

The Writer and the Critic: prompts to paste

AI Administrator · Day 5 · Session 9 · Multi-Step and Multi-Agent Workflows

Participant copy-paste sheet

Two **Basic LLM Chain** nodes, two prompts. Paste each one into the node's **Prompt** field with *Define below* selected, and make sure the field is in **Expression** mode (the braces turn green) so the `{{ $json... }}` parts are filled in. AI Caravan Ltd. is fictional.

Both prompts are the module's, with three additions found while testing on 21 October 2026: the writer sees the **previous draft**, the critic sees the **purpose** (without it the critic called the task itself “invented”), and the critic has a **sixth check** (does the draft do the job?), without which an impossible request was “approved” because the writer simply left the impossible part out.

1. The Writer (Gemini 2.5 Flash, temperature 0)

Write a professional email to the supplier below on behalf of the Finance team of AI Caravan Ltd.

Use only the facts given. Do not promise payment dates or amounts not listed. Under 150 words. Sign off as "Finance Operations".

If reviewer feedback is present, revise the previous draft to address every numbered point.

Reply with the email body only: no subject line, no commentary.

Supplier: `{{ $json.supplier }}`

Purpose: `{{ $json.purpose }}`

Facts: `{{ $json.facts }}`

Deadline: `{{ $json.deadline || 'none given' }}`

Reviewer feedback: `{{ $json.feedback || 'none' }}`

Previous draft: `{{ $json.draft || 'none' }}`

2. The Critic (Gemini 2.5 Pro, temperature 0: a different and stronger model than the writer)

You review outgoing supplier emails for the Finance team. Check the draft against this list:

1. Uses only the facts provided; nothing invented.
2. No payment date or amount promised unless in the facts.
3. Polite, professional, under 150 words.
4. Mentions the deadline if one was given.
5. No phone numbers, home addresses or other personal data; naming the people involved is fine.
6. Does what the purpose asks; an email that avoids the task is not finished.

Reply with exactly APPROVED if all six pass. Otherwise reply with REVISE: followed by numbered points, each naming the failing item and the fix.

Purpose of the email (the task the writer was given): {{ \$json.purpose }}

Facts: {{ \$json.facts }}

Deadline: {{ \$json.deadline || 'none given' }}

Draft: {{ \$json.draft }}

Why the critic answers in a fixed form

The **IF** node after the Review log checks one thing: *verdict starts with APPROVED*. Everything else is a REVISE, and the numbered points go back to the writer as feedback. A critic that answers in prose (“Looks mostly fine, but...”) breaks the loop; the fixed form is what makes two models work as one process.

The hand-off fields

Both chains output only their text. The two **Edit Fields (Set)** nodes after them carry the task and the state forward:

Field	Set where	Carries
supplier, supplier_email, your_email, purpose, facts, deadline	Start, then Keep draft and Verdict (from the Start node)	the task and the constraints
revision (Number)	Start = 0; Next revision = revision + 1	the loop cap
feedback	Start = empty; Next revision = the critic's verdict	the critic's points for the writer
draft	Keep draft = the writer's text	the current draft
verdict	Verdict = the critic's text	APPROVED or REVISE: ...

Task, facts, constraints: anything not carried is lost. That is the module's hand-off rule, in node form.

