INNOVATION AI LAB

INO AI LAB

[AI EDUCATION • COURSE MATERIAL]

Prompt Engineering Masterclass

Advanced techniques for ChatGPT, Claude, Gemini

LEVEL	DURATION	LESSONS
Intermediate	**4 hours**	**8**

Innovation AI Lab — Tools, Prompts & Performance

// 2026 EDITION

[00]

Table of Contents

[LESSON 01]

The Anatomy of a Production Prompt

A weak prompt asks a question. A production prompt sets up a system. The five-part structure works for 90% of cases: Role ('You are a senior B2B SaaS copywriter'), Context ('We sell PM software to 50–500-engineer teams'), Task ('Write three launch subject lines'), Constraints ('Under 50 chars, no emojis, no exclamation marks'), and Format ('Numbered list with a one-line rationale').

This structure removes ambiguity and makes outputs reproducible. When something is wrong, you can pinpoint which part failed. Save your best prompts as templates with variables and version them like code. Teams that treat prompts as engineered artifacts — reviewed, tested, versioned — get 3–5x more value than teams that just chat with the bot.

// KEY TAKEAWAYS

- Five parts: Role, Context, Task, Constraints, Format.
- Save and version prompts like code.
- Reproducibility comes from explicit constraints.

[LESSON 02]

Few-Shot Learning: Teach by Example

Models learn faster from examples than from descriptions. Paste two or three input-output pairs you want, then ask for a fourth. This few-shot pattern dramatically increases consistency, especially for formatting, tone, and edge cases that are hard to describe in words. It's the single fastest way to improve a stubborn prompt.

The sweet spot is 3–5 examples. One is rarely enough; ten burns context. Diversify examples — show easy AND tricky cases — so the model learns the full pattern. For classification, cover every class. For generation, span the variation you want in production. Audit examples regularly; stale demonstrations silently degrade output quality over time.

// KEY TAKEAWAYS

- 3–5 diverse examples is the sweet spot.
- Include edge cases, not just typical ones.
- Few-shot beats long explanations for style.

[LESSON 03]

Chain-of-Thought & Reasoning Modes

Asking a model to 'think step by step' before answering improves accuracy on math, logic, and multi-step problems by 20–40%. Modern reasoning models (o3, Claude Opus with thinking, Gemini 2.5 Pro) do this automatically in a hidden scratchpad. For non-reasoning models, trigger it explicitly: 'Work through this carefully. Show reasoning, then give the final answer on a new line starting with FINAL:'.

Use reasoning modes for math, code debugging, legal analysis, and multi-document synthesis. Skip them for simple lookups, summarization, and casual chat — they triple latency and cost for no quality gain. The skill is matching the cognitive load of the task to the cognitive mode of the model. Measure both quality and cost when you switch modes.

// KEY TAKEAWAYS

- 'Think step by step' boosts reasoning accuracy.
- Reserve reasoning models for genuinely hard tasks.
- Match cognitive mode to task difficulty.

[LESSON 04]

Structured Output, Guardrails & Chaining

For anything that feeds another system, demand structured output. Use JSON mode or strict tool schemas (OpenAI, Gemini, Claude) and define every field, type, and required flag. When JSON mode isn't available, wrap output in XML tags like <answer>...</answer> — it's more robust because the model can't break the structure with an unescaped quote. Always validate parsed output and retry with the error message included.

Layer guardrails for safety-critical apps: a system prompt with hard limits, an output validator, and a separate judge model that flags risky outputs. Split complex tasks into chains (research → outline → draft → edit) but don't over-chain — each step adds latency, cost, and a failure point. Log every intermediate output so you know which link broke.

// KEY TAKEAWAYS

- Use JSON mode or strict schemas when available.
- Layer guardrails: prompt + validator + judge.
- Chain prompts, but only when quality measurably improves.

[NEXT]

Keep Going

You've completed this course material. The real learning starts when you apply what you've read. Pick one idea from this PDF and run a small experiment this week. Document what worked and what didn't. Share your findings with the community.

Explore more free courses, daily AI tips, and curated tools at:

innovationailab.com

Have feedback or want to suggest a topic? We read every message.

hello@innovationailab.com

— Innovation AI Lab Team —

// part of LumiLife Tech