



CREATIVE & DESIGN TOOLS

Raster Editing Fundamentals

Photoshop & GIMP, Compared Throughout

Selections, Layers & Masks

Color Correction & Basic Retouching

Capstone: One Photo, Edited End to End in Both Tools

Philip Osztromok

Generated with Claude

Table of Contents

Raster Editing Fundamentals — 10 Chapters

CHAPTER	TITLE
Chapter 1	Two Tools, One Discipline
Chapter 2	The Raster Canvas
Chapter 3	Selections: The Foundation of Everything
Chapter 4	Layers: Non-Destructive Stacking
Chapter 5	Layer Masks: Non-Destructive Hiding, Not Deleting
Chapter 6	Adjustment Layers & Color Correction
Chapter 7	The Clone Stamp & Healing Tools
Chapter 8	Basic Filters & Effects
Chapter 9	Exporting & File Formats
Chapter 10	Capstone: Editing a Real Photo End to End in Both Tools

Raster Editing Fundamentals

Chapter 1 · Two Tools, One Discipline

This course teaches raster image editing as a discipline, using Photoshop and GIMP as two parallel, independently-built implementations of it. Before any of that, it's worth being precise about what "raster editing" actually means, and what this course deliberately isn't.

Raster vs. Vector: A Genuinely Different Kind of Image

A **raster** image is a fixed grid of individual pixels — resolution-dependent, meaning scaling it up beyond its original size degrades quality, since there's simply no more detail to reveal. A **vector** image instead stores mathematical paths and shapes, scaling to any size with no quality loss at all. This course covers raster editing specifically — vector tools like Illustrator and Inkscape belong to a genuinely different discipline, covered separately (Vector Graphics, still on this site's own bucket list) rather than being a different tool for the identical job.

Two Tools, One Discipline

Unlike the JetBrains family (Text Editors & IDEs Survey's own Chapter 4), Photoshop and GIMP share no underlying platform at all — they're separate, unrelated products built by entirely different teams. What they genuinely share is the underlying *discipline*: both operate on the same kind of image, a grid of pixels, and both solve the same categories of problem — selections, layers, masks, color correction — independently. This course teaches that discipline once per concept, then shows it concretely in both tools.

Proprietary vs. Free/Open-Source

Photoshop is Adobe's own commercial product, sold via a Creative Cloud subscription, and remains the industry standard across many professional contexts. **GIMP** is free and open-source, developed by a community rather than a single company. This is a real, practical difference worth knowing upfront — not a technical footnote to skim past.

How This Course Is Structured

Each concept chapter teaches one idea conceptually first, then shows it concretely in both tools with terminology mapped between them — Photoshop's own name for a feature alongside GIMP's own name for the equivalent concept. The capstone (Chapter 10) applies every concept from this course to one real photo, edited end to end, with both tools' own workflows shown side by side at each step.

ASPECT	PHOTOSHOP	GIMP
Cost model	Commercial, Creative Cloud subscription	Free, open-source
Development	Adobe (single company)	Community-developed
Native format	PSD	XCF
Industry standing	Widely considered the standard	A genuine, capable alternative

WHY RASTER VS. VECTOR MATTERS BEYOND THIS COURSE

The raster/vector distinction this chapter draws has a direct, practical web implication: a logo should almost always be exported as vector (SVG), since it needs to scale cleanly across sizes with no quality loss, while a photo should almost always stay raster (JPEG/PNG), since it's fundamentally pixel-based data to begin with. Choosing the wrong format for the wrong kind of image is a genuinely common, avoidable mistake.

GIMP IS NOT "A FREE COPY OF PHOTOSHOP"

It's tempting to assume GIMP simply reproduces every Photoshop capability under different button names. This undersells real differences this course covers honestly throughout — non-destructive editing maturity (Chapter 6), scripting language and paradigm (imageedit2's own Chapter 6), and plugin ecosystem cost (imageedit2's own Chapter 9). GIMP is a genuinely different, independently-developed piece of software that happens to solve many of the same problems — not a clone attempting to match Photoshop feature-for-feature.

Hands-On Exercises

EXERCISE 1

A company logo needs to appear on a business card, a billboard, and a website favicon — three wildly different sizes. Using this chapter's own material, explain whether it should be created and stored as a raster or vector image, and why.

 [View solution](#)

EXERCISE 2

Explain why Photoshop and GIMP are described in this chapter as sharing "a discipline" rather than sharing "a platform," and why that distinction matters for how this course is structured.

 [View solution](#)

EXERCISE 3

A colleague says "GIMP is basically just free Photoshop, so there's nothing really to learn separately about it." Using this chapter's own warning box, explain what's wrong with this framing.

 [View solution](#)

CHAPTER 1 QUICK REFERENCE

- **Raster** — a fixed pixel grid, resolution-dependent. **Vector** — mathematical paths, scales infinitely. This course covers raster only.
- Photoshop and GIMP share **a discipline**, not a platform — independently built, solving the same problems separately
- **Photoshop** — commercial, Creative Cloud subscription. **GIMP** — free, open-source, community-developed
- This course teaches each concept once, then shows it concretely in **both tools**, with terminology mapped between them
- GIMP is **genuinely different software**, not a feature-for-feature clone of Photoshop

Raster Editing Fundamentals

Chapter 2 · The Raster Canvas

Chapter 1 defined a raster image as a fixed grid of pixels. This chapter covers the technical properties of that grid directly: how it relates to physical size, what color mode it stores its values in, and how much precision each of those values actually holds.

