

Prompt Engineering Field Guide

Practical techniques for anyone who works with AI tools

5 Modules | Cheat Sheet | Ready-to-Use Templates

The Mental Model

Every prompt is a job description. Whether you are asking an AI to write, analyze, code, or summarize, the quality of your output depends almost entirely on the quality of your input. Most weak results come from prompts that are missing one or more of four essential components:

Component	What It Answers
Role	Who is the AI being? What expertise should it apply?
Task	What exactly should it do? (Be specific and concrete)
Context	What background does it need to do the job well?
Output Format	How should the response look? (Length, structure, tone)

Key insight: Most advanced prompting techniques are just more precise ways of filling in these four components. Start here before reaching for anything else.

Quick Example

Weak prompt:

Summarize this article.

Strong prompt (all four components present):

You are a financial analyst. Summarize the following article for a non-technical executive audience. Highlight the 3 most important business implications. Use plain language, no jargon. 150 words max.

The second prompt has all four components. The first has none. That gap explains most of the variation people experience in AI output quality.

Patterns That Actually Matter

These five techniques cover the vast majority of real-world prompting needs. Each one solves a specific, named problem.

Chain of Thought

Force reasoning before conclusions. Prevents the AI from jumping to an answer without working through the problem.

```
Think through this step by step before giving me your final answer.
```

Best for: Complex decisions, debugging, multi-variable analysis

Role Priming

Set expertise before the task. The AI calibrates its response to the role you assign.

```
You are a senior options trader specializing in iron condor strategies.  
Analyze this trade setup and identify key risks.
```

Best for: Domain-specific advice, professional-grade output

Constraint Injection

Define what NOT to do. Especially useful when the output feeds into another system.

```
Return only the JSON object. No markdown. No introductory text. No  
explanation.
```

Best for: API integrations, automations, structured data pipelines

Few-Shot Examples

Show the pattern instead of describing it. Works when the desired format is easier to demonstrate than explain.

```
Here are two examples of how I want this formatted: [example 1] / [example  
2]. Now do the same for: [new input].
```

Best for: Consistent formatting, style matching, repetitive output tasks

Persona + Audience

Calibrate complexity and tone by naming your target audience.

Explain this to a non-technical business owner who has never used software beyond a spreadsheet.

Best for: Client communications, training materials, documentation

The Iterative Refinement Loop

The most important skill in prompting is not writing the perfect prompt on the first try – it is knowing how to diagnose and fix a bad output. Use this loop:

Step	Action
1	Run the prompt
2	Identify the failure type (see diagnostic table below)
3	Change ONE variable and re-run
4	Repeat until the output meets your standard

Diagnosing Failure Types

Symptom	Fix
Output is too vague or generic	Add a role and domain context
Format is wrong	Specify the exact format at the end of your prompt
Part of your request was ignored	Number your requirements (1, 2, 3...)
Response is too long or too short	State the target length explicitly
It made something up (hallucinated)	Ask it to only use information you provide, or to cite its reasoning
Inconsistent results across runs	Use lower temperature or add more specific constraints

Pro tip: Keep a running prompt library – a simple document where you save prompts that worked and note why. After a few weeks you will have a personal toolkit worth more than any course.

Advanced Techniques

These techniques are worth knowing but are not needed for most everyday tasks. Reach for them when the basics are not enough.

Prompt Chaining

Break complex workflows into sequential prompts where the output of step 1 feeds step 2. More reliable than cramming a multi-step task into a single long prompt. Especially effective for automation pipelines.

Structured Output Enforcement

When integrating AI output into an app or workflow, explicitly request a schema. Example: "Return your answer as a JSON object with these keys: name, date, amount, status." Then validate the output programmatically before using it.

Self-Critique Loop

After getting a response, ask the AI to evaluate its own work: "Now review your response above. Identify any weaknesses or gaps, then rewrite it." This single step often produces meaningfully better output.

System vs. User Prompt

When working with AI via API, the system prompt sets persistent behavior — persona, tone, rules. The user prompt handles dynamic input. Keep them separate and well-defined for consistent results at scale.

Temperature Awareness

Temperature controls randomness. Higher = more creative and varied. Lower = more deterministic and consistent. Use low temperature for factual tasks and structured output. Use higher temperature for brainstorming and creative work.

Ready-to-Use Templates

Copy, adapt, and use these templates immediately. Each follows the four-component structure from Module 1 and incorporates the patterns from Module 2. Replace the bracketed placeholders with your specifics.

Process Documentation

You are a business process analyst. Interview me about [process name] by asking one question at a time. After each answer, ask a follow-up that clarifies who does it, when, and what tool they use. Stop after 10 questions and summarize the process as a numbered workflow with roles assigned to each step.

Data Analysis

You are a data analyst. I will provide a dataset below. Identify the top 3 trends, flag any outliers or anomalies, and suggest one actionable recommendation. Format your response as: Trends / Anomalies / Recommendation. Dataset: [paste data]

Email or Document Draft

You are a professional business writer. Draft a [type of email/document] from [sender role] to [recipient role]. Tone: [e.g., formal, friendly, direct]. Goal: [what you want the reader to do or understand]. Key points to include: [1, 2, 3]. Length: [word count or description].

Technical Explanation

You are a senior [engineer / developer / analyst]. Explain [concept or system] to an audience familiar with technology but with no background in [specific domain]. Use an analogy to make it concrete. Keep it under 200 words.

Structured JSON Output

You are a data extraction assistant. Extract the following fields from the text below and return them as a JSON object: [field1], [field2], [field3]. If a field is not present, set its value to null. Return only the JSON. No explanation. Text: [paste input]

One-Page Cheat Sheet

When a prompt is not working, start here.

Prompt Anatomy

Component	Question It Answers	If Missing...
Role	Who is the AI?	Output will be generic, not expert-level
Task	What exactly should it do?	Output will be vague or off-target
Context	What does it need to know?	Output will miss important nuance
Output Format	How should the response look?	Output will be inconsistently structured

Fix-It Reference

Problem	Fix
Too vague	Add role and domain context
Wrong format	Specify format explicitly at the end of the prompt
Ignored part of request	Number your requirements (1, 2, 3...)
Too long or too short	State target length explicitly
Hallucinated facts	Ask it to only use information you provide
Inconsistent results	Lower temperature and add tighter constraints
Complex multi-step task failing	Chain prompts – do not cram everything into one
Good prompt, bad style	Add few-shot examples showing the style you want

5 Patterns at a Glance

Pattern	When to Use It
Chain of Thought	Complex reasoning, debugging, multi-step logic
Role Priming	You need expert-level domain calibration
Constraint Injection	Output feeds into an app, API, or automation
Few-Shot Examples	Format is easier to show than describe
Persona + Audience	Tone and complexity must match a specific reader

Created with Perplexity Computer | perplexity.ai