STRETCHLEARN · WORKBOOK

Stable Diffusion & Open-Source AI Art — Workbook

This workbook turns the course into reps you can do at your own machine or in Leonardo.ai. Each section maps to one course module and mixes guided exercises, fill-in worksheets, and checklists so you build a repeatable, documented practice. Work through it with your generator open beside you, and save every prompt, seed, and setting as you go.

Getting Started with Stable Diffusion

Choose your runtime, install it, and prove the pipeline works with a verified first image.

Worksheet: Your Setup Profile

Fill this in honestly before installing anything. It decides which tool and base model you should start with and prevents wasted downloads.

GPU make and model (or 'none')

VRAM in GB (dedicated graphics memory, not system RAM)

Operating system

Chosen runtime: Automatic1111 / ComfyUI / Leonardo.ai

Reason for that choice (hardware, tinkering preference, or no GPU)

Recommended starting base model: SD 1.5 / SDXL / SD3.5

Install location path (keep it short, e.g. C:\AI\sd)

Exercise: Verified First Image

Run the exact known-good test from the course so you have a baseline to compare every later result against. Do not change models or add LoRA yet.

• Generate: a photo of a red apple on a wooden table, soft natural light, sharp focus. Settings: steps 25, CFG 7, sampler DPM++ 2M Karras, 512x512 (SD 1.5) or 1024x1024 (SDXL).

• Record where the output saved and what the console printed as the active device (GPU name, not CPU).

• If the image is black, errored, or melted, note which troubleshooting fix from the lesson resolved it.

• Save the apple image and its full settings as your baseline reference.

Checklist: Install Readiness

[ ] Python 3.10.x installed with Add to PATH ticked (local installs only)
[ ] Git installed (local installs only)
[ ] Runtime downloaded and unzipped to a short path
[ ] First launch completed and browser address opened (127.0.0.1:7860 or :8188)
[ ] At least one base model checkpoint present in the models folder
[ ] Verified first image generated and saved as baseline

Prompts, Models, and Core Settings

Build a reusable prompt structure and learn the five settings that make output repeatable.

Worksheet: Structured Prompt Builder

Compose one prompt by filling each slot of the course skeleton, then assemble them in order. Keep the most important descriptors first.

Subject (concrete detail)

Action or pose

Setting / background

Style and medium

Quality and lighting modifiers

Negative prompt terms to avoid

Assembled full positive prompt

One term you will weight, e.g. (red dress:1.3), and why

Exercise: Isolate One Variable

Practice the change-one-thing-at-a-time habit using a locked seed so you can attribute every difference to a single setting.

• Generate any image you like on an SDXL checkpoint at steps 28, CFG 6, sampler DPM++ 2M Karras, seed -1.

• Copy the seed from the output and paste it back into the seed field to lock it.

• Change ONLY CFG from 6 to 8 and regenerate; write down what changed in the image.

• Now reset CFG to 6 and change ONLY steps from 28 to 18; write down the difference in fidelity.

Worksheet: Five-Dial Settings Log

For one image you are happy with, record all five core settings so you can reproduce it exactly later.

Checkpoint model name and version

Sampling steps

CFG scale

Sampler

Seed (the locked number, not -1)

VAE used (external name or 'baked-in')

Resolution

Checklist: Settings Mastery Check

[ ] I can explain what steps, CFG, sampler, seed, and VAE each do
[ ] I loaded an external VAE to fix washed-out SD 1.5 color (if applicable)
[ ] I can reproduce an image by matching checkpoint, prompt, seed, steps, CFG, and sampler
[ ] I confirmed my base model generation before downloading any add-ons
[ ] I only download models from reputable sources and prefer .safetensors files

Customizing with LoRA, Embeddings, and ControlNet

Add precise style, subject, and composition control with LoRA, embeddings, and ControlNet.

Worksheet: LoRA Stack Planner

Plan a multi-customization image before generating so the combined strength stays under control and you can debug it.