Resolution: PPI vs. DPI

PPI (pixels per inch) describes an image file's own resolution — how densely its pixel data is packed. **DPI** (dots per inch) is technically a printing term, describing how densely a printer lays down physical ink dots. The two are often used interchangeably in casual conversation, but they aren't strictly identical: PPI describes the file itself, DPI describes a printer's own output. Web images generally only care about absolute pixel dimensions; print work genuinely cares about both.

Image Dimensions vs. Print Size

A raster image's total pixel dimensions (say, 3000×2000) combined with its PPI setting together determine its actual physical print size. The exact same pixel dimensions can print small-and-sharp at a high PPI value, or large-and-blurry at a low PPI value — the pixel data itself hasn't changed, only how many inches it's being spread across. This relationship is a genuinely common source of confusion worth understanding directly rather than treating PPI as a mysterious quality dial.

Color Modes: RGB vs. CMYK

RGB is additive color — red, green, and blue light combining, used for anything viewed on a screen. **CMYK** is subtractive color — cyan, magenta, yellow, and black ink absorbing light, used for professional printing. Converting between the two isn't always lossless: some RGB colors, especially vibrant, saturated ones, simply can't be reproduced exactly within CMYK's own smaller color gamut. This is a real, practical reason to work in the correct mode from the very start of a print project, rather than converting only at the very end and discovering colors have shifted.

Bit Depth

Bit depth describes how many distinct tonal values each color channel can represent — **8-bit** per channel is the common default; **16-bit** per channel offers dramatically more tonal precision. This matters specifically during extensive editing or color grading: 8-bit images can show visible banding (abrupt tonal steps instead of a smooth gradient) after aggressive adjustments, where 16-bit tolerates the same adjustments far better. Both Photoshop and GIMP support both bit depths today, though GIMP's own full 16-bit (and higher) support was a more recent addition, arriving in GIMP 2.10.

SETTING	PHOTOSHOP	GIMP
Resolution (PPI)	Image > Image Size	Image > Print Size
Color mode	Image > Mode	Image > Mode
Bit depth	Image > Mode > 8/16/32 Bits	Image > Precision

WEB WORK SIDESTEPS THE PPI/DPI QUESTION ALMOST ENTIRELY

Browsers render images purely by pixel dimensions and ignore an image file's own PPI metadata entirely — a 1200×800 pixel image displays identically on the web whether its own metadata says 72 PPI or 300 PPI. Web work (this site's own CSS courses included) generally cares only about pixel dimensions, unlike print work, where PPI genuinely determines physical output size.

CHANGING THE PPI NUMBER DOESN'T ADD REAL DETAIL

It's a common misconception that raising an image's PPI value in software makes it print sharper by adding quality. Simply changing the PPI number in an image's own metadata — without resampling or adding real pixel data — only changes the print *size*, spreading the same fixed pixel data across a smaller physical area. This does make the print look sharper, but purely by shrinking how large it's printed, not by adding any real information. Genuinely upsampling an image to add detail requires actual resampling, which invents new pixel data through interpolation — and that process has real, hard quality limits, not a free source of extra detail.

Hands-On Exercises

EXERCISE 1

A photographer changes a 3000x2000 pixel image's PPI setting from 72 to 300 without resampling, and expects the image to now print with visibly more detail at the same physical size. Explain what actually happens, using this chapter's own material.

 [View solution](#)

EXERCISE 2

A designer creates a poster in RGB mode with several bright, saturated colors, then converts it to CMYK only at the very last step before sending it to a professional printer. Explain the risk in this workflow, using this chapter's own material.

 [View solution](#)

EXERCISE 3

Explain why a heavily color-graded image edited in 8-bit mode is more likely to show visible banding than the same edits performed in 16-bit mode.

 [View solution](#)

CHAPTER 2 QUICK REFERENCE

- **PPI** describes the image file; **DPI** describes a printer's own output — related but not identical
- Pixel dimensions + PPI together determine **print size** — the pixel data itself doesn't change when PPI is edited
- **RGB** (screens, additive) vs. **CMYK** (print, subtractive) — not always losslessly convertible; work in the right mode from the start for print
- **Bit depth** (8-bit vs. 16-bit) controls tonal precision — higher bit depth resists banding under heavy editing
- Raising PPI alone (without resampling) **does not add real detail** — it only changes print size

Raster Editing Fundamentals

Chapter 3 · Selections: The Foundation of Everything

Chapter 2 covered a raster canvas's own technical properties. This chapter is the first true editing concept — and arguably the single most important one in this entire course, since every other technique from here on assumes it.

The Core Idea: Nothing Happens Outside an Active Selection

Any edit tool — paint, adjustment, filter — only affects pixels within the current selection. With no selection active at all, the entire canvas is implicitly selected, which is why beginners often don't notice this rule exists until it produces confusing behavior. Understanding this one rule alone explains a huge share of "why isn't this tool doing anything" confusion in raster editing.

Basic Selection Tools

Marquee (Photoshop) / **Rectangle** and **Ellipse Select** (GIMP) — geometric shapes. **Lasso** (Photoshop) / **Free Select** (GIMP) — freehand drawing. **Magic Wand** (Photoshop) / **Select by Color** and **Fuzzy Select** (GIMP) — color/tone-based selection, controlled by a **tolerance** setting that determines how similar adjacent pixels need to be before they're included in the selection.

Combining Selections

Holding **Shift** adds to an existing selection; **Alt/Option** subtracts from it; **Shift+Alt** intersects the new selection with the existing one. This modifier-key logic is genuinely identical across both tools — a rare case of near-total UI parity between Photoshop and GIMP. One more related operation worth knowing: **Select > Inverse** (Photoshop) / **Select > Invert** (GIMP) flips the current selection to its exact opposite in one step — genuinely useful when it's easier to select the one thing you don't want, then invert, than to select everything you do want directly.

