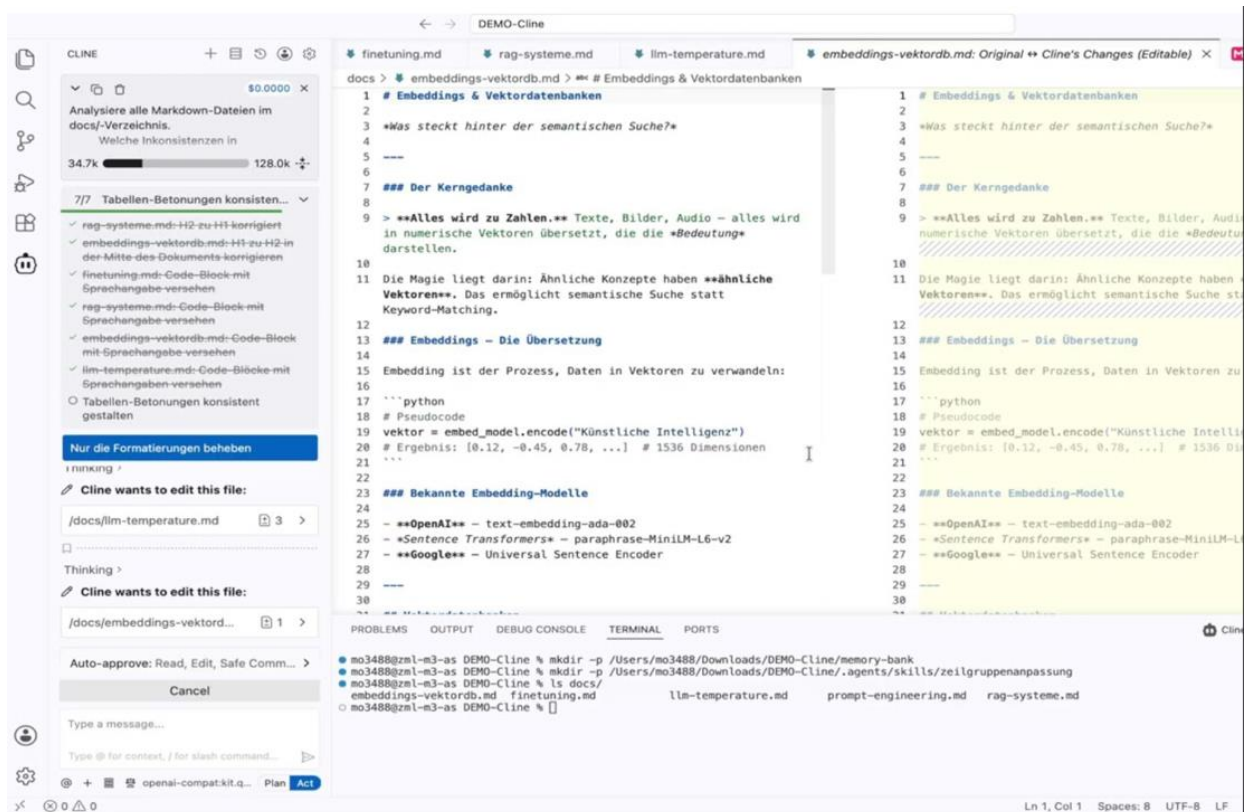
Cline is your AI-powered working partner for working with text documents. It can:

- Read, analyze, and edit files
- Execute structured processes
- Format texts according to your standards
- Preserve project knowledge across meetings

Important: You ALWAYS remain in control. Cline makes suggestions, you approve.

