ChatGPT Part 8: Multi-Step Workflows and Prompt Chaining

For a complex job, one long prompt often produces a vague or incomplete answer. Prompt chaining can work better: we split the job into steps, and the result of one step becomes the input of the next. Four habits make a reliable workflow: split, chain, structure and check. Each habit answers one simple question, and in this part, Hadeel, an HR coordinator at a training company, shows how it works.

1. Split

A complex job becomes easier when it is cut into stages with one clear goal each. The question is: what are the stages, and what does each one produce? Typical stages are collect, organise, draft and review. Writing the plan down first, even as a short list, saves time later. Smaller steps are easier to check, and when something goes wrong, we know which step to fix.

Example: A student has to write a history essay. She uses the Split question: what are the stages, and what does each one produce? She plans four steps: choose the angle, make an outline, write a draft and proofread it.

At work: Hadeel must prepare a one-day onboarding course for new employees. She uses the Split question: what are the stages, and what does each one produce? She plans four steps: list the topics, build an outline, draft the slide text and write a short quiz.

2. Chain

Chaining means that each step receives the useful output of the previous one. The question is: what does this step hand to the next? We paste the approved result into the next prompt, say clearly what it is, and then give the next instruction. In a long conversation, ChatGPT may lose track of earlier details, so restating the key points keeps the chain firm. Many teams also save the prompt of each step in a document, so the workflow can be repeated.

Example: The student has an outline from the first step. She uses the Chain question: what does this step hand to the next? She pastes the approved outline into the next prompt and asks for a draft of section one only.

At work: Hadeel has a list of eight topics. She uses the Chain question: what does this step hand to the next? She pastes the final topic list into the next prompt and writes: build a day plan from these topics, with breaks.

3. Structure

A step is easier to reuse when its output has a clear shape. The question is: what format does the next step need? We can ask for a table, a numbered list or JSON, a text format with named

This document is a property of Asaseeyat Academy - © Copy Right Protected.

fields that programs can read. Fixed column or field names make results easy to compare, copy into a spreadsheet and feed into the next prompt. If the format matters, we show a short example of it.

Example: The student wants to compare her sources. She uses the Structure question: what format does the next step need? She asks for a table with the columns title, author, year and main point.

At work: Hadeel wants to load the course plan into a spreadsheet. She uses the Structure question: what format does the next step need? She asks for a table with the columns session, topic, minutes and trainer, and later reuses it to request matching quiz questions.

4. Check

A mistake in an early step travels to every later step. The question is: is this result good enough to go on? Between steps, we check facts, completeness and format, and we can ask ChatGPT itself to review its answer against a short checklist. Three or four checklist points are usually enough. The final judgement stays with a person, especially for facts, numbers and anything about people.

Example: The student's draft quotes a date and a statistic. She uses the Check question: is this result good enough to go on? She verifies both in a reliable source and corrects one date before the next step.

At work: The quiz questions are ready. Hadeel uses the Check question: is this result good enough to go on? She asks ChatGPT to compare each question with the course topics, then reads them herself and removes one unclear question.

Conclusion

In summary, a multi-step workflow turns one big prompt into a chain of small, checkable steps. We split the job, chain the results, give each result a clear structure and check before going on. Because tools and names change over time, we keep our own list of steps and prompts. Before each step, the key question is simple: what does the next step need from this one?

Sources

- OpenAI Platform Docs: Prompt engineering (strategies, including splitting complex tasks into simpler subtasks) – https://platform.openai.com/docs/guides/prompt-engineering
- OpenAI Platform Docs: Structured Outputs (responses that follow a JSON format) – https://platform.openai.com/docs/guides/structured-outputs
- OpenAI Help Center: Prompt engineering best practices for ChatGPT – https://help.openai.com/en/articles/10032626-prompt-engineering-best-practices-for-chatgpt

- ChatGPT and OpenAI are trademarks of OpenAI. This video is independent training material and is not affiliated with OpenAI.

This document is a property of Asaseeyat Academy - © Copy Right Protected.