EIS Coding 101 - Workshop C - Web Basics

mini-project to build a Basic Personal Profile Page

This hands-on workshop guides complete beginners through creating and publishing a personal profile webpage. No prior coding experience needed!

What you'll achieve: A live website displaying your name, bio, photo, and a simple interactive element – all hosted for free online.

What You Need

- **Laptop with Wi-Fi** - Any operating system with a modern browser (Chrome, Firefox, Safari, Edge)
- **Personal photo** - JPG or PNG format (keep file size under 1MB)
- **Free accounts** - You'll create these during the workshop:
  - CodeSandbox (for coding)
  - GitHub (for publishing)

Part 1: Coding Your Profile Page

Step 1: Set Up Your Coding Environment

1. Visit CodeSandbox and create a free account
2. Click "Create" and choose "HTML + CSS" in Frontend section
3. Choose Sandbox in Runtime and create
4. You'll see a workspace with a file tree on the left side

Step 2: Upload Your Profile Photo

1. In the file panel (left side), right click to Upload
2. Select "Upload Files" and choose your photo
3. Your image will appear in the file list

Step 3: Use Some Template

1. Delete the original files HTML+Javascript+CSS
2. Right Click the file panel and click "upload"
3. Unzip the Template1, and choose the 3 files.
4. Edit the file using your information.

Step 4: Preview and Export

1. See your live preview in the right panel
2. Make any adjustments to your code as needed
3. When satisfied, right click left panel and download
4. Save the ZIP file to your computer (we'll need it for Part 2)

Additional Step: try more complex template (after part 2)

1. Use the template2 to create your page.

Part 2: Publishing Your Website

Step 1: Create a GitHub Account

1. Go to GitHub and sign up for a free account
2. Complete the email verification process

Step 2: Create Your Repository

1. Once logged in, click the "+" icon in the top-right corner
2. Select "New repository"
3. Name your repository: `username.github.io` (replace "username" with your GitHub username)
4. Make sure "Public" is selected
5. Click "Create repository"

Step 3: Upload Your Website Files

1. On your new repository page, click "Add file" → "Upload files"
2. Unzip the file you downloaded from CodeSandbox
3. Drag and drop all the files from the unzipped folder:
   - index.html
   - styles.css
   - script.js
   - Your photo file
4. Scroll down and click "Commit changes"

Step 4: Activate GitHub Pages

1. Go to the "Settings" tab of your repository
2. Scroll down to "Pages" in the left sidebar
3. Under "Source", select "Deploy from a branch"
4. Under "Branch", select "main" and "/(root)" folder
5. Click "Save"
6. Wait about 1-2 minutes for your site to deploy

Step 5: View Your Live Website

1. Return to the "Settings - Pages" section after waiting
2. You'll see a message saying Your site is live at with your site URL: `https://username.github.io`
3. Click the link "Visit site" to see your live profile page!

Troubleshooting Guide

Problem	Solution
Photo not displaying	Double-check the filename in your HTML. GitHub is case-sensitive, so "Photo.jpg" and "photo.jpg" are different files.

Problem	Solution
CSS not applying	Try a hard refresh (Ctrl+F5 or Cmd+Shift+R) to clear your browser cache.
Website not appearing	It can take up to 5 minutes for GitHub Pages to deploy. Check that your repository is named correctly (username.github.io).
Error during file upload	Make sure you're uploading the individual files, not the ZIP file or nested folders.

Next Steps & Resources

Continue Learning

- **HTML & CSS Basics**: MDN Web Docs
- **JavaScript Fundamentals**: JavaScript.info
- **Free Design Resources**: Unsplash (images), Font Awesome (icons)

Enhance Your Page

- **Add Social Media Links:**
    - Include icons and links to your social media profiles (e.g., LinkedIn, Twitter) using Font Awesome for icons.
- **Expand Your Content:**
    - Add additional pages for projects, a blog, or a resume. Use navigation links to connect these pages.
- **Use Advanced Frameworks:**
    - Explore frameworks like Next.js for server-side rendering and Tailwind CSS for utility-first styling to enhance your site's performance and design.
- **Leverage AI Tools:**
    - Use tools like ChatGPT for content generation and GitHub Copilot for coding assistance to streamline your development process.
- **Incorporate Interactive Elements:**
    - Add interactive elements like carousels, modals, or animations using JavaScript libraries such as jQuery or GSAP.

Congratulations!

You've successfully created and published your own website! You now have a professional online presence that you can share with friends, family, and potential employers. Feel free to continue experimenting and adding to your site as you learn more.Happy coding!