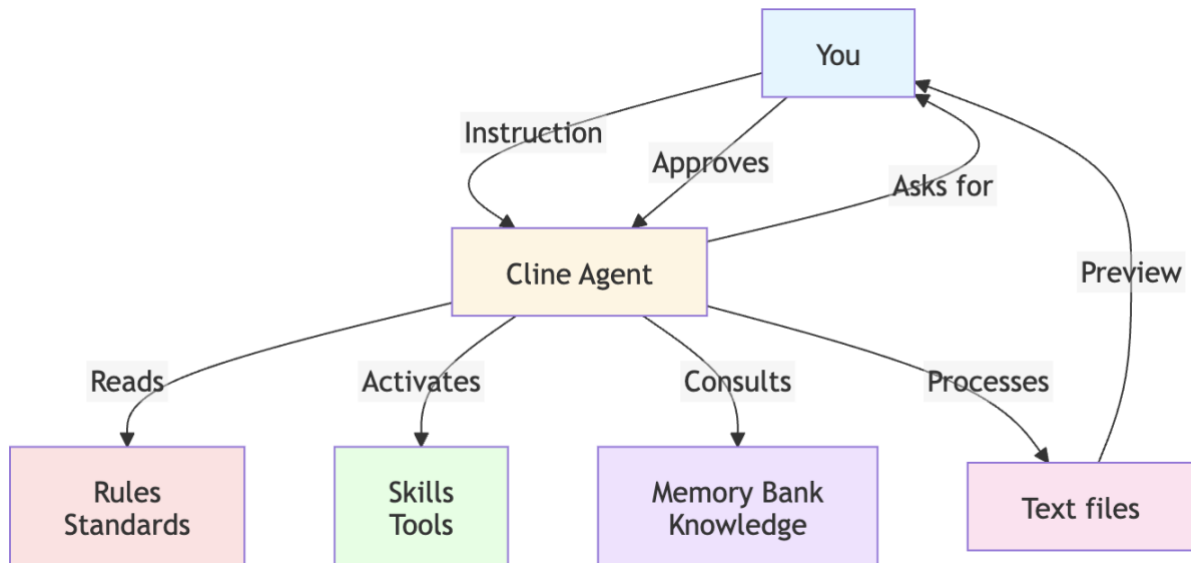
Video tutorial: Cline in action

To see the concepts from this guide live, watch [this video](#) . It demonstrates how to use Cline for text work – from request to approval:



Video Content: Plan Mode & Act Mode · Rules & Standards · Text formatting · Approval workflow

The Cline Ecosystem



Why Cline for text work?

Advantage	Description
 Speed	Processes dozens of files in minutes instead of hours
 Consistency	Consistent formatting across all documents
 Context	Understands relationships between documents
 Knowledge	Maintains project standards and decisions
 Workflows	Automate recurring tasks

2. Understand the basics

Before you can use Cline effectively, it's important to understand the basic concepts behind this AI-powered work partner. Cline is fundamentally different from classic chatbots: It does not act autonomously, but works together with you according to clear principles.

In this section, you'll learn the key concepts that shape your collaboration with Cline, from control over every step of the process to the right communication style.

The Human-in-the-Loop Principle

Cline works according to the human-in-the-loop principle: you always remain in control. Any action, especially high-risk operations such as deleting files or running programs, requires your approval. In the settings, you can individually specify which operations Cline should ask for.

 *Tip: You can pause, reject changes, or adjust the plan at any time. In the settings, configure which operations Cline should ask for.*

Plan & Act Mode – Thinking before Action

Cline has two modes that you can switch with one click:

Plan Mode (Thinking)

In **Plan Mode** , Cline analyzes and plans without making any changes – here it is not executed, but only thought/analyzed

What for: strategy, analysis, discussion

May:

-  Read and analyze files
-  Suggest structures
-  Ask questions
-  Develop plans

Can't:

-  Create or modify files
-  Execute commands

Typical use cases:

- "How should I structure this documentation?"
- "Analyze these 20 text files and suggest improvements"
- "What's the best approach to this problem?"

Act Mode (⚡ Act)

In **Act Mode** , Cline executes actions and prompts for risky operations (such as deleting files, executing programs)

What for: Implementation, processing, execution

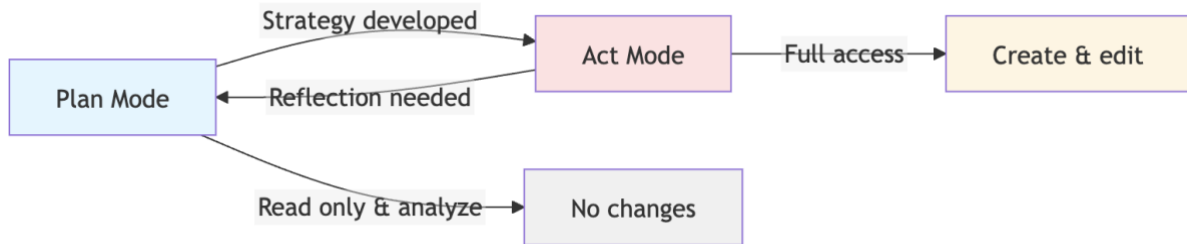
May:

-  Everything Plan Mode can do
-  Create and edit files
-  Execute commands
-  Perform workflows

Typical use cases:

- "Create a README.md with this structure"
- "Format all Markdown files according to the style guide"
- "Revise the introductory text"

 Important : *Even in Act Mode, you have to approve every action!*



Effective communication

The quality of the results depends directly on the quality of the requests. Cline is powerful – but it can't read your mind. The more precise, contextual and structured you formulate your instructions, the better Cline understands your intention and delivers appropriate results. Good communication with Cline follows simple principles that you should quickly internalize.

