Portfolio Project Cheat Sheet (For Nicol)

This cheat sheet explains the words and requirements in your **Portfolio Project Brief**. Think of this as a quick reference to guide you step by step. Don't worry if it feels like a lot — once you break it down, it's simple.

Key Sections of Your Site

- **Hero Section** → The very top of your page. It usually has your name, a short statement (like "Aspiring Web Developer"), and a button that scrolls to your projects.
- **Projects Section** → A list or grid showing the projects you've built. Each project should have:
- Title
- 1–2 sentence description
- Tech tags (HTML, CSS, JS)
- Links to the live project and GitHub code
- **About Section** → A short bio (2–3 sentences) about you and what you're learning.
- **Contact Section** → A way for people to reach you. For V1, just a `mailto:` link that opens email, or a simple form that checks for valid input.

JavaScript Enhancements (Pick 2–3)

These are small interactive features to make the site feel dynamic: - **Filter Projects** → Buttons that show All / HTML / CSS / JS projects. - **Dynamic Project List** → Store project info in a JS array or JSON file, then display it with JS. - **Copy Link Button** → Button to copy a project's link to clipboard. - **Theme Toggle** → Light/Dark mode, saved with `localStorage`. - **Form Validation** → If you use a form, check that fields aren't empty and the email looks valid.

Git & GitHub Basics

- **Repo:** A folder on GitHub that stores your code online.
- **Branch:** A copy of your code to make changes safely before merging to `main`.
- **Commit:** A saved change. Write short messages like "add project cards" or "fix typo in README".
- **PR (Pull Request):** A request to merge your changes into `main`. For now, just self-review before merging.
- **Self-Review Checklist:** Before merging, ask yourself:
- Does it run without errors?
- Is it responsive on small, medium, large screens?
- Are links and buttons working?
- Any typos or broken code?

Deployment (Putting Your Site Online)

- Use **GitHub Pages**.
- In your repo settings → Pages → set source to `main` branch.
- Your site will be live at `https://<yourusername>.github.io/nicol-portfolio/`.
- Add this link to your README file.

V1 vs V2

- **V1 (Now):** Simple, static HTML/CSS/JS. Focus on structure, working links, and 2–3 JS features.
- **V2 (Later, after Front End Libraries):** Upgrade with React, Bootstrap/Sass, nicer styling, dark mode, and animations.

Design & Accessibility

- **Responsive:** Works on phone, tablet, laptop.
- **Contrast:** Text easy to read (dark text on light background or vice versa).
- **Alt text:** Every image should have an `alt` description.
- **Semantic HTML:** Use `<header>`, `<main>`, `<footer>`, not just `<div>` everywhere.

Tools to Keep Handy

- **VSCode** → Your editor.
- **GitHub Desktop (optional)** → Easier Git management if CLI is too much.
- **MDN Web Docs** → Look up HTML, CSS, JS references.
- **FreeCodeCamp Forum** → Help if you get stuck.
- **Lighthouse (Chrome DevTools)** → Check performance and accessibility.

Mindset

- This is your **first version** — it doesn't need to be perfect.
- Think of it like shipping a product: build something functional now, then upgrade it later.
- Keep commits small and push often.
- Ask yourself: *"Am I better than yesterday?"* That's the real goal.

Remember: V1 is about getting your portfolio live with working links and simple features. V2 will be the polished upgrade once you've learned React and libraries.