Feathering & Refining Edges

Feathering softens a selection's own hard edge, producing a smoother blend wherever that selection is subsequently used. This is only a first mention — Advanced Compositing & Retouching's own Chapter 2 goes much deeper into precise edge refinement using channels and Photoshop's Select and Mask workspace.

Saving & Loading Selections

A selection doesn't have to be lost the moment it's deselected — it can be saved as a reusable alpha channel, then reloaded later in the same project. Genuinely useful when a complex, time-consuming selection might need to be reused again further along in the same editing session.

TOOL	PHOTOSHOP NAME	GIMP NAME
Geometric selection	Marquee	Rectangle / Ellipse Select
Freehand selection	Lasso	Free Select
Color-based selection	Magic Wand	Select by Color / Fuzzy Select

SELECTIONS AND MASKS: TWO SIDES OF THE SAME IDEA

A carefully built, precise selection is exactly what becomes the basis of a layer mask — the next chapter's own topic. Selections and masks aren't two unrelated concepts; a selection is essentially a mask's own starting point, made permanent and reusable.

"THE TOOL ISN'T WORKING" IS OFTEN JUST A LEFTOVER SELECTION

A genuinely common beginner mistake: an edit "isn't working" or seems to only affect part of the image, when in reality a selection made several steps earlier is still active, silently restricting where the current edit can apply — a direct, confusing consequence of this chapter's own core rule. Before assuming a tool is broken, checking **Select > Deselect** (or GIMP's equivalent) is almost always the first thing worth trying.

Hands-On Exercises

EXERCISE 1

You need to select a sky in a photo that's mostly one continuous shade of blue, but a few clouds break it up. Which selection tool from this chapter would you reach for first, and what setting would you adjust if the initial selection missed some slightly different-toned patches of sky?

 [View solution](#)

EXERCISE 2

A user paints with a brush tool and notices color is only appearing in a small rectangular area in the middle of an otherwise much larger canvas. Using this chapter's own material, explain the most likely cause and the fix.

 [View solution](#)

EXERCISE 3

You need to select everything in a photo EXCEPT a person's face, which you've already carefully selected using the Lasso/Free Select tool. Explain which modifier key combination and selection approach would let you achieve this efficiently, without redrawing the entire selection from scratch.

 [View solution](#)

CHAPTER 3 QUICK REFERENCE

- **The core rule:** edits only affect pixels inside the active selection; no selection means the whole canvas is implicitly selected
- **Marquee/Lasso/Magic Wand** (Photoshop) map to **Rectangle-Ellipse/Free Select/Select by Color-Fuzzy Select** (GIMP)
- **Shift** adds, **Alt/Option** subtracts, **Shift+Alt** intersects — identical in both tools
- Selections can be **saved as alpha channels** and reloaded later, rather than lost on deselect
- A stuck-open, forgotten selection is one of the most common causes of "this tool isn't working"

Raster Editing Fundamentals

Chapter 4 · Layers: Non-Destructive Stacking

Chapter 3 covered how to restrict where an edit applies. This chapter covers a genuinely different foundational concept — keeping different pieces of content on separate, independently editable layers, rather than merging everything into one permanent, flattened image.

The Core Idea: Layers Are Independent, Stackable Sheets

A layer works like a transparent sheet stacked on top of others — each one holds its own content, editable, movable, or deletable entirely independently, without affecting any layer above or below it. This is the direct opposite of a flattened, single-layer image, where everything has already been permanently merged into one and can no longer be separated.

Layer Order & Stacking

Layers stack top-to-bottom in a panel or dialog, and whatever is higher in that stack visually covers whatever sits below it — exactly like physical transparent sheets laid on top of each other.

Reordering layers directly changes what's visible in the final image.

Opacity

A layer's own **opacity** setting controls how much of whatever's beneath it shows through — 100% is fully opaque, 0% is fully invisible. This is distinct from a layer **mask**'s own selective transparency (Chapter 5): opacity affects the entire layer uniformly, while a mask can make different parts of the same layer transparent by different amounts.

Blend Modes

A layer can combine with whatever's beneath it using math well beyond simple opacity. Three of the most commonly used starting points: **Multiply** (darkens), **Screen** (lightens), and **Overlay** (increases contrast). This blend-mode math is genuinely identical in both Photoshop and GIMP — these are standard, well-defined compositing operations, not something proprietary to either tool.

Layer Groups/Folders

Many layers can be organized into a collapsible group — Photoshop's own **Groups**, GIMP's own **Layer Groups** — genuinely useful once a project accumulates dozens of individual layers and the panel itself starts becoming hard to navigate.

CONCEPT	PHOTOSHOP	GIMP
Layer panel	Layers panel	Layers dialog
Opacity control	Opacity slider (Layers panel)	Opacity slider (Layers dialog)
Blend modes	Blend mode dropdown	Mode dropdown
Organizing many layers	Groups	Layer Groups

WHERE THIS LEADS DIRECTLY NEXT

This chapter's own opacity setting affects a layer uniformly, everywhere at once. Chapter 5's own layer masks build directly on that same idea, but let visibility vary by area instead — hiding part of a layer while keeping the rest fully visible, all without touching the layer's own uniform opacity setting at all.

FLATTENING EARLY TO SAVE FILE SIZE IS A COMMON, REGRETTABLE HABIT

Flattening merges every layer into one, permanently discarding the ability to adjust any individual layer afterward. A genuinely common, frustrating scenario: a client or stakeholder requests "just one more small change" after the file was already flattened to save disk space, with no way left to isolate and adjust the one element that needs changing. Standard practice is keeping every layer intact throughout the entire editing process, flattening — or better, exporting a separate flattened *copy* — only at the very final step, once no further changes are expected.