Good inquiries are:

-  **Specifically:** "Shorten the introductory text by 30%" instead of "Make it shorter"
-  **Contextual:** "For beginners with no previous knowledge" instead of just "Write a text"
-  **Step-by-step:** Breaking down large tasks into sub-steps

Example:

-  **Bad:** "Make the documentary better"
-  **Gut:** "Analyze the README.md and suggest improvements to the structure. Focus: Beginners should be able to start quickly."

Understanding context

Cline sees:

-  File structure in the project
-  Contents of files it reads
-  Rules, Skills and Memory Bank
-  Your conversation history (within the session)

Cline does NOT see:

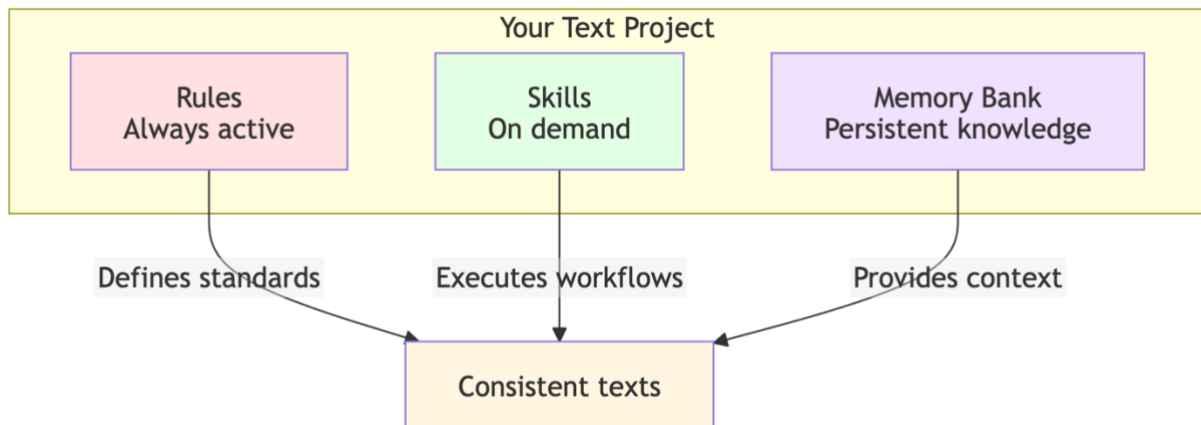
-  What's in your head
-  Unspoken expectations
-  Previous sessions (without memory bank)
-  External contexts outside the project

 *Tip: Be explicit! What seems obvious to you needs to be explained to Cline.*

3. The 3 pillars for text work

Now it's getting concrete: Cline uses three powerful mechanisms to deliver consistent, high-quality work while preserving your individual understanding of projects.

These three pillars – Rules, Skills and Memory Bank – form the foundation for professional text work with Cline. They differ in purpose, activation, and scope, but work together perfectly. Understanding these three components is key to transforming Cline from a helpful assistant to a valuable work partner.



Rules – Your permanent standards

Imagine if you could explain to Cline how texts should be formatted in your project – and Cline would remember these standards permanently. This is exactly what the so-called rules do: They are permanent instructions that are automatically taken into account in every interaction.

Whether it's formatting, language style or structure specifications – rules ensure that Cline works consistently according to your specifications without you having to give all the details again with every request.

