01 / EDITING YOUR CULTURAL FOOD WEBSITE

# Start here

A practical guide to the HTML, images, folders, and CSS in your student restaurant website.

**1. Make a working copy.** Copy the whole project folder before editing. Keep its internal folder structure intact.

**2. Open two views.** Open an HTML file in a plain-text/code editor to change it. Double-click index.html to preview the website in a browser. Do not use Word to edit code.

**3. Change, save, refresh.** Make one small change, press Ctrl+S, and refresh the browser with Ctrl+R. Use Ctrl+F in your editor to find the exact visible text.

**4. Keep the file type.** Save as UTF-8 and keep .html, .css, or .js. In File Explorer, show file name extensions so a file does not accidentally become index.html.txt.

## Where each edit belongs

| File or folder | What it controls |
|---|---|
| index.html | Home page, introduction, featured dishes, and story previews. |
| menu.html | 27 menu cards, category filters, names, descriptions, and prices. |
| about.html / contact.html | Your food story and project contact/event information. |
| blog.html | Food-story listing, search, and category filters. |
| pages/blog/ | blog_1.html through blog_7.html are articles; blog.html is an additional listing. |
| assets/theme.css | Shared colors, fonts, spacing, cards, and responsive layout. |
| assets/theme.js | Mobile navigation, filtering, search, and preview-form behavior. |
| assets/placeholders/ | Current labeled SVG images. Put replacement photos in assets/images/. |
| tools/ / backups/ / output/ | Maintenance scripts / earlier files / ZIP packages and review outputs. |

Read next: HTML text (2), menu cards (3), images (4), CSS (5), links and pages (6), checking and sharing (7).

# Edit HTML text

HTML supplies the content and structure. Change the words inside elements while keeping their tags intact.

## Replace a heading and paragraph

In index.html, search for **[Your food.** or a sentence you can see in the preview. Replace the prompt with your own writing. The example below is a model, not a claim about your background.

```
<h1>Food from my table</h1>
<p>I share the dishes, traditions, and food memories
   that connect me with my family and community.</p>
```

| HTML | Meaning and editing tip |
|---|---|
| <h1> … </h1> | Main page heading. Keep one clear main heading per page. |
| <h2> / <h3> | Section headings / smaller headings such as dish names. |
|  …  | A paragraph. Replace the prompt; keep the closing tag. |
| <br> /  | A line break / emphasized text. Keep only where useful. |
| class="…" / id="…" | Styling and behavior hooks. Leave these intact when changing text. |

## Update the browser title and description

Near the top of each HTML file, find the <title> and description meta tag. Give every finished page its own accurate title and short summary; these are separate from the visible heading.

```
<title>My Cultural Food Menu | Student Food Project</title>
<meta name="description"
      content="Explore my favorite dishes and the stories
               and traditions that make them meaningful.">
```

## Keep code and writing separate

Use **&** in HTML code to display an ampersand, such as **Sweets & drinks**. For literal angle brackets, use **<** and **>**. Keep straight quotation marks around attribute values.

Do not delete surrounding <section>, , <article>, or closing tags just to remove a sentence. To remove a whole card, select its complete opening-to-closing block.

**Repeated content:** Headers, footers, and contact forms are copied into the HTML pages. Editing one copy does not update all other pages. Search the project for the old wording and update the relevant copies.

# Edit menu cards and filters

Keep student writing prompts until you are ready to replace them with your own food descriptions.

**Current categories:** Starters, Everyday meals, Main dishes, Shared plates, and Sweets & drinks. The Everything button shows all 27 entries.

## Edit one complete card in menu.html

Find **id="dish-01"**. Keep the ID, image class, and layout wrappers. Replace the heading, price, and paragraph. Here is a shortened example using the existing placeholder:

```
<article class="menu-item" data-category="Starters"
         id="dish-01">
  <img class="menu-photo"
       src="assets/placeholders/dish-01.svg"
       width="900" height="600"
       alt="Placeholder for my starter photo" loading="lazy">
  <div class="eyebrow">Starters</div>
  <div class="meta">
    <h3>[Your dish name]</h3>
    <span class="price">[Price]</span>
  </div>
  <p>[Describe ingredients, flavor, and your connection.]</p>
</article>
```

## Rename a category in three places

**1. Filter button.** Change both the button label and its data-filter value.

**2. Matching cards.** Change data-category on every card in that group to the exact same spelling and capitalization.

**3. Visible labels.** Change the eyebrow text on those cards, then click the filter in the browser.

```
<button type="button" data-filter="Family favorites"
        aria-pressed="false">Family favorites</button>
<!-- Matching card attributes and label: -->
<article class="menu-item" data-category="Family favorites"
         id="dish-01">
  <div class="eyebrow">Family favorites</div>
  <!-- Keep the rest of the card here. -->
</article>
```

**Add or remove a dish:** Duplicate a full card and assign a unique ID such as dish-28, or delete its whole <article> block. Reuse a valid category. Do not duplicate an existing ID.

