

A website is content and design living on a host, reached through a domain, and improved one published version at a time.

01 CONTENT

The rooms and furniture.

Your words, photos, offers, proof, services, and contact information.

02 DESIGN

The architecture and flow.

Color, type, spacing, sections, and what visitors notice first.

03 CTA / FORM

The doorbell or front desk.

The next action: email, call, book, buy, inquire, or submit.

04 PUBLISH / DEPLOY

Turn on the lights and open.

A saved version becomes available at a real visitor URL.

05 DOMAIN

The street address.

A memorable name such as yourbusiness.com. You register or own it.

06 DNS

The directions to the house.

Records tell browsers which hosting service answers for your domain.

07 HOSTING

The land, power, and building.

The computers and network that keep your website available online.

08 HTTPS / SSL

The locked front door.

An encrypted connection between the visitor and the website.

SESSION 1: build the home, walk through it, ask for changes, and save it.

LATER: custom domain, DNS, advanced hosting, and integrations.

A SIMPLE 20-MINUTE TEACHING PATH

0-3 MIN

What is a site?

A message, arranged visually, with one next step.

3-8 MIN

Decode the words

Home, address, directions, hosting, security.

8-12 MIN

Start with Sites

Describe, preview, refine, save. Fewer handoffs.

12-17 MIN

Compare paths

Sites, Claude, template builders, custom hosting.

17-20 MIN

Draw the boundary

Build today. Domain, DNS, and extras can wait.

FOUR VALID PATHS. ONE SIMPLE START.

Recommendation for this class: start with ChatGPT Sites because it combines the beginner workflow - create, host, refine, version, deploy, and share - without requiring a separate deployment setup.

CHATGPT + SITES

BEST FOR: THE SHORTEST BEGINNER PATH

- **Prompt to hosted site:** describe, preview, refine, save, deploy, share.
- **Visual advantage:** image generation is available in the same ChatGPT/Codex environment.
- **Team reality:** access can be limited to owners, users/groups, a workspace, or the public where supported.
- **Trade-off:** Sites is in public beta; plans, limits, runtimes, and sharing options vary.

CLAUDE + ARTIFACTS

BEST FOR: ARTIFACTS, CODE, AND ITERATION

- **Can build:** single-page HTML sites, React components, SVGs, diagrams, and interactive artifacts.
- **Can publish/share:** availability and audience rules depend on the Claude plan.
- **Visual difference:** Claude does not generate photos or illustrations; it can create HTML/SVG visuals.
- **Practical effect:** high-end site imagery may require another image tool or supplied assets.

TEMPLATE SITE BUILDERS

BEST FOR: VISUAL CONTROL AND PACKAGED FEATURES

- **Strengths:** templates, visual editors, commerce, booking, forms, and established ecosystems.
- **Customization:** excellent when you want to control every page and setting manually.
- **Trade-off for this class:** more menus, choices, clicking, and platform-specific learning.
- **Use later when:** a feature or ecosystem clearly serves the business better.

CODEX + RAILWAY / CUSTOM

BEST FOR: CUSTOM APPS AND INFRASTRUCTURE

- **Railway's job:** deploy code and services from GitHub, a local project, Docker, or templates.
- **More power:** custom backends, databases, environments, logs, scaling, and deployment controls.
- **Not a design tool:** the website must already be designed and built in code.
- **Use later when:** the project outgrows the supported Sites runtime or needs custom infrastructure.