

Advanced Techniques Combination

Part 3 Mastery: Integrating Multiple Prompt Strategies

```
def advanced_completion(task, context, examples=None):
    # System message (Role) + Few-shot (Examples) +
    Chain-of-thought
    system_msg = "You are an expert technical writer.
    Think step-by-step."

    user_prompt = f"""Context: {context}

    {'Examples: ' + examples if examples else ''}

    Task: {task}

    Please:
    1. Analyze the requirements
    2. Consider the context
    3. Apply best practices
    4. Provide a detailed response

    Let me work through this step by step:"""

    return litellm.completion(
        model="gpt-3.5-turbo",
        messages=[
            {"role": "system", "content": system_msg},
            {"role": "user", "content": user_prompt}
        ],
        temperature=0.3,
        max_tokens=400
    )
```

⌚ 25-30 minutes | Lab Step 14/17

☰ Technique Stack

- ✓ System Messages (Role Definition)
- ✓ Few-Shot Learning (Examples)
- ✓ Chain-of-Thought (Reasoning)
- ✓ Parameter Tuning (Optimization)

⊗ Mastery Challenge

Create a technical documentation assistant using all four techniques combined in the master prompt template.

💡 Best Practices

Combine techniques for complex tasks requiring nuance and structure. Be mindful of over-prompting, which can confuse the model or exceed token limits.

🌟 Advanced Mastery