01 / START YOUR PROJECT

Make the template your own.

A practical guide to building a website about an artist or musician of your choice.

Your assignment Replace the instructional text and every image placeholder. Write original content, credit your sources, choose your own link destinations and create backlinks to your finished project.

Start with a working copy

Extract `Student-Artist-Musician-Template.zip` before editing. Make a backup of the extracted folder. Open that folder in a code editor, and open `dist/index.html` in a browser to see the website.

The project folder contains `dist/`, `README.md`, `sync-book.py` and `verify.py`. The `dist/` folder is the website. Keep its subfolders and filenames together.

Plan your content before you edit

Choose one subject and collect: a short introduction, a biography with five milestones, six works or releases, four creative processes, three useful destinations, and image and research credits. Adapt the wording for visual art, music or performance.

Use this editing cycle

1. Edit	2. Save	3. Refresh	4. Check
Change one section in its source file.	Save your HTML, JavaScript and images.	After HTML edits, run the cache helper (page 6).	Reload and test the changed page and controls.

Find the instructions you need

Pages	What you will do
2-3	Replace text, branding, metadata and images.
4-5	Update interactive examples, links and backlinks.
6-7	Refresh the book, fix common problems and prepare your submission.

Project note: this guide describes the supplied seven-page student template. Keep seven pages and the existing example counts for your first version.

02 / EDIT TEXT & IDENTITY

Replace the text & identity.

Edit the existing content while preserving the HTML structure that keeps the book working.

File inside dist/	Replace with your own content
`index.html`	Site identity, opening message, featured image caption and buttons.
`collection.html`	Collection introduction, category labels and the fallback gallery inside `<noscript>`.
`artist.html`	Biography, portrait caption, expandable details and five timeline milestones.
`craft.html`	Process introduction, four tab labels, starting process panel and feature banner.
`visit.html`	Three destinations, their images, descriptions and links.
`credits.html`	Your image credits, research sources and backlink record.
`about-template.html`	A final about page explaining your project and creative choices.

Change the words, keep the tags

For example, replace the words inside a paragraph. Preserve the opening and closing tags and identifiers such as `id="main"` and `id="book-page"`. Keep the page filenames for your first version.

```
<p class="hero-description">
  Introduce your chosen artist or musician in your own words.
</p>
```

Update the shared parts on all seven pages

The brand, monogram, desktop and mobile navigation, footer, page controls and dialogs are repeated. Use your editor's search across files to find `YOUR ARTIST` and other prompts. Change visible labels and useful accessibility labels, but keep working `href` values and structural IDs.

Give each page its own metadata

In each `<head>` section, replace the `<title>`, description and keyword prompts. A `rel="canonical"` link identifies the final address of that specific page. Replace the old preview addresses when you publish your own site.

Remember: gallery cards and many dialog details come from `app.js`, not only the HTML. Continue on page 4.

03 / REPLACE THE IMAGES

Replace every image.

The dimension labels show source image sizes in pixels, not fixed sizes on a visitor's screen.

Placeholder filename in assets/	Size in pixels	Common uses
`placeholder-1440x825.svg`	1440 x 825	Home, work 01, destination 1
`placeholder-1280x968.svg`	1280 x 968	Work 02, process 2, banner, destination 2
`placeholder-1280x1181.svg`	1280 x 1181	Work 03, process 1
`placeholder-960x1440.svg`	960 x 1440	Work 04
`placeholder-1280x1285.svg`	1280 x 1285	Work 05, process 4, destination 3
`placeholder-885x1200.svg`	885 x 1200	Portrait, work 06, process 3
`placeholder-192x192.svg`	192 x 192	Browser tab / site icon

Replace an image in four steps

1. Export a JPG or PNG you can use, preferably at the listed size or a similar aspect ratio. Use a descriptive lowercase filename, such as `artist-portrait.jpg`. Put it in `dist/assets/`.

2. Search the source HTML files and `app.js` for the old placeholder filename. Replace each relevant reference. The same placeholder may appear in several contexts, which can use different new images.

3. Set `width` and `height` to the new file's actual pixel dimensions. Write an `alt` description, update `title`, and replace the visible caption and credit.

4. Refresh the book cache after HTML edits. Check the image on its page, in the gallery viewer and in any process tab that uses it.

```
<img src="assets/artist-portrait.jpg"
     width="885" height="1200"
     alt="Describe the person, setting and relevant action."
     title="Your artist name - portrait">
```

Do not just rename an SVG to .jpg. Export a real image file. If you use a PNG site icon, update its path, `type="image/png"` and sizes in all seven HTML files.

The supplied CSS uses `object-fit: contain` to keep whole images visible. For intentional crops, adjust selected rules in `student.css` and check the result at both phone and desktop widths.

04 / EDIT INTERACTIVE CONTENT

Update the interactive content.

Edit the data near the top of dist/app.js. Keep its quotes, commas, braces and square brackets intact.

Data group	What to edit	Keep connected
`artworks`	Six gallery examples: titles, dates, descriptions, images, credits and source/related links.	Keep IDs `work-01` through `work-06`.
`mediums`	Four process tabs: labels, headings, descriptions, images and linked example.	Keep keys `process-1` through `process-4`.
`venues`	Three destination choices: name, location/platform, description and URL.	Keep keys `destination-1` through `destination-3`.

