

## REPO SUMMARY

# ComfyUI PromptStyler

Version 3.1.1. Custom ComfyUI node pack that applies curated style templates to a positive prompt and returns conditioning for downstream generation. It is structured around JSON style packs plus lightweight Python runtime code that composes the final prompt text at node execution time.

**WHO IT'S FOR**

Primary persona: ComfyUI users building image-generation workflows who want reusable, curated prompt styling without hand-writing long style suffixes.

**INSTALL SHAPE**

Drop the repo into  
ComfyUI/custom\_nodes/ComfyUI\_PromptStyler  
and reload custom nodes.

**WHAT IT DOES**

- Exposes a ComfyUI node named PromptStyler: Prompt - > Conditioning (Style Picker).
- Loads style templates from styles/packs/\*.json in filename order.
- Falls back to styles/styles\_v1.json if packs are missing or unusable.
- Presents styles as categorized dropdown choices in Category | Name | id format.
- Supports direct style\_id\_override and a template\_variant selector.
- Builds a deduplicated styled prompt, then tokenizes and encodes it through the connected CLIP text encoder.
- Ships maintainer tools to validate, audit, generate, sync, and add style packs.

**HOW TO RUN**

- Clone or copy this folder into ComfyUI/custom\_nodes/ComfyUI\_PromptStyler.
- Restart ComfyUI or use "Reload custom nodes".
- Add CheckpointLoaderSimple, PromptStyler, and KSampler to a workflow.
- Wire CLIP -> text\_encoder, your prompt text into prompt, and positive -> KSampler.positive.
- Optional repo validation: run tools/validate\_styles.py; audit with tools/audit\_styles.py.

**HOW IT WORKS**

- \_\_init\_\_.py is the ComfyUI entry point: it reads VERSION and exports node mappings.
- nodes.py defines the runtime node that accepts the user prompt, style selection inputs, and a ComfyUI CLIP text encoder connection.
- The node scans styles/packs/, merges JSON packs in sorted order, or falls back to the legacy single-file library.
- Selected style prefix/suffix phrases are combined with the user prompt, normalized, and de-duplicated before encoding.
- The encoded result is returned as CONDITIONING plus the exact styled\_prompt string for inspection/debugging.
- style\_library.py, tests, and maintainer docs describe broader search/overlay logic, but an advanced node implementation was **Not found in repo** in the current nodes.py.
- .github/workflows/ci.yml runs version sync, style validation, style audit, and unit tests on pushes and pull requests.

If you need external services, hosted APIs, or a packaged installer: **Not found in repo**.