Structure:

your-projekt/

```

|—— .clinerules/
|   |—— textstandards.md    # Formatting
|   |—— glossar.md          # Technical terms
|   |—— strukturvorgaben.md # Document structure
  
```

Example: text standards.md

German text standards

Language & style

- Language: German (Germany)
- Salutation: Sie-Form, professional
- Tone: Friendly but factual

Formatting

- German quotation marks: "Text" instead of "Text"
- Dashes: — (em-dash)
- Separate paragraphs with a blank line⁴
- No blank line after headings

Structure

- Every document starts with introduction
- Maximum heading level: H3
- Summary at the end
- Internal links for cross-references

Markdown conventions

- Headings in sentence notation
- Lists with minus signs (-)
- Code blocks with language identifiers
- Tables for comparisons

When to create rules:

Use case	Example
Formatting	Quotation marks, blank lines, Markdown style
Language style	Duzen/Siezen, technical terms, tonality
Structure	Document structure, headings, sections
Glossary	Uniform terms and spellings

💡 *Tip: One rule file per topic. This makes them easier to maintain and to switch on/off in a targeted manner.*

Skills – Task-related tools

While rules define how Cline works, skills determine what it should do. Skills are specialized instruction sets for specific tasks – from the review process to content creation to publication preparation. The ingenious thing is that they only charge when they are needed, thus keeping your work context lean.

With Skills, you can transform Cline from a general text assistant into a specialized expert for your individual workflows.

Difference to Rules:

	Rules	Skills
Activation	Always active	Only when needed
Scope	In short, standards	Detailed, Workflows
Purpose	How to work	What to do
Example	"Use German quotation marks"	"Carry out a review process"

Structure:

.cline/skills/

|—— text-review/

| |—— SKILL.md # Main Instructions

| └── docs/
| └── checkliste.md # Additional Resources
└── blog-publikation/
 └── SKILL.md

Example: text-review/SKILL.md

name: text-review
description: Performs a systematic review of text documents. Checks structure, language, formatting, and consistency.---

Text Review Skill

This skill performs a structured review process.

Procedure

1. **Check structure**

- Is there an introduction?
- Is the structure logical?
- Are important sections missing?

2. **Check language**

- Consistent salutation (you/you)?
- Technical terms used correctly?
- Tonality fits the target group?

3. **Check formatting**

- quotation marks correct?
- Headings consistent?
- Used Markdown correctly?

4. ****Check consistency****

- terms used consistently?
- Links work?
- Cross-references correct?

Output

Create a review report with:

- ☒ What works well
- ☐ What should be improved
- ☐ Concrete suggestions for improvement

When to create skills:

Use case	Example
Review processes	"Check document for completeness"
Content creation	"Create blog articles by template"
Workflows	"Prepare document for publication"
Special tasks	"Convert notes to articles"

 *Tip: Skills activate automatically if your request matches the skill description!*

Memory Bank – The Project Memory

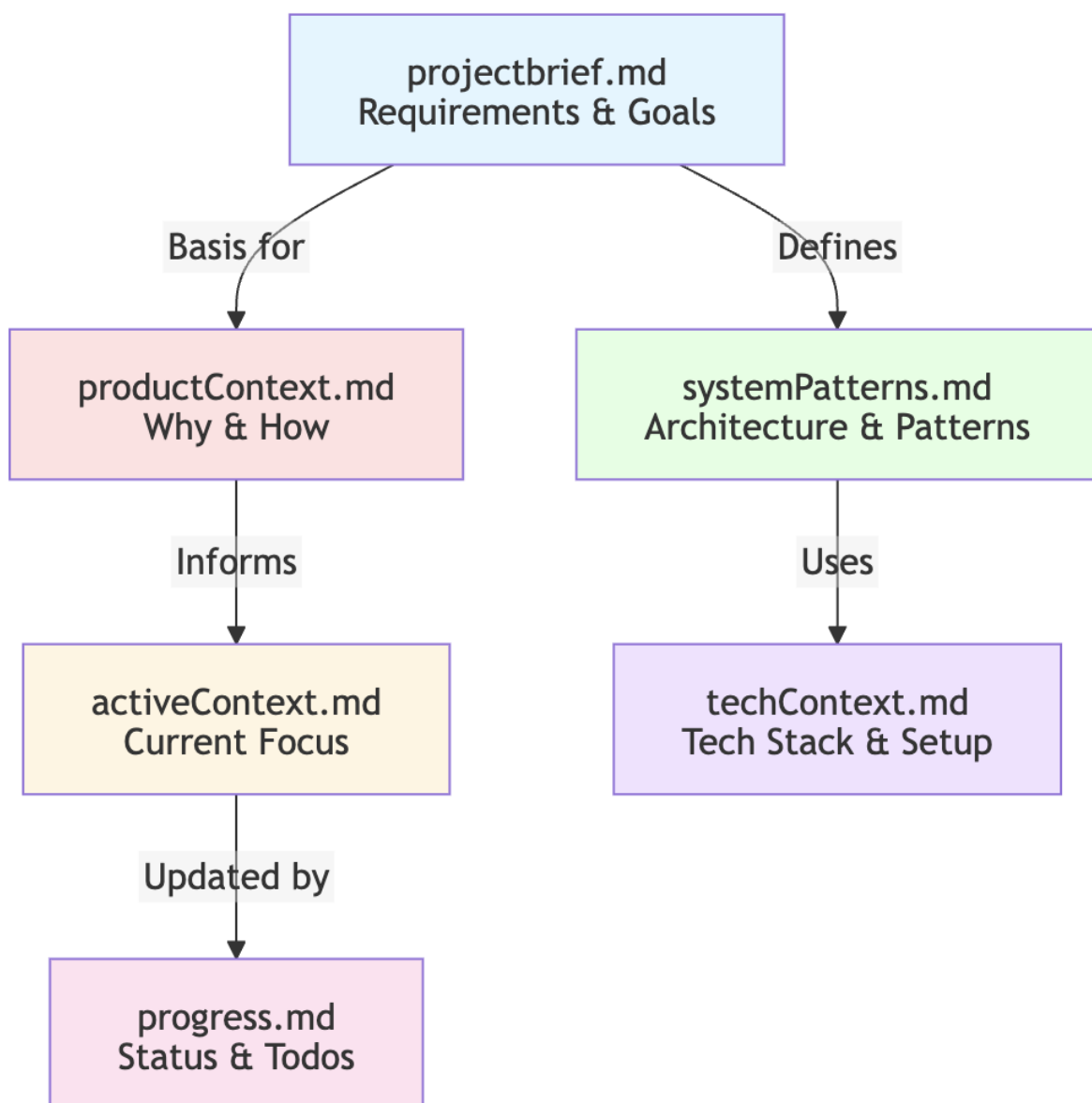
Here comes the most powerful component: Memory Bank solves one of the most fundamental problems of Agentic AI systems – forgetting. Without a memory bank, Cline loses the entire project context after each session. With Memory Bank, you can create a