**Home-page copies:** index.html features dish-07, dish-19, and dish-22. If you change those dishes, update their home-page titles, descriptions, prices, photos, and links too.

# Replace and organize images

Use descriptive filenames, suitable export dimensions, and the correct relative path from each page.

| Image use | Export size (pixels) | Suggested file target |
|---|---|---|
| Home hero | 1440 x 960 | 200 KB or less |
| Menu / story card | 900 x 600 | 100 KB or less |
| Home / About story | 1000 x 1000 | 150 KB or less |
| Article hero | 1200 x 800 | 180 KB or less |
| Community / location | 1200 x 480 | 100 KB or less |
| Header / footer logo | 620 x 152 | 30 KB or less |
| Social icon / browser icon | 64 x 64 / 32 x 32 | 5 KB or less each |

These are the template's export sizes and practical compression targets, not a guarantee of search ranking. Use WebP or JPEG for photos, PNG for transparency, and SVG for suitable vector logos/icons. Crop to the intended shape before exporting.

## Replace a menu placeholder with a photo

**1. Save the file.** Export a real 900 x 600 photo into assets/images/, for example family-dumplings-900.webp. Renaming an SVG to .jpg does not convert it.

**2. Update the HTML.** Change src, alt, width, and height to match the actual replacement. Keep the menu-photo class. Update every page that uses the same image.

```
<img class="menu-photo"
     src="assets/images/family-dumplings-900.webp"
     width="900" height="600"
     alt="Steamed dumplings with scallions on a blue plate"
     decoding="async" loading="lazy">
```

Write alt text that describes the actual image. Keep loading="eager" and fetchpriority="high" on the existing main hero image; use lazy loading for images farther down the page.

## Paths change inside nested folders

```
<!-- From index.html or menu.html: -->
src="assets/images/family-dumplings-900.webp"
<!-- From pages/blog/blog_1.html: -->
src="../../assets/images/family-dumplings-900.webp"
```

Each ../ moves up one folder. Use forward slashes in HTML paths, exact filename capitalization, and simple lowercase names with hyphens. Do not use a C:\Users\... path; it will only work on your computer.

**Keep an inventory:** Record each new filename and every page using it. Old photos already exist in assets/images/; remove a file only after checking that no HTML page references it.

# Change the CSS

assets/theme.css styles the whole website. One shared change can affect every page.

## Start with the color variables

The :root block near the beginning defines the palette and font families. Edit these existing values to change the site consistently:

```
:root {
  --paper: #f7f4ec;  /* main background */
  --ink: #29372e;    /* dark text and buttons */
  --muted: #62685e;  /* paragraph text */
  --accent: #a14e36; /* labels and prices */
  --line: #deded1;   /* borders */
  --soft: #eeeee4;   /* section backgrounds */
  --white: #fffdf8;  /* fields and light surfaces */
}
```

## Understand selectors and the cascade

**.menu-item** selects a class; **#reservation** selects an ID. Braces group declarations, a colon separates a property from its value, and a semicolon ends a declaration. The stylesheet contains repeated rules; later rules can override earlier rules with the same specificity.

For a small custom change, add a clearly labeled block at the end of the file, outside all existing braces. Test the result at both desktop and phone widths.

```
/* My custom styles */
.menu-item h3 { font-size: 24px; }
.menu-item p { line-height: 1.8; }

@media (max-width: 600px) {
  .menu-item h3 { font-size: 25px; }
}
```

## Useful places to look

| Selector | Changes you can make |
|---|---|
| .hero h1 / .page-intro h1 | Home heading / interior-page heading size and spacing. |
| .section / .wrap | Vertical section space / page content width. |
| .button / .button:hover | Button appearance / hover appearance. |
| .food-grid / .menu-list | Card layout: currently three columns, two at 900px, one at 600px. |
| .hero-image > img / .story-img | Image fit. Current placeholder rules use contain; cover fills a frame but crops. |

Keep readable text contrast and keyboard focus outlines. Avoid fixed heights on text blocks; long dish names and mobile layouts need room to grow. If a change appears ineffective, inspect the element in the browser to find the rule that wins.