Hands-On Exercises

EXERCISE 1

A project has a background layer, a text layer, and a logo layer stacked in that order (background at the bottom). Explain what would happen visually if the text layer were moved below the background layer in the stack.

 [View solution](#)

EXERCISE 2

Explain the practical difference between setting a layer's own opacity to 50% versus applying a layer mask that makes exactly half of that same layer fully transparent and the other half fully visible.

 [View solution](#)

EXERCISE 3

A designer flattens a multi-layer poster design immediately after finishing a first draft "to keep the file small," then a client asks for the logo to be repositioned two days later. Using this chapter's own warning box, explain the problem this creates and what the designer should have done instead.

 [View solution](#)

CHAPTER 4 QUICK REFERENCE

- Layers are **independent, stackable sheets** — editable separately, unlike a flattened image
- Stack order determines visibility — **higher layers cover lower ones**
- **Opacity** affects a whole layer uniformly; a **mask** (next chapter) varies by area instead
- **Blend modes** (Multiply, Screen, Overlay) use identical, standard compositing math in both tools
- Keep layers intact throughout editing — **flatten only at the very final step**, ideally as a separate exported copy

Raster Editing Fundamentals

Chapter 5 · Layer Masks: Non-Destructive Hiding, Not Deleting

Chapter 4 closed by promising that masks let visibility vary by area rather than uniformly across a whole layer. This chapter is that concept in full — arguably the single most important non-destructive editing technique this course covers.

The Core Idea: Black Conceals, White Reveals

A layer mask is a grayscale channel attached to a layer. Painting **black** on the mask hides that area of the layer; painting **white** reveals it; **gray** produces partial transparency somewhere in between. This simple rule works identically in both Photoshop and GIMP, and is worth memorizing exactly as stated — black conceals, white reveals.

Masking vs. Erasing: The Critical Distinction

The Eraser tool permanently deletes pixel data from a layer — genuinely destructive, with no way back except undo history or starting over from an earlier saved version. A layer mask instead *hides* pixels without touching the underlying pixel data at all. Repainting white on the mask instantly restores full visibility, since the original pixels were never actually removed in the first place — only ever hidden.

Painting on a Mask

The Brush tool must be applied directly to the mask's own thumbnail — not the layer's own image thumbnail — for it to actually paint on the mask rather than the image itself. A genuinely common beginner mistake: accidentally painting on the image directly because the wrong thumbnail was selected. Both Photoshop and GIMP show a highlighted border around whichever thumbnail is currently active, worth checking before painting anything.

Creating a Mask From a Selection

Chapter 3's own selection tools become directly useful here: when a mask is added while a selection is active, that selection automatically becomes the mask's own black-and-white pattern — the selected area becomes visible (white), everything outside it becomes hidden (black). This is the natural bridge between Chapter 3's own selection work and this chapter's own masking concept.

Layer Masks in GIMP vs. Photoshop

Both tools support masking identically in concept, with one minor workflow difference: Photoshop's own mask thumbnail appears automatically next to a layer's own thumbnail the moment a mask is added. GIMP requires explicitly adding a Layer Mask via its own menu command first, before a mask thumbnail appears at all — a workflow difference, not a conceptual one.

ACTION	PHOTOSHOP	GIMP
Adding a mask	Layer > Layer Mask	Layer > Mask > Add Layer Mask
Painting the mask	Brush on mask thumbnail	Brush on mask thumbnail
Mask rule	Black hides, white reveals	Black hides, white reveals

THE FOUNDATION FOR EVERYTHING IMAGEEDIT2 BUILDS ON

This chapter's own black/white/gray rule is exactly what Advanced Compositing & Retouching's own Chapter 2 (Advanced Masking: Channels & Refine Edge) builds on directly — channel-based selection for fine detail like hair ultimately produces the same black/white mask pattern this chapter introduces, just generated through a more sophisticated process.

DELETING A MASK ISN'T THE SAME AS TEMPORARILY DISABLING IT

It's easy to assume deleting a layer mask is basically the same as just turning it off for a moment. Deleting a mask permanently discards it (usually with a prompt asking whether to apply its current effect or discard it first). **Disabling** a mask instead — shift-clicking its thumbnail, or the equivalent in either tool — temporarily ignores it entirely without losing any of the actual mask data underneath. This is a genuinely useful, fully non-destructive way to compare "with mask" against "without mask" side by side, without committing to deleting anything at all.

Hands-On Exercises

EXERCISE 1

You've carefully selected a person in a photo using the Lasso/Free Select tool from Chapter 3, and want everything outside that selection to become hidden on this layer. Explain what happens when a mask is added while that selection is still active.

 [View solution](#)

EXERCISE 2

A user paints black with a large brush, intending to hide part of a layer via its mask, but instead permanently damages the actual image content. Using this chapter's own material, explain the most likely mistake.

 [View solution](#)

EXERCISE 3

You want to quickly compare how an image looks with a particular mask applied versus without it, without risking losing the mask's own careful work if you decide to keep it. Using this chapter's own warning box, explain the correct way to do this.

 [View solution](#)

CHAPTER 5 QUICK REFERENCE

- The core rule: **black conceals, white reveals**, gray is partial — identical in both tools
- A mask **hides** pixels non-destructively; the Eraser **permanently deletes** them — a critical distinction
- Always confirm the **mask thumbnail** (not the image thumbnail) is active before painting
- An active selection, when a mask is added, becomes the mask's own **black/white pattern** automatically
- **Disabling** a mask (temporary) is not the same as **deleting** it (permanent) — disabling lets you compare freely with no risk

Raster Editing Fundamentals

Chapter 6 · Adjustment Layers & Color Correction

Chapter 5's own masking concept applied non-destructive editing to visibility. This chapter applies that same philosophy to color correction — and is the first place this course flags a genuine, real difference between Photoshop and GIMP rather than a purely cosmetic one.

