

## Education & Learning / Tutoring

Create step-by-step worked examples with commentary — cognitive load reduction for complex problem-solving.

Difficulty: Intermediate

Model: GPT-4 / Claude / Gemini

Use Case: Skill Demonstration, Problem-Solving

Updated: June 2026

Why This Prompt Exists

Novices don't know how experts solve problems. Worked examples reveal the hidden steps. Most textbooks show the answer, not the thinking process.

You get:

- students who can't replicate expert problem-solving
- missing intermediate steps (gaps in understanding)
- no explanation of why each step is taken
- students memorizing solutions instead of understanding process
- inability to solve similar problems independently

But worked examples have structure:

- problem statement: what needs to be solved
- step-by-step solution: each action clearly shown
- commentary: why each step is taken (not just what)

- common pitfalls: what to watch for
- self-explanation prompts: questions for the learner

Without worked examples, students don't learn process.

This prompt designs worked examples with explanatory commentary.

The Prompt

Assume the role of an instructional designer who creates worked examples.

Your task is to demonstrate problem-solving steps with explanatory commentary.

Generate:

#### 1. EXAMPLE OVERVIEW

- Problem: [full problem statement]
- Prerequisite knowledge: [what learner should already know]
- Learning objective: [what this example teaches]

#### 2. STEP-BY-STEP SOLUTION

**\*\*Step 1: [Action description]\*\***

- What: [specific operation performed]
- Why: [reason for this step]
- How: [method or formula used]
- Result: [intermediate outcome]

**\*\*Step 2: [Action description]\*\***

- What: [specific operation performed]
- Why: [reason for this step]
- How: [method or formula used]
- Result: [intermediate outcome]

[Continue for all steps]

**\*\*Final Step: [Solution]\*\***

- Result: [final answer]

### 3. SOLUTION CHECK (verification step)

- How to verify: [method to check answer]
- Verification: [applying the check]

### 4. COMMON PITFALLS (what could go wrong)

Step	Potential Error	Why It Happens	How to Avoid
[step]	[mistake]	[reason]	[strategy]

### 5. SELF-EXPLANATION PROMPTS (for learner)

- Prompt 1: "Why did we [action] instead of [alternative]?"
- Prompt 2: "What would happen if we [changed variable]?"
- Prompt 3: "How is this similar to [related problem]?"

### 6. VARIATION PRACTICE (similar problem for learner)

Problem: [parallel problem with same structure]

Hint: [optional guidance]

INPUTS:

Problem to demonstrate:

[PASTE THE PROBLEM]

Subject area:

[E.G., "Algebra", "Physics", "Statistics", "Chemistry"]

Audience level:

[BEGINNER / INTERMEDIATE / ADVANCED]

Specific technique to highlight:

[E.G., "Factoring quadratics", "Unit conversion", "Hypothesis testing"]

RULES:

- Show every step (don't skip intermediate calculations)
- Explain why each step is taken (not just what)
- Include a verification step (checking the answer)
- Anticipate common errors at each step
- Use self-explanation prompts (active learning)
- Provide a parallel practice problem (transfer)
- Keep commentary concise but informative

How To Use It

- Show every step — don't skip intermediate calculations; novices need to see the full path.

- Explain why each step is taken — not just what; the reasoning is what transfers.
- Include a verification step — checking the answer reinforces understanding.
- Anticipate common errors at each step — address pitfalls before they happen.
- Use self-explanation prompts — ask learners why steps work, don't just tell them.
- Provide a parallel practice problem — transfer requires similar but different problems.
- Keep commentary concise but informative — too much text overwhelms.

Example Input

**Problem to demonstrate:** “Solve for x:  $2(x + 3) = 10$ ”

**Subject area:** “Algebra”

**Audience level:** “BEGINNER”

**Specific technique to highlight:** “Solving linear equations with parentheses”

Why It Works

Experts solve problems automatically — skipping steps novices need. Worked examples reveal the hidden process.

This framework improves outcomes by forcing: step-by-step demonstration, reasoning commentary, verification method, pitfall anticipation, self-explanation prompts, and variation practice.

**Failure modes this prevents:** Missing intermediate steps, no reasoning explanation, memorized solutions without understanding, inability to transfer.

**This improves on:** Answer-only solutions. Worked examples teach process, not just results.

**Related to:** TU-01 (Socratic Questions) for guided practice; TU-06 (Transfer Problems) for application.

## Build Better AI Systems

Subscribe for advanced prompt engineering, AI coding tools, debugging frameworks, and practical strategies for developers and engineers.

Carefully engineered prompts for people doing real work.