# Manage links and pages

A visible label and its destination are separate. Change the right part for the result you want.

```
<a href="menu.html">Explore the menu</a>
<a href="#reservation">Book a table</a>
<a href="contact.html#reservation">Book a table</a>
<a href="https://example.com">Visit a website</a>
```

The first link opens another page. The second jumps to id="reservation" on the same page. The third opens that section on the contact page. Inside pages/blog/, use ../../contact.html#reservation for a contact-page link.

## Keep the booking button connected

The current navigation says **Book a table** and uses **#reservation**. Keep the form's id="reservation" for that jump link to work. The form is a display-only preview: entries are not sent or saved, and the button is disabled. Changing its label or removing disabled does not create a booking service.

## Edit or add a food-story page

**1. Existing article.** Edit pages/blog/blog_1.html through blog_7.html: headline, introduction, byline, photo, body, and credits.

**2. Listing copies.** Update the matching card in blog.html and pages/blog/blog.html. Update a home-page card if that story is featured there.

**3. New article.** Copy an article to blog_8.html in the same folder. Give it its own title and text, then add linked cards to the listings and adjust any next-story links. The site does not discover new files automatically.

## Headers, footers, and social links

Navigation, logo, social icons, and footer text are repeated across pages. Update each relevant HTML copy. Keep aria-current="page" only on the navigation link for the page being viewed. Social icons still have placeholder links; replace both the destination and accessible label when using a real profile.

## Link to this PDF if you choose

```
<!-- From a root-level HTML page: -->
<a href="HTML-CSS-EDITING-GUIDE.pdf">Editing guide (PDF)</a>
<!-- From a page inside pages/blog/: -->
<a href="../../HTML-CSS-EDITING-GUIDE.pdf">Editing guide</a>
```

Some existing links still point to STUDENT-EDITING-GUIDE.html, which is absent from the current folder. If you replace those links with this PDF, remove old #images or #social-links fragments; this PDF does not use those HTML section IDs. This guide does not change the site navigation.

# Check and share your work

Use the edited folder as your source of truth. A ZIP is a separate snapshot and does not update itself.

## A quick check after every change

**1. Preview.** Save the edited files, refresh index.html, and navigate through Menu, Our story, Blog, and Contact. Use Ctrl+Shift+R if old CSS or images remain cached.

**2. Try the controls.** Open the phone menu, click every food category, try the story search, and follow Book a table. Check that each image loads and each link reaches the intended place.

**3. Resize.** Check a wide desktop view and narrow widths around 900px and 390px. Look for horizontal scrolling, cropped words, large gaps, and text that touches its container edges.

**4. Read and tab.** Proofread the finished text, check titles and descriptions, and use the Tab key to confirm links and controls are reachable.

## Common problems and fixes

| Problem | First thing to check |
|---|---|
| A picture is missing | File extension, capitalization, src path, and whether ../../ is needed. |
| A category shows nothing | Button data-filter and card data-category must match exactly. |
| The page layout breaks | A deleted closing tag, unmatched quote, missing CSS brace, or duplicated ID. |
| A style does not change | Saved file, correct stylesheet link, cache, and later overriding CSS rules. |
| Only one page changed | Shared HTML sections are copied; update their other occurrences. |
| A link opens an error | Confirm that the named file exists and any #anchor ID is present. |

## Choose one editing workflow

**Recommended for students:** Edit the HTML and assets/theme.css directly. You do not need to run a build for this static website.

**For template maintainers:** tools/menu_data.py stores menu and story prompts; tools/build_theme.py generates pages and placeholders. Running the generator replaces manual HTML edits and can recreate guide pages. Back up first, edit generator sources intentionally, and review the output before sharing.

## Make a fresh handoff folder or ZIP

Copy the edited root HTML files, pages/, the assets actually used, and this PDF into a clean handoff folder while preserving their paths. Exclude backups, tools, screenshots, and older ZIPs. Compress that folder, extract a test copy, and open index.html to check it.

The existing ZIP under output/ is an earlier package. The current package_theme.py expects files that are absent and does not automatically include this PDF or new replacement photos. Use the manual handoff method above unless a maintainer updates and checks that script.