Direct Adjustment vs. Adjustment Layer

A direct adjustment (Photoshop's own `Image > Adjustments` menu) permanently alters the pixel data of whichever layer is currently active — destructive, in the same category as the Eraser from Chapter 5. An **Adjustment Layer** instead sits above a layer in the stack and affects everything beneath it without altering any actual pixel data at all — fully editable, removable, or maskable later, using Chapter 5's own masking concept directly on the adjustment itself.

Levels

The **histogram** graphs how many pixels exist at each brightness value across the image. Levels provides black point, white point, and midtone sliders, remapping the darkest, brightest, and middle values respectively — a fundamental first tool for correcting a flat, low-contrast image.

Curves

Curves is a more flexible, generalized version of Levels — instead of three fixed sliders, it plots an arbitrary curve mapping input brightness directly to output brightness. Genuinely more powerful and precise, Curves is the standard tool professional retouchers reach for once Levels' own simpler three-point control isn't precise enough.

Hue/Saturation

Hue/Saturation adjusts a color's own hue (the color itself), saturation (intensity), and lightness independently — and can target a specific color range (only reds, for instance) rather than affecting the entire image uniformly.

Adjustment Layers in Photoshop vs. GIMP

In Photoshop, Adjustment Layers are a first-class, fully non-destructive citizen right in the Layers panel, automatically arriving with their own mask attached. GIMP historically applied color adjustments directly and destructively at the layer level, via its own GEGL-based operations — GIMP 2.10 introduced genuine non-destructive filter and adjustment capability, but it remains real, worth flagging honestly, that GIMP's own non-destructive color-correction workflow is less mature and less automatic than Photoshop's, exactly the kind of honest difference Chapter 1's own warning box promised this course wouldn't smooth over.

TOOL	PHOTOSHOP	GIMP
Adjustment mechanism	Adjustment Layers (non-destructive by default)	GEGL-based operations (non-destructive support newer, less automatic)
Levels/Curves	Available as an Adjustment Layer	Available, destructive unless deliberately using non-destructive filters
Automatic masking	Yes — comes with the adjustment by default	Not automatic — must be added deliberately

THE SAME BLACK/WHITE RULE, NOW ON A COLOR CORRECTION

An Adjustment Layer's own mask works using exactly the black/white rule already learned in Chapter 5 — painting black on that mask excludes part of the image from the color correction entirely, letting a single adjustment brighten only a subject's face, for instance, without affecting the rest of the photo at all.

GIMP BEING CAPABLE DOESN'T MEAN ITS ADJUSTMENTS ARE NON-DESTRUCTIVE BY DEFAULT

It's tempting to assume that since GIMP is a genuinely capable, real image editor, its own color-adjustment workflow must be just as non-destructive as Photoshop's out of the box. This is a real, concrete difference worth being deliberate about: a GIMP user wanting genuinely non-destructive color correction needs to actively choose duplicate layers or newer GEGL-based non-destructive filters specifically, rather than assuming every adjustment automatically behaves non-destructively the way Adjustment Layers do in Photoshop by default.

Hands-On Exercises

EXERCISE 1

A photo is generally well-lit, but the subject's face is slightly darker than the rest of the frame. Using this chapter's own material, explain how an Adjustment Layer combined with a mask would let you brighten only the face without affecting the rest of the photo.

 [View solution](#)

EXERCISE 2

A GIMP user applies a color adjustment directly to a layer, assuming it will behave non-destructively the same way Photoshop's Adjustment Layers do, then later needs to undo just that one specific adjustment after several other edits. Using this chapter's own warning box, explain the problem they're likely to run into.

 [View solution](#)

EXERCISE 3

Explain the practical difference between using Levels and using Curves to correct a photo's own contrast, and why a professional retoucher might prefer Curves for a particularly tricky correction.

 [View solution](#)

CHAPTER 6 QUICK REFERENCE

- **Direct adjustments** alter pixel data permanently; **Adjustment Layers** sit above a layer non-destructively, fully editable later
- **Levels** — black/white/midtone sliders; **Curves** — a fully flexible input-to-output mapping, more powerful and precise
- **Hue/Saturation** can target a specific color range, not just the whole image
- An Adjustment Layer's own mask uses the **same black/white rule** from Chapter 5
- **GIMP's own non-destructive adjustment workflow is genuinely less mature** than Photoshop's — a real difference, not smoothed over

Raster Editing Fundamentals

Chapter 7 · The Clone Stamp & Healing Tools

Chapter 6 covered adjusting an image's overall tone and color. This chapter is the first hands-on retouching technique — actually removing unwanted elements from a photo, rather than adjusting how the whole image looks.

The Clone Stamp: Copying Pixels Directly

Sampling a source point (**Alt** / **Option** +click) and then painting that exact sampled content elsewhere is the Clone Stamp's entire mechanism — a direct, literal pixel copy, with no blending or analysis involved. It's genuinely simple mechanically, but requires real skill to use convincingly: it copies *exactly*, texture, lighting, and shadow included, which can create an obviously repeated pattern if the same source area is reused carelessly nearby.

The Healing Brush / Spot Healing

The Healing Brush is a smarter tool: it also samples from a source point, but automatically blends the sampled texture with the surrounding area's own tone and lighting, rather than a literal 1:1 copy. This makes it far better suited to blemish removal on skin, where lighting varies subtly across the surface. **Spot Healing** goes one step further — it doesn't even require manually choosing a source point at all, analyzing surrounding pixels automatically instead.

