Let Claude think (chain of thought prompting) to increase performance

Copy page

While these tips apply broadly to all Claude models, you can find prompting tips specific to extended thinking models here.

When faced with complex tasks like research, analysis, or problem-solving, giving Claude space to think can dramatically improve its performance. This technique, known as chain of thought (CoT) prompting, encourages Claude to break down problems step-by-step, leading to more accurate and nuanced outputs.

Before implementing CoT

Why let Claude think?

- **Accuracy:** Stepping through problems reduces errors, especially in math, logic, analysis, or generally complex tasks.
- **Coherence:** Structured thinking leads to more cohesive, well-organized responses.
- **Debugging:** Seeing Claude's thought process helps you pinpoint where prompts may be unclear.

Why not let Claude think?

- Increased output length may impact latency.
- Not all tasks require in-depth thinking. Use CoT judiciously to ensure the right balance of performance and latency.

Use CoT for tasks that a human would need to think through, like complex math, multi-step analysis, writing complex documents, or decisions with many factors.

Claude Docs

How to prompt for thinking

The chain of thought techniques below are **ordered from least to most complex**. Less complex methods take up less space in the context window, but are also generally less powerful.

> **CoT tip**: Always have Claude output its thinking. Without outputting its thought process, no thinking occurs!

- **Basic prompt**: Include "Think step-by-step" in your prompt.
  - Lacks guidance on *how* to think (which is especially not ideal if a task is very specific to your app, use case, or organization)

  > **Example: Writing donor emails (basic CoT)**

- **Guided prompt**: Outline specific steps for Claude to follow in its thinking process.
  - Lacks structuring to make it easy to strip out and separate the answer from the thinking.

  > **Example: Writing donor emails (guided CoT)**

- **Structured prompt**: Use XML tags like `<thinking>` and `<answer>` to separate reasoning from the final answer.

  > **Example: Writing donor emails (structured guided CoT)**

Examples

> **Example: Financial analysis without thinking**

> **Example: Financial analysis with thinking**

Claude Docs

Prompt library

Get inspired by a curated selection of prompts for various tasks and use cases.

GitHub prompting tutorial

An example-filled tutorial that covers the prompt engineering concepts found in our docs.

Google Sheets prompting tutorial

A lighter weight version of our prompt engineering tutorial via an interactive spreadsheet.

Claude Docs

Solutions

AI agents

Code modernization

Coding

Customer support

Education

Financial services

Government

Life sciences

Partners

Amazon Bedrock

Google Cloud's Vertex AI

Learn

Blog

Catalog

Courses

Use cases

Connectors

Customer stories

Engineering at Anthropic

Events

Powered by Claude

Service partners

Startups program

Claude Docs

Careers

Economic Futures

Research

News

Responsible Scaling Policy

Security and compliance

Transparency

Help and security

Availability

Status

Support

Discord

Terms and policies

Privacy policy

Responsible disclosure policy

Terms of service: Commercial

Terms of service: Consumer

Usage policy