Example: edit fields inside an artwork record

The excerpt below uses a fictional title. Replace these values inside the existing `work-01` record; leave its other fields in place.

```
"title": "Night Sessions",
"date": "2025",
"medium": "Live performance",
"location": "Your chosen venue",
"image": "assets/night-sessions.jpg",
"alt": "A singer at a microphone beside a pianist.",
"imageTitle": "Night Sessions - live performance",
"width": 1440,
"height": 825,
```

Complete each record

Add your own `description` and `credit`. Set `source` to the image source and `visit` to a relevant work, venue or artist page. In `mediums`, the `art` value must match an existing work ID. In `venues`, replace each `url`.

Keep labels and behavior in agreement

If you rename categories, update the `category` values in `artworks` and the matching `data-filter` values and button labels in `collection.html`. Use simple lowercase keys such as `albums` or `performances`.

Match the initial process panel in `craft.html` to `mediums["process-1"]`. Match the selector options and initial destination dialog in all HTML files to `venues`. Also replace the generated button labels in functions such as `renderCollection`, `resetZoom` and `showArtwork` if they still say "placeholder" or "example".

05 / BUILD LINKS & BACKLINKS

Create your own links & backlinks.

Use descriptive link text and relevant destinations. A source link and a backlink point in different directions.

Link type	Direction	Project example
Internal link	Your site to another page on your site	A biography links to a related work in your collection.
Source / outgoing link	Your site to another website	An image credit links to the page where you found the image.
Backlink	Another website to your site	Your portfolio or an approved class directory links to your finished project.

Replace the unfinished link slots

Search source HTML and `app.js` for `credits.html#backlink-guide`. For each unfinished slot, add your chosen URL and useful link text. Remove `data-link-placeholder` from completed HTML links. Update JavaScript `source`, `visit` and `url` values too.

The addresses below are examples to replace. The first is an internal link; the second opens a source in a new tab.

```
<a href="artist.html">Read the artist biography</a>

<a href="https://example.com/artist"
   target="_blank" rel="noopener noreferrer">
  Visit the artist's official website
</a>
```

Build and verify a real backlink

1. Publish the project at its final address. **2.** On a portfolio, class directory or other page you control or have permission to edit, add a descriptive link to that address. **3.** Click it from the other page and confirm it opens the correct project. **4.** Record the referring page and destination on your sources page.

Linking to someone else's website does not automatically create a backlink to yours. If you cannot edit the other page, request permission or use another appropriate page you control.

Credit the work you use

On `credits.html`, list image titles, creators, source URLs and relevant permission or license information. Cite the research behind your biography and work descriptions. Keep the bundled font license notices. Replace the student backlink guide with your own notes when all link slots are complete.

06 / PREVIEW & TROUBLESHOOT

Refresh, preview & troubleshoot.

Save the source files first. Then rebuild the book cache so direct page loads and animated page turns show the same content.

Run the cache helper from the project folder

Open a terminal in the folder that contains `sync-book.py` and `dist/`. Run:

```
python sync-book.py
```

On Windows, `py sync-book.py` may work if the Python launcher is installed. If neither command is available, ask your instructor for help setting up Python. A successful run reports that seven pages were cached. Do not edit `book-pages.js` directly.

Preview: reload `dist/index.html` in your browser. Visit every page directly and through the numbered navigation. Open all gallery cards, change filters, try the process tabs and use the destination chooser. Check the mobile menu and keyboard controls.

If you prefer a local server, run the following from the same project folder, visit `http://localhost:4173`, and press Ctrl+C in the terminal when finished.

```
python -m http.server 4173 --directory dist
```

Common problems and fixes

What you see	What to check
Old copy after turning a page	Run `sync-book.py` after saving the HTML; then reload the browser.
A missing image	Check the filename, letter case, extension and `assets/` path. Check the HTML and JavaScript records.
An empty filter or wrong example	Match category keys to `data-filter` values. Match process `art` values to work IDs.
A gallery or tab stops working	Undo the last edit and check for a missing quote, comma or bracket in `app.js`.

Optional file checks: run `python verify.py` to check local files, page structure, metadata, asset hashes and the cache. If you change or remove registered assets, update `dist/asset-register.json` too; changed files need new byte sizes and SHA-256 values. Ask your instructor for help if needed. The checker does not verify external links or judge your writing and credits.

07 / PUBLISH & SUBMIT

Publish & submit your project.

Finish the content first. Use your instructor's required hosting and submission method.

Prepare the final website

Replace the old preview canonical URLs in all seven HTML files with the final address of each page. Save and run `python sync-book.py` again. For static hosting, upload the **contents of** `dist/`, keeping `index.html`, the other pages, scripts, styles and `assets/` together. Do not upload Git history, credentials or local QA files.

Final student checklist

- ☐ My site consistently features one chosen artist or musician.
- ☐ I replaced the site name, monogram, student instructions, headings, dates, captions and footer.
- ☐ I added original writing for all seven pages and credited my research.
- ☐ I replaced every displayed placeholder, including gallery details, process tabs and the site icon.
- ☐ My images have accurate dimensions, useful descriptions and credits.
- ☐ All six work records, four processes and three destinations are complete and consistent.
- ☐ I replaced unfinished link slots, checked external destinations and documented my own backlinks.
- ☐ Each page has a suitable title, description and final canonical address.
- ☐ I rebuilt the cache and checked every page, dialog, filter and menu on narrow and wide screens.
- ☐ I kept the font notices, saved a backup and tested the published website from its real URL.

Include with your submission

Submit the final website address and, if requested, a ZIP of your project folder. Include your source credits and backlink record. Keep `sync-book.py` with your editable project so future HTML changes can be synchronized.

Student: _____ Class: _____

Artist / musician: _____