Photoshop vs. GIMP Naming

Clone Stamp (Photoshop) / **Clone Tool** (GIMP). Healing Brush (Photoshop) / **Heal Tool** (GIMP). Spot Healing Brush has no perfectly identical, one-click equivalent in GIMP historically — GIMP's own Heal Tool covers similar ground but still requires manually choosing a source point, unlike Photoshop's own fully automatic Spot Healing Brush. Another honest, concrete difference worth flagging directly, per Chapter 1's own commitment to naming real gaps rather than smoothing them over.

Working on a Duplicate Layer for Safety

Retouching directly on the original photo layer is genuinely destructive if done carelessly. A standard professional habit: duplicate the layer first (or use a new empty layer with "Sample: All Layers" enabled), retouch on that duplicate or empty layer instead, and keep the original completely untouched underneath as a safety net — a direct application of Chapter 4's own non-destructive layering philosophy to retouching specifically.

TOOL	PHOTOSHOP	GIMP	KEY BEHAVIOR
Direct pixel copy	Clone Stamp	Clone Tool	Exact 1:1 copy, no blending
Blended healing	Healing Brush	Heal Tool	Blends with surrounding tone/lighting
Automatic source	Spot Healing Brush	No exact equivalent	Analyzes surroundings automatically

A MORE ADVANCED EVOLUTION OF THIS SAME IDEA

Advanced Compositing & Retouching's own Chapter 5 (Frequency Separation) is a genuinely more advanced, professional evolution of the basic retouching this chapter introduces — rather than copying or blending pixels directly, it separates an image's own texture from its tone into independent layers, editable separately.

CLONE STAMP AND HEALING BRUSH AREN'T SIMPLY "BETTER VS. WORSE"

It's tempting to treat the Healing Brush as the strictly upgraded version of the Clone Stamp and always reach for it instead. The Clone Stamp's exact 1:1 copying is actually the **right** choice specifically when a hard edge or precise detail needs preserving — cloning a straight architectural line, for instance — where the Healing Brush's own automatic blending would incorrectly soften or blur that edge. Picking the wrong one for the situation produces a visibly wrong result; these are two tools built for two different jobs, not a better and a worse version of the same one.

Hands-On Exercises

EXERCISE 1

You need to remove a small blemish from a person's cheek, where the lighting varies subtly across the skin. Which tool from this chapter would you reach for, and why would the alternative be a worse choice here?

 [View solution](#)

EXERCISE 2

You need to extend a straight architectural roofline in a photo by cloning part of the existing roof further along. Explain why the Healing Brush would likely produce a visibly wrong result here, and why the Clone Stamp is the better choice.

 [View solution](#)

EXERCISE 3

Explain why retouching directly on a photo's own original layer, rather than a duplicate, is a real risk — and what standard habit avoids it, connecting back to an earlier chapter's own material.

 [View solution](#)

CHAPTER 7 QUICK REFERENCE

- **Clone Stamp** — exact 1:1 pixel copy, no blending; best for preserving hard edges and precise detail
- **Healing Brush** — samples, then blends with surrounding tone/lighting; best for skin and subtly varying surfaces
- **Spot Healing Brush** analyzes surroundings automatically — GIMP has no perfectly identical one-click equivalent
- Always retouch on a **duplicate or empty layer**, keeping the original untouched as a safety net
- These two tools solve **different problems** — not a better and a worse version of the same tool

Raster Editing Fundamentals

Chapter 8 · Basic Filters & Effects

Chapter 7 covered retouching by copying and blending pixels directly. This chapter is a deliberately basic first look at filters — Advanced Compositing & Retouching's own Chapter 8 goes considerably deeper into filter galleries and stacking multiple filters together.

What a Filter Actually Does

A filter applies a mathematical transformation to pixel data. Unlike an Adjustment Layer (Chapter 6), most basic filters directly alter pixel data destructively, unless applied to a duplicate layer or — in Photoshop specifically — converted to a Smart Filter first, a capability Advanced Compositing & Retouching's own Chapter 4 covers fully.

Gaussian Blur

The most common blur filter, **Gaussian Blur** averages each pixel with its neighbors within a chosen radius, producing a smooth, soft blur. It's used for faking a background's own depth-of-field blur, general noise reduction, or as a building block for more advanced techniques like Frequency Separation (Advanced Compositing & Retouching's own Chapter 5).

Sharpen / Unsharp Mask

Unsharp Mask is the standard sharpening tool in both Photoshop and GIMP — a counterintuitive name, historically borrowed from a darkroom photography technique. It increases apparent sharpness by boosting contrast specifically at edges, detected by comparing the image against a deliberately blurred — "unsharp" — version of itself used as a reference mask. Worth explaining the odd name directly, since it genuinely confuses many beginners who assume "unsharp" means the opposite of what the tool actually does.

Applying a Filter Non-Destructively

Working on a duplicate layer — Chapter 7's own established habit — applies here again before running any filter. Photoshop's own **Smart Filters** (introduced briefly here, covered fully in Advanced Compositing & Retouching's own Chapter 4) go further, letting a filter be re-edited or removed entirely after the fact, without even needing a duplicate layer as a safety net.

FILTER	PHOTOSHOP	GIMP
Blur	Filter > Blur > Gaussian Blur	Filters > Blur > Gaussian Blur
Sharpen	Filter > Sharpen > Unsharp Mask	Filters > Enhance > Sharpen (Unsharp Mask)

DELIBERATELY JUST THE BASICS HERE

This chapter covers only Blur and Sharpen as a foundation. Advanced Compositing & Retouching's own Chapter 8 (Advanced Filters & Effect Stacking) covers filter galleries, combining multiple filters together, and GIMP's own GEGL-based filter system compared against Photoshop's filter gallery — genuinely more ground than belongs in a fundamentals-level chapter.