structured, cross-session project memory. It documents not only what you do, but also why – and makes those decisions available to Cline (and your team) at any time.

For longer projects, Memory Bank is not optional, but essential. Without a memory bank, you have to start every new time:

- ❌ Re-explain the entire project
- ❌ Repeat all decisions
- ❌ Recommunicating standards
- ❌ Describe the current state

The 6 core files:



Structure:

memory-bank/

- |—— projectbrief.md # What & Why (Foundation)
- |—— productContext.md # Issues & Goals
- |—— activeContext.md # Current State (changes frequently)
- |—— systemPatterns.md # Architecture & Decisions
- |—— techContext.md # Technology & Tools |—— progress.md # Status & Todos

Example: Text Project activeContext.md

Active context

Current focus

- Revision of introductory texts for beginners
- Simplification of complex concepts
- Addition of practical examples

Recent changes

- Restructured README.md with a new structure
- Glossary expanded by 15 technical terms
- All code examples with syntax highlighting

Next steps

- [] Create tutorial section
- [] Add screenshots for UI guides

- [] Start English translation

Active decisions

- Address instead of informal form (target group: young developers)
- Practical examples more important than theoretical explanations
- Maximum 3 heading levels (H1-H3)

Important patterns

- Each section: Problem → solution → example
- Callout boxes for warnings and tips
- Use tables for comparisons

When to use Memory Bank:

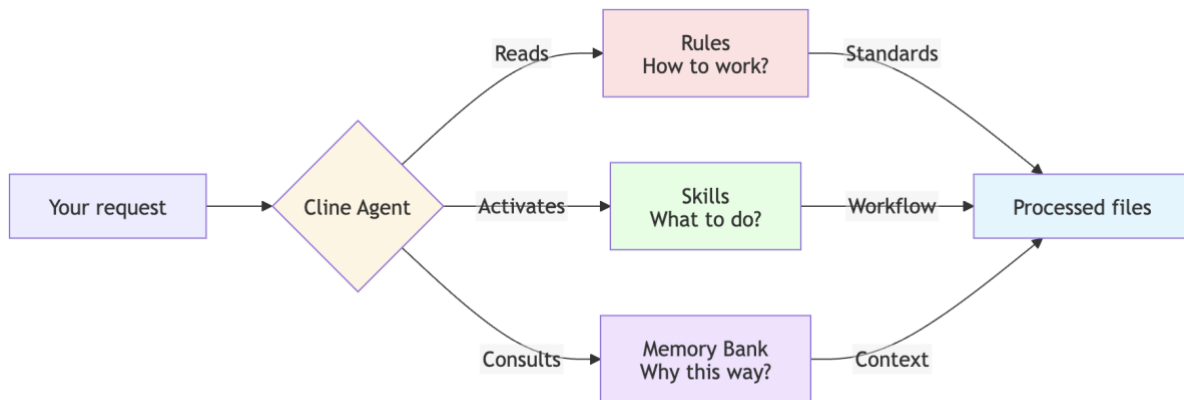
Scenario	Benefits
Longer projects	Context is maintained for weeks/months
Teamwork	Everyone has the same level of knowledge
Complex workflows	Decisions and justifications documented
Extensive documentation	Structure and standards centrally managed

