

Prompt Engineering Fundamentals

Step 1: Know what a good prompt contains

A strong prompt usually has 4 parts:

- Goal – what you want
- Context – background info + constraints
- Format – how you want the output (bullets, table, code, etc.)
- Checks – ask it to verify, ask questions, or show assumptions

Bad prompt:

“Teach me Excel.”

Better prompt:

“Teach me Excel basics for beginners. I have 1 hour/day for 7 days. Give a day-by-day plan with exercises and expected outcomes.”

Step 2: Use the simplest reliable template (copy/paste)

Use this whenever you’re unsure:

You are: [role]

Task: [what to do]

Context: [my situation + details]

Constraints: [limits: time, tools, level]

Output format: [bullets/table/steps]

Quality bar: [be concise, include examples, etc.]

Example

“You are an IT trainer. Task: teach me prompt engineering. Context: I’m a beginner. Constraints: 15 minutes per lesson. Output: step-by-step with 2 practice prompts each. Quality bar: simple language.”

Step 3: Add “role + audience” to steer tone and depth

This changes the quality a lot.

“Explain like I’m 12” → simpler

“Teach like a university lecturer” → deeper

“Act as a tutor and quiz me” → interactive

Try this prompt

“Act as a friendly tutor. Teach me prompt engineering like I’m a beginner. Ask me 1 question after each concept.”

Step 4: Give examples or data (garbage in, garbage out)

Chatbots respond best when you provide input to work with.

Instead of: “Write an email.”

Do: “Write an email declining a meeting. Tone: polite and firm. Recipient: my manager. Reason: schedule conflict. Offer two alternative times.”

Step 5: Control the output with formatting instructions

If you don’t specify format, you’ll often get long, mixed answers.

Useful format commands:

“Use bullet points only”

“Create a table with columns: ...”

“Give me 3 options: casual / professional / formal”

“Keep it under 120 words”

Step 6: Ask for thinking without asking for hidden reasoning

You can improve quality by asking for:

- assumptions
- steps
- checklist
- pros/cons
- risk flags

Good:

“List assumptions you’re making.”

“Provide a checklist I can follow.”

“Give pros/cons and a recommendation.”

Step 7: Iterate: prompt engineering is a loop

Use this loop:

1. First draft prompt
2. See output
3. Improve with: “revise based on these changes: ...”

Example:

“Revise to be shorter, add examples, and use a table.”

More Examples

Studying / Learning Something New

Beginner Prompt

“Teach me SQL.”

Improved Prompt

“Teach me SQL basics for beginners. Assume I have no background. Use simple language and give examples.”

Pro Prompt (Recommended)

“You are a patient tutor. Teach me SQL fundamentals for beginners.

Context: I’m new to databases.

Constraints: 20 minutes per lesson.

Output: step-by-step explanation, one example per concept, and one practice question.

After each lesson, quiz me with 2 questions.”

Work Email / Professional Writing

Beginner Prompt

“Write a professional email.”

Improved Prompt

“Write a professional email asking for a meeting next week.”

Pro Prompt

“Write a professional email requesting a meeting.

Context: I’m asking my manager for a 30-minute meeting.

Purpose: discuss project status.

Tone: polite and confident.

Constraints: under 120 words.

Output: subject line + email body.”

Prompt tip

Always specify recipient + tone + purpose + length for writing tasks.

Coding / Technical Help

Beginner Prompt

“Fix my code.”

Improved Prompt

“Fix this Python code and explain the error.”

Pro Prompt

“Act as a senior developer.

Task: debug the following Python code.

Context: beginner-level script.

Output:

1. identify the error
2. explain why it happens in simple terms
3. provide corrected code
4. give one best practice tip.”

Golden rule for coding prompts

Always paste:

- the code
- the error message
- what you expected to happen

Lesson Plan / Training Content

Beginner Prompt

“Create a lesson plan.”

Improved Prompt

“Create a lesson plan on AI for beginners.”

Pro Prompt

“You are an IT instructor.

Task: create a beginner-friendly lesson plan on AI basics.

Audience: non-technical learners.

Duration: 2 hours.

Output: objectives, agenda, activities, and assessment questions.

Teaching style: interactive and practical.”

Universal “Fix My Prompt” Prompt

Use this when results aren't good:

“Improve the previous response by making it simpler, more structured, and adding examples. Ask me clarifying questions if needed.”