SHARPENING CAN'T FIX A GENUINELY OUT-OF-FOCUS PHOTO

It's a common misconception that sharpening can rescue a blurry or out-of-focus photo. Unsharp Mask only boosts contrast at edges that already exist in the image data — it cannot invent detail that was never actually captured. A severely out-of-focus photo has no real edge information left for Unsharp Mask to work with at all, so sharpening it aggressively just amplifies noise and produces a harsh, artificial-looking result, rather than genuinely restoring the focus that was never there to begin with.

Hands-On Exercises

EXERCISE 1

You want to fake a shallow depth-of-field look, keeping a subject sharp while blurring the background behind them. Using this chapter's own material and Chapter 3's own selections, describe the general approach.

 [View solution](#)

EXERCISE 2

A colleague is confused that a tool called "Unsharp Mask" is actually used to sharpen an image, not blur it. Using this chapter's own material, explain where the name actually comes from.

 [View solution](#)

EXERCISE 3

Someone applies aggressive Unsharp Mask settings to a photo that was genuinely out of focus when it was taken, hoping to rescue it, and ends up with a harsh, noisy-looking image. Using this chapter's own warning box, explain why this happened.

 [View solution](#)

CHAPTER 8 QUICK REFERENCE

- Most basic filters alter pixel data **destructively** unless applied to a duplicate layer or a Smart Filter
- **Gaussian Blur** averages pixels within a radius — used for fake depth of field, noise reduction, and as a building block elsewhere
- **Unsharp Mask** boosts contrast at existing edges, using a deliberately blurred reference — an odd but historically accurate name
- This chapter is deliberately basic — **Advanced Compositing & Retouching's own Chapter 8** goes much deeper
- Sharpening **cannot invent detail** that was never captured — it only boosts contrast at edges that already exist

Raster Editing Fundamentals

Chapter 9 · Exporting & File Formats

Every prior chapter built up a finished, fully-layered editing project. This final technical chapter before the capstone covers getting that project out of the tool correctly — without losing the ability to keep editing it later.

Native Formats: PSD vs. XCF

Photoshop's own **PSD** and GIMP's own **XCF** both preserve every layer, mask, adjustment layer, and editing state fully — the "working file" format, meant for continued editing, not final delivery. A genuine interoperability gap is worth naming honestly here: opening a PSD in GIMP, or an XCF in Photoshop, works with varying and often imperfect fidelity, since neither company builds guaranteed, fully faithful native support for the other's own format.

Export vs. Save

Save/Save As keeps the native, fully-editable format intact. **Export** instead creates a separate, different-format file specifically for delivery or use elsewhere — Photoshop's own "Export As," GIMP's own "Export As." Conceptually identical in both tools: the native working file stays untouched, and a new delivery file is generated alongside it.

Web-Ready Formats: PNG vs. JPG

PNG is lossless, supports transparency, and produces a larger file size — best for graphics, logos, or anything needing a transparent background. **JPG** uses lossy compression, produces a smaller file size, and has no transparency support at all — best for photographs, where some quality loss is genuinely imperceptible at reasonable settings. A genuinely practical, common real-world decision point, not an academic distinction.

JPG Quality & Compression Artifacts

JPG's own lossy compression is adjustable — higher quality settings produce a larger file with less visible artifacting. Aggressive compression produces visible blocky artifacts, especially around hard edges and text, since JPG's own compression algorithm was originally designed around smooth photographic content, not sharp, hard-edged graphics. A real, common mistake: over-compressing a JPG containing text or sharp graphic elements, producing visibly ugly blocky artifacting around exactly those edges.

Flattening on Export

Revisiting Chapter 4's own warning box directly: exporting to PNG or JPG requires flattening, since neither format has any concept of separate layers at all. This is exactly why exporting a separate flattened *copy* — rather than flattening the actual working file — is the right practice, letting the export happen freely without ever endangering the original layered file.

FORMAT	LAYERS PRESERVED?	LOSSY/LOSSLESS	TRANSPARENCY	BEST FOR
PSD/XCF	Yes	Lossless	Yes	The working file itself
PNG	No	Lossless	Yes	Graphics, logos, transparency
JPG	No	Lossy	No	Photographs

A GENUINELY COMMON WEB DEVELOPMENT DECISION

Choosing between PNG and JPG for a website's own images is a real, everyday decision web developers make constantly — this site's own CSS courses assume images are already correctly exported by the time they reach a stylesheet, but this chapter's own material is exactly what determines that choice in the first place.

"JUST ALWAYS USE PNG SINCE IT'S LOSSLESS" IS THE WRONG DEFAULT

It's tempting to assume PNG is simply the better choice across the board, since it's lossless and technically higher quality. PNG's own lossless nature means file sizes for photographic content can be dramatically larger than an equivalent JPG at a perfectly acceptable quality setting — a real, practical problem for web performance, page load times, and bandwidth. The correct choice depends on the content type (photo vs. graphic) and whether transparency is actually needed — not a blanket assumption that lossless always wins.

Hands-On Exercises

EXERCISE 1

A website needs a photograph of a landscape and a logo with a transparent background. Using this chapter's own material, recommend a file format for each, and justify both choices.

 [View solution](#)

EXERCISE 2

A designer exports a graphic containing sharp text over a solid background as a heavily-compressed, low-quality JPG to save file size, and the text comes out looking blocky and ugly. Explain what went wrong, using this chapter's own material.

 [View solution](#)

EXERCISE 3

A photographer's finished project has 40 layers, masks, and adjustment layers. Explain the correct way to produce a JPG for a client's website while keeping the original project fully editable for future revisions.