Setup in 3 steps:

1. **Create directory:** memory-bank/
2. **Add Custom Instructions** (see [Official Instructions](#) at Cline)
3. **Hire Cline:** "Initialize Memory Bank for this Project"

Important: Memory Bank requires initial setup, but pays off for every longer project!

Interaction of the 3 pillars



4. Quick Start Tutorial

Theory is important – but now it's time to get down to the nitty-gritty. In this tutorial, we'll walk you through your first real text project with Cline step by step. You will get to know the typical workflow: from the initial analysis to the creation of rules to the automated processing of several files. By the end of this chapter, you will not only have understood how Cline works, but you will also have applied it practically.

Prerequisites

- ☒ VSCodium or Visual Studio Code (or other editor with Cline support)
- ☒ Cline Extension installed
- ☒ AI model configured (local model of the university, Claude, GPT, etc.)
- ☒ A directory of text files (Markdown, TXT, etc.) to work with

Project Setup Checklist

Setup steps

1. ☐ Open the project directory in Notepad
2. ☐ '.clinerules/' Create Folder
3. ☐ Open Cline extension in the sidebar
4. ☐ Create your first rule file
5. ☐ Optional: Set up Memory Bank

Tutorial: Your First Text Project

Now it's getting concrete: In this practical walkthrough, you will work on a realistic scenario that is typical for working with text documents. You have multiple Markdown files with inconsistent formatting and want to unify them—a task that would take hours to do manually. With Cline, you can do it in minutes.

Follow the five steps to get to know the complete workflow from analysis to automated processing.

Scenario: You have a directory of 10 Markdown files that you want to format and unify.

Step 1: Project Analysis (Plan Mode)

Action: Switch to Plan

ModeRequest: "Analyze all Markdown files in the docs/ directory.

What are the inconsistencies in formatting and structure?"

Cline will:

- Scan all files
- Identify inconsistencies
- Create a report

You will receive: Overview of problems (e.g. "5 files use wrong quotation mark format")

Step 2: Create Rules (Act Mode)

Action: Switch to Act Mode

Request: "Create a .clinerules/textstandards.md with the following standards:

- German quotation marks (""")
- Headings in sentence notation
- Blank line between paragraphs
- Code blocks with language identifiers"

Cline will:

- Create Rule File

- Submit for approval

You approve: "Create File"

Step 3: Format Files

Request: "Format all Markdown files in the docs/ directory according to the standards in .clinerules/textstandards.md"

Cline will:

- Create a plan ("I'm going to edit 10 files...")
- Waiting for your approval

You approve: "Continue"

Cline edited: Each file individually, shows you the changes

Step 4: Conduct a review (Plan Mode)

Action: Switch to Plan Mode

Request: "Check the formatted files. Are all standards met?"

Cline will:

- - Confirm successful changes
- - Report any problems

Step 5: Create Memory Bank (Optional)

Action: Stay in Plan Mode

Request: "Create a memory bank for this project. Document the formatting standards and the current status."

