

Steal Any Website With One Prompt

You asked for SCRAPE. Here's the exact prompt — plus a few more slightly-illegal things you can do with Claude.

■ The problem

You see a website you love — a competitor's landing page, a slick portfolio, a pricing section that just works — and you want that for yourself. So you sit there trying to eyeball it and rebuild it from scratch, guessing at the spacing, the colors, the layout. It takes forever and it comes out almost-right.

There's a faster way. You let Claude pull the whole site down into your project, break down exactly how it's built, and rebuild a clean version you own. One prompt.

■ What it feels like

You paste a URL. A minute later there's a folder on your machine with the page, its styles, and its assets — and Claude hands you a plain breakdown of the layout, the colors, and the fonts. Then it scaffolds your own version in your stack. You went from "I want something like that" to "I have something like that, in my code" without opening the inspector once.

■ The prompt (copy this)

Drop this into Claude Code (or any agent that can run shell commands), inside the project folder you want to build in:

You're working inside this project folder. I want to study and rebuild this website: <PASTE URL>

Do this, step by step:

1. Make a folder called `scraped/`.
2. Download the page into it with a browser user-agent:
`curl -L -A "Mozilla/5.0" "<PASTE URL>" -o scraped/index.html`
3. Open scraped/index.html, find every linked stylesheet, script, and image,
and download each one into scraped/ (keep the folder structure).
4. Give me a short breakdown of the page: the sections top to bottom, the
color palette (as hex codes), the fonts, and the main components.
5. Then scaffold a clean rebuild of it in THIS project using <YOUR STACK,
e.g. Next.js + Tailwind> – match the layout and style, but write fresh,
readable code that I own. Don't copy their code; rebuild it.

Don't skip the CSS or the assets – I want enough to actually rebuild it.

Swap in the URL and your stack. That's the whole thing.

■ How to use it

- **Best in Claude Code** (or Cursor / any agent with terminal access) — it needs to actually run `curl` and write files. In a plain chat it can't

download anything.

- Point it at an **empty or scratch project folder** the first time so it doesn't touch your real code.
- If a site is heavy on JavaScript (the content loads after the page does), tell Claude to "render it first" — it can use a headless browser (Playwright) to grab the finished HTML instead of the empty shell.
- **Rebuild, don't rip.** The move is to learn the structure and make your own clean version — not to publish someone else's code. That's what keeps it useful and yours.

■ 3 more prompts worth stealing

Screenshot → code. Paste a screenshot of any page or component and say:

```
Rebuild this screenshot as a responsive component in <your
stack>. Match the
layout, spacing, and colors as closely as you can. Ask me before
guessing on
anything you can't see.
```

Competitor teardown. Point Claude at a competitor's page:

```
Read this page: <URL>. Break down its structure, its offer, and
the exact copy
it uses section by section. Then tell me 5 things I could do
better on mine.
```

Design-system lift. From a scraped or pasted site:

```
Pull the design system out of this: the color palette (hex), the  
type scale,  
the spacing, the button and card styles. Give it to me as  
Tailwind config +  
a few reusable components I can drop into my project.
```

■ The bigger picture

This is one move in a bigger habit: stop building from a blank page. Anything you can see, Claude can pull apart and rebuild — a site, a component, a design system, a competitor's whole funnel. The people shipping fast aren't more talented, they just start from something real instead of a blank file. Scraping-to-rebuild is how you do that on demand.

■ Your next step

Grab a site you've been meaning to copy the *feel* of, open a scratch folder, and run the prompt. Ten minutes and you'll have its bones in your own code. Then keep the three bonus prompts somewhere handy — they come up more than you'd think.

Built by Axionox. I build AI systems and automations for a living. Follow for more like this.