



AI Prompt Engineering Cheatsheet

This cheat sheet provides a structured reference for generating effective AI prompts. It is organized to help you quickly recall key concepts, techniques, and troubleshooting tips. Each section includes a short tutorial description, drawing from AI prompt engineering best practices. Use this as a reference when working with LLMs like ChatGPT, Claude, Gemini, or Copilot.

Six Key Elements of a Good Prompt

1. Instruction

Clearly state what you want the AI to do. Be explicit about the expected output.

2. Context

Provide background details to help the AI understand the situation.

3. Role

Assign the AI a role (e.g., Cybersecurity Analyst, Business Consultant) to frame its responses.

4. Formatting

Specify output format (bullet points, tables, narrative) to ensure readability.

5. Tone

Indicate the style or tone (professional, casual, tutorial) to match the audience.

6. Examples

Provide examples of effective responses to inform the AI's output style and depth.

Prompting Strategies

Zero-Shot Chain of Thought Prompting

Ask the AI to explain its reasoning without providing examples. Use cues like 'Let's go step-by-step.'

Example: 'How many hours are in a year? Let's go step-by-step.'

Few-Shot Prompting

Provide a few examples of desired input-output pairs to guide the AI's behavior.

Example: 'Translate these phrases: Hello → Hola, Goodbye → Adiós. Now translate: Thank you.'

Few-Shot Chain of Thought Prompting

Combine examples with explicit reasoning steps to improve logical accuracy.

Example: 'Example 1: $2+2=4$ (reasoning: basic addition). Example 2: $5+3=8$ (reasoning: basic addition). Now solve: $7+6$.'

Generated Knowledge

Ask the AI to first research or brainstorm before giving a final answer.

Example: 'Research three cybersecurity trends and summarize them.'

Least-to-Most Prompting

Break complex problems into smaller sub-problems for sequential solving.

Example: 'Step 1: Identify phishing email indicators. Step 2: Explain how to block them.'

Emotional Prompting

Encourage confidence in responses (e.g., 'You are an expert in...').

Example: 'You are a confident, world-class security analyst. Explain how to detect malware.'

Self-Consistency Prompting

Ask for multiple reasoning paths and select the most consistent result.

Example: 'Provide three different methods for securing a Wi-Fi network, and choose the best one.'

Role Prompting

Use one or more roles to frame different parts of the AI's task.

Example: 'Act as a cybersecurity auditor and review this security policy.'

Adjusting Temperature

Lower temperature for factual, consistent output; higher for creativity.

Example: 'At temperature 0.2: Summarize the NIST CSF in bullet points.'

Troubleshooting AI Responses

Hallucinations

Fact-check all AI outputs. Cross-reference with reliable sources.

Example: AI states, 'NIST SP 800-99 exists.' Verify and correct (it does not exist)

Bias

Randomize examples and avoid repetitive patterns to reduce bias influence.

Example: AI favors certain countries in answers, randomize example inputs.

Conflicting Outputs

Use self-consistency prompting or ask the AI to explain discrepancies.

Example: AI gives two different incident response steps in separate runs, ask it to reconcile.

Over-Talkative Responses

Limit its scope and request concise answers. Gen AI is a people pleaser and tends to over-explain.

Example: AI writes a 1,000-word answer for a yes/no question. Limit its scope and remind it to be concise.

Lack of Specificity

Reinforce instructions and provide context/examples.

Example: AI says, 'Use security controls.' Request specific control names and frameworks.

Advanced Techniques

Retrieval Augmented Generation (RAG)

Enhance responses with external factual sources.

Self-Evaluation Prompting

Ask the AI to review and improve its own answer before finalizing.

Debiasing Prompts

Structure questions to challenge assumptions and remove bias.

Workflow for Prompt Development

1. Draft initial prompt using the Six Key Elements.
2. Select an appropriate prompting strategy.
3. Test and refine through trial and error.
4. Fact-check outputs and verify against requirements.
5. Iterate until results are consistent and useful.