AIFreeForever

The OpenAI Prompt Guide for 2026

OpenAI models answer what you ask them, in ChatGPT or through the OpenAI API. A clear prompt says the task, the context, who the answer is for, the tone and the format. This guide covers the parts of a strong prompt, an expanded prompt template, system and user messages for the API, how to ask for JSON, the task types with examples, and a checklist to fix a weak answer.

The parts of a strong prompt

1. **Clear instructions.** One action word and exactly what to do: write, summarize, explain, analyze or answer.
2. **Proper context.** The facts the model needs: your topic, your goal and the text or data to work on.
3. **Audience.** Who reads the answer: a student, a developer, a child, an executive or a general reader.
4. **Tone.** Friendly, formal, educational, conversational, neutral or creative.
5. **Format specification.** Paragraphs, a list, steps, a table, JSON or a dialogue, and how long.
6. **Constraints.** What to leave out, such as "no facts that are not in the text".

The expanded prompt template

Task: _____ (write / summarize / explain...) _____.
Context: _____.
Audience: _____.
Tone: _____.
Format: _____ of about _____ words.
Constraints: Do not _____.

System and user messages for the API

In ChatGPT you type one message. Through the OpenAI API, a developer sends a system message that sets the model's role, rules and answer format, then a user message with the request. Keep the rules in the system message, so every request gets answers in the same shape.

System: You are an assistant that summarizes meeting notes for busy managers. Always answer in JSON with the keys "decisions", "open_questions" and "next_steps", each a list of short strings. Never add facts that are not in the notes.

User: Summarize these meeting notes: [paste the notes here]

Asking for JSON

1. Name every key the JSON must have, and what goes in each.
2. Say "answer with the JSON only, no text before or after it".
3. Give one short example of the JSON you want when the shape is unusual.
4. Check the answer in your code before you use it.

The task types and what to add

Task	Add to your prompt
Writing	The length, the reader and the main points
Code Generation	The language, the input and output, and any error message
Summarization	The text itself, the number of points, and "no new facts"
Explanation	The reader's level and "use one everyday example"
Q&A	The question, the reader, and how long the answer can be
Analysis	The data or text, what to look for, and "as a table"
Instructional	The goal, the reader's starting point, and "as numbered steps"

Ten ready prompts to try

1. Write a bedtime story about a robot that learns to dream, for a 6-year-old, in under 400 words, with a gentle ending.
2. Write a 600-word blog post for small business owners about why a simple website still matters, with three short headings.
3. You are a senior Python developer. Write a function that totals monthly sales from a CSV file, with type hints and comments, then explain it in three sentences.
4. Summarize the report I paste below for an executive in 5 bullet points. Start with the most important finding and add no new facts.
5. Explain how vaccines work to a 12-year-old in two short paragraphs with one everyday comparison.
6. Answer this for a child in under 120 words: why do volcanoes erupt? Then give two follow-up questions with short answers.
7. Analyze the customer reviews I paste below. Group them into themes in a table with a count for each, then give the three changes that would help most.
8. Write a step-by-step guide for setting up a first budget spreadsheet, for a student, in 8 steps or fewer.
9. Return a JSON object with the keys "title", "summary" and "tags" for the article I paste below. Answer with the JSON only.
10. System: You are a friendly tutor who answers in short numbered steps. User: How do I find the area of a triangle?

Fix a weak answer

1. **Too general?** Add your own facts and your goal.
2. **Wrong shape?** Name the format, or move it into the system message.
3. **Wrong level?** Name the audience.
4. **Different every time?** Keep the same system message and ask for JSON with named keys.
5. **Still not right?** Test another phrasing and keep the version that works best.

© 2026 AIFreeForever