 [View solution](#)

CHAPTER 9 QUICK REFERENCE

- **PSD/XCF** — the native working format, preserving every layer; imperfect fidelity when opened cross-tool
- **Save** keeps the native file; **Export** creates a separate delivery file, leaving the working file untouched
- **PNG** — lossless, supports transparency, larger files, best for graphics. **JPG** — lossy, smaller files, no transparency, best for photos
- Aggressive JPG compression produces **visible artifacts around hard edges/text** specifically
- Export a **flattened copy**, never the original working file — the same safety habit from Chapter 4
- "Always use PNG" is wrong — the right format depends on **content type and transparency needs**, not a blanket lossless preference

Raster Editing Fundamentals

Chapter 10 · Capstone: Editing a Real Photo End to End in Both Tools

Every prior chapter introduced one concept at a time. This capstone edits a single real portrait photo end to end, applying each of those concepts in sequence, with both Photoshop's own workflow and GIMP's own workflow shown side by side at every step — closing the course the way it began, treating both tools as equally real implementations of one shared discipline.

The Scenario

A portrait photo needs four things before delivery: a slight color cast corrected, one stray blemish removed, subtle background blur added for a more flattering sense of depth, and final export for both a client web preview and the original working file preserved for any future revision.

Step 1 — Setting Up the Canvas (Chapter 2)

Before touching anything, confirm the working file's own resolution and color mode. Since delivery includes both a web preview and a possible future print use, work in RGB at a high enough resolution and bit depth to comfortably cover both — a dedicated CMYK conversion, if a print file is ever specifically needed, happens as its own separate step later, not now.

Step 2 — Selecting the Subject (Chapter 3)

Use the Lasso/Free Select tool to rough out a selection around the subject, separating them from the background — the foundation every later step in this capstone builds on.

Step 3 — Duplicating the Layer for Safety (Chapters 4 & 7)

Duplicate the original photo layer before making any further changes, keeping the untouched original safely underneath — the same non-destructive habit established across this entire course.

Step 4 — Masking the Background Blur (Chapter 5)

Duplicate the background specifically, apply Gaussian Blur (Chapter 8) to that duplicate, then invert the subject selection from Step 2 and use it to build a mask — hiding the blur wherever the subject is, revealing it everywhere else, softening the background without touching the subject at all.

Step 5 — Correcting Color Non-Destructively (Chapter 6)

Add a Curves Adjustment Layer above everything to correct the slight color cast, keeping the correction fully editable and removable afterward — never applied as a destructive direct adjustment.

Step 6 — Removing a Blemish (Chapter 7)

On the duplicated subject layer from Step 3, use the Healing Brush/Heal Tool to remove the stray blemish, letting its automatic blending match the subtly varying skin tone around it.

Step 7 — A Light Touch of Sharpening (Chapter 8)

Apply Unsharp Mask carefully and modestly — this photo is already reasonably in focus, so per Chapter 8's own warning box, sharpening here only needs to enhance existing detail slightly, not attempt to rescue anything.

Step 8 — Exporting for Both Delivery Needs (Chapter 9)

Save the fully layered PSD/XCF working file first, keeping every layer, mask, and adjustment intact. Then Export a separate flattened JPG for the client's web preview — the working file itself is never flattened or overwritten.

CAPSTONE STEP	CHAPTER IT DRAWS FROM
Step 1 — Canvas setup	Chapter 2
Step 2 — Subject selection	Chapter 3
Step 3 — Duplicate for safety	Chapters 4 & 7
Step 4 — Masked background blur	Chapter 5 (+ Chapter 8's Blur)
Step 5 — Non-destructive color correction	Chapter 6
Step 6 — Blemish removal	Chapter 7
Step 7 — Light sharpening	Chapter 8
Step 8 — Export for delivery	Chapter 9

WHERE THIS SCALES UP NEXT

This capstone worked one photo, end to end, by hand. Advanced Compositing & Retouching's own capstone scales this exact same discipline up to a real production batch of many photos at once, automated via scripting — the natural next step once editing a single image confidently is second nature.

THIS CAPSTONE DELIBERATELY STAYS AT A FUNDAMENTALS LEVEL

Genuinely advanced techniques — channel-based hair masking, frequency separation, Smart Objects — are intentionally not attempted here. This capstone exercises exactly the fundamentals covered across Chapters 2 through 9; those more advanced techniques are Advanced Compositing & Retouching's own job, building on this exact foundation rather than repeating it.

Hands-On Exercises

EXERCISE 1

In this capstone's own Step 4, why is the subject selection from Step 2 inverted before being used to mask the blurred background layer, rather than used directly?

 [View solution](#)

EXERCISE 2

A client later asks for the color correction from Step 5 to be adjusted slightly. Explain why this request is easy to fulfill given how this capstone's own workflow was structured, referencing the specific chapter that made this possible.

 [View solution](#)

EXERCISE 3

Explain why this capstone's own shape — one photo, worked through step by step in both tools side by side — is a genuinely different capstone format than Web Servers Fundamentals' own multi-scenario capstone or Nginx In Depth's own single-config build, and why it fits this specific course.

 [View solution](#)

COURSE COMPLETE

Raster Editing Fundamentals — 10 of 10 chapters complete. Advanced Compositing & Retouching remains outstanding as this two-course track's own next planned course.

CHAPTER 10 QUICK REFERENCE

- One real photo, worked **end to end**, with both Photoshop's and GIMP's own workflows shown side by side at each step
- Every step draws directly from a specific earlier chapter — **nothing new is introduced** in this capstone itself
- Safety habits (duplicate layers, non-destructive adjustments, exporting a copy) are applied throughout, not just mentioned once
- This capstone stays deliberately at a **fundamentals level** — advanced techniques belong to Advanced Compositing & Retouching