Cline will:

- memory-bank/ Create directory
- 6 Creating Core Files
- Populate with project information

Practical Commands & Patterns

For text analysis:

"Analyze [file/directory] and list all [headings/links/technical terms]"

"Find all places where [term] is used"

"Create a summary of [document]"

For editing:

"Format [file] to [Standard/Rule]"

"Shorten [section] by [X]%"

"Replace [term A] with [term B] in all files"

"Add [item] to all [file type] files"

For workflows:

"Conduct a review of [document]"

"Prepare [document] for publication"

"Create a [document type] by [template]"

For Memory Bank:

"Update Memory Bank" (after major changes)

"What is the current status of the project?" (reads memory bank)

"Continue the Work" (uses Memory Bank context)

Getting Started Quick Reference

I would like to...	Recommended Fashion	Sample Request
Analyze	Plan Mode	"Analyze the structure of all docs"
Plan	Plan Mode	"How should we organize the documentary?"
Edit	Act Mode	"Format all MD files"
Workflow	Act Mode	"Carry out a review process"
Setup	Act Mode	"Create rules for text standards"

4. Next steps

Congratulations – you've mastered the quick start and now understand the basics of working with Cline! What you have learned here forms the solid foundation for professional and efficient text work. But that's just the beginning: Cline offers a host of advanced features and capabilities to help you increase your productivity even further.

This section will show you how to proceed – from in-depth resources to common beginner mistakes to concrete next steps.

You can now:

-  Use Cline effectively for text work
-  Use Plan & Act Mode in a targeted manner
-  Create rules for standards
-  Use skills for workflows
-  Set up a memory bank for project knowledge

Deepening Resources

Official documentation:

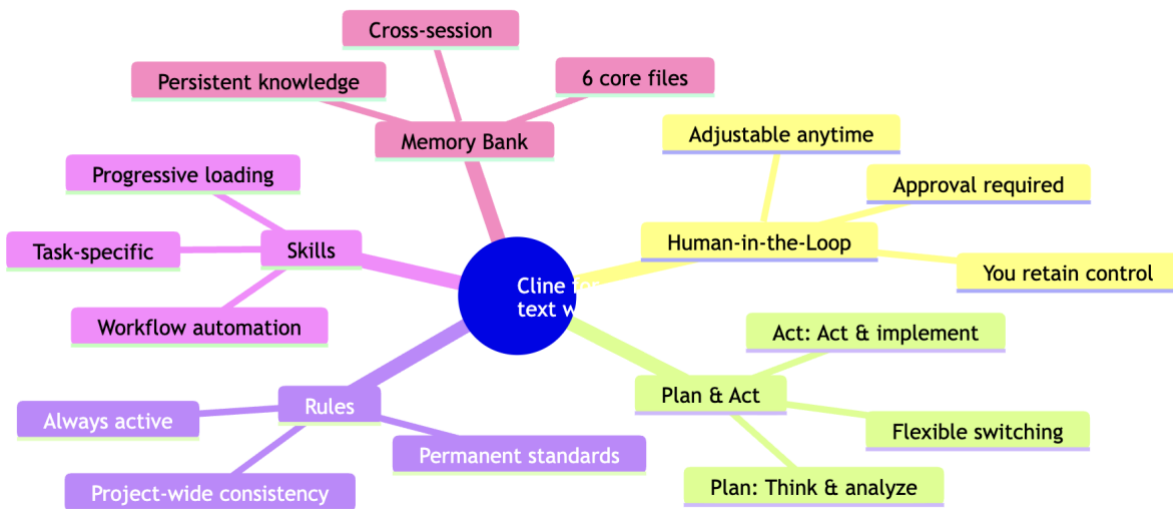
- [Cline Docs](#) - Full Documentation
- [Customization Guide](#) - Rules, Skills, Workflows
- [Memory Bank Guide](#) - Detailed Guide
- [Core Workflows](#) - Plan & Act Mode

Avoid common beginner mistakes

 Errors	 Better
"Do it better"	"Improve the structure: Add subheadings"
Too many files at once	Start with 1-3 files to test
Change rules after each task	Rules stable, adjust only when necessary
Do not update Memory Bank	After important changes: "Update Memory Bank"

Conclusion

The 5 Core Concepts



The most important finding

Cline is not an autopilot, but your co-pilot.

They set the direction, Cline supports the implementation. The quality of the results depends directly on the quality of your instructions and the structure of your project.

Tip: Start with a small project, experiment with rules and skills, and expand your use gradually.

Info & Contact

License Notice



This manual from the Center for Medial Learning (ZML) at the Karlsruhe Institute of Technology (KIT) is licensed under a Creative Commons Attribution 4.0 International License.

Imprint

Publisher: Karlsruhe Institute of Technology (KIT) Kaiserstraße 12 76131 Karlsruhe

Contact: InformatiKOM Adenauer Ring 12 76131 Karlsruhe Germany Phone: +49 721 608-48200 E-mail: info@zml.kit.edu