LoRA 1 name and base-model generation it requires

LoRA 1 trigger word(s) from the model card

LoRA 1 starting strength (0.7-0.9)

LoRA 2 name (if stacking) and strength (drop toward 0.5-0.7)

Negative embedding to include (e.g. EasyNegative for SD 1.5)

Base checkpoint these all sit on

Predicted issue if combined strength is too high

Exercise: ControlNet Pose Transfer

Reproduce the course's worked example: keep a pose from a reference photo while replacing the subject entirely.

• Pick a clear single-person reference photo and enable ControlNet with the OpenPose preprocessor and OpenPose model.

• Set control weight to about 1.0 and prompt for a different subject in that pose, e.g. a knight in ornate steel armor, dramatic lighting.

• Generate, then note whether the pose held; if ignored, raise control weight, if it looks traced, lower toward 0.6.

• Repeat once with the Depth or Canny preprocessor and record how the captured feature differs.

Checklist: Customization Troubleshooting

[ ] LoRA matches the base model generation and includes its trigger word
[ ] Combined LoRA strength reduced when the image looked burnt
[ ] Negative embedding placed in the negative prompt box, not the positive
[ ] ControlNet model matches the base model generation
[ ] Correct ControlNet preprocessor chosen for the feature I want to keep
[ ] Control weight tuned between roughly 0.6 and 1.2 for a natural result

Refining, Scaling, and Production Workflows

Fix flaws, upscale to high resolution, build a reusable pipeline, and handle rights and export.

Exercise: Inpaint and Upscale a Keeper

Take one image you like through the full finishing pipeline: repair, then enlarge without losing identity.

• Send the image to Inpaint, mask only a flawed area (e.g. a hand), prompt a relaxed human hand, five fingers, denoising 0.4-0.6, and regenerate.

• Run hires fix or the Extras upscaler at 1.5x-2x with a sharp upscaler like 4x-UltraSharp and denoising 0.25-0.4.

• Confirm faces did not change identity after upscaling; if they did, lower the upscale denoising.

• If you ran out of VRAM, redo the enlargement with Ultimate SD Upscale tiling.

Worksheet: Rights and Delivery Record

Before using any image commercially, complete this licensing and export record for the exact stack that made it.

Base model and its license (e.g. SD3.5 Community License — does the revenue threshold apply to me?)

Checkpoint model card commercial-use allowed? (yes/no/conditions)

Each LoRA's commercial-use status from its model card

Confirmed no living-artist-by-name or trademarked-character prompts for commercial use (yes/no)

Master saved as PNG with embedded parameters (yes/no)

Delivery export format and size (e.g. JPEG 1080x1350 for Instagram portrait)

Reproduction log saved: prompt, seed, model, LoRA (yes/no)

Checklist: Production-Ready Sign-Off

[ ] Generated at native resolution, then upscaled (not rendered huge directly)
[ ] Faces and hands repaired via inpainting or ADetailer
[ ] Reusable ComfyUI workflow saved as JSON (if using ComfyUI)
[ ] License checked for base model and every LoRA in the stack
[ ] Master saved as PNG; delivery copy exported at destination size
[ ] Prompt, seed, model, and LoRA logged for reproduction

Your Action Plan

1. Complete your Setup Profile worksheet and install the matching runtime (Automatic1111, ComfyUI, or sign up for Leonardo.ai).
2. Generate the verified first image (red apple, steps 25, CFG 7) and save it as your baseline.
3. Build three prompts with the Structured Prompt Builder and run the isolate-one-variable exercise to feel how CFG and steps change output.
4. Download one reputable .safetensors checkpoint for your goal (e.g. an SDXL photorealism or anime model) and confirm its base generation.
5. Add one LoRA at strength 0.8 with its trigger word, and one negative embedding if on SD 1.5; log the stack.
6. Install ControlNet and complete the OpenPose pose-transfer exercise, then repeat with Depth or Canny.
7. Run one image through inpainting to fix hands or faces, then upscale it 2x with a sharp upscaler at low denoising.
8. If using ComfyUI, wire and save a reusable workflow JSON; if using Automatic1111, save your full settings preset.
9. Complete the Rights and Delivery Record for one image you intend to publish or sell.
10. Export a master PNG and a sized delivery copy, and start a running reproduction log for your best images.