chawla22

Lordjesus2424@

Coding Challenge 01

Launching Your Digital Brand

In this first coding challenge, you’ll begin to establish your professional identity as a developer by creating and publishing a personal portfolio webpage. You will build your site locally in your Documents folder and then use GitHub to host and publish it via GitHub Pages.

This project introduces the foundational workflow you’ll use throughout the course: creating, editing, styling, versioning, and publishing your code.

Important Note

All future projects will be submitted through GitHub. Completing this challenge correctly ensures your environment is set up and ready.

Instructions

The first challenge will be to complete GitHub Portfolio Site. It should take about 2 hours to complete this challenge.

Challenge Instructions

✅ Step 1: Create a Folder for Your Project

Open File Explorer (Windows) or Finder (Mac)

Navigate to your Documents folder

Create a new folder named: portfolio-site

✅ Step 2: Open the Folder in VSCode

Open VSCode

Go to File > Open Folder...

Select the portfolio-site folder you just created

You are now ready to start coding!

✅ Step 3: Create Your HTML and CSS Files

Inside your portfolio-site folder:

Create a file named index.html

Include:

Your name

A short paragraph describing your career goals

A photo or placeholder image

Links to your GitHub, LinkedIn, or email

Create a second file named style.css

Add styling such as:

Background color

Font choices

Page layout

Link the CSS to your HTML using:

<link rel="stylesheet" href="style.css">

✅ Step 4: Initialize Git and Publish to GitHub

In VSCode, open the Source Control tab (Ctrl+Shift+G)

Click Initialize Repository

Go to the GitHub tab in the VSCode sidebar (or sign in if prompted)

Click "Publish to GitHub"

Name your repository: portfolio-site

Make sure it's set to Public

Click Publish

You may need to authorize GitHub and connect your account during this process.

✅ Step 5: Make Your First Commit

In the Source Control tab:

You should see your changed files listed

In the message box, type: "Initial portfolio page"

Click the checkmark icon to commit

Click “Sync Changes” to push your commit to GitHub

✅ Step 6: Publish Your Site with GitHub Pages

Go to your repository on github.com

Click Settings > Pages

Under Source, choose the main branch and / (root) folder

Click Save

GitHub will generate a live link:

https://yourusername.github.io/portfolio-site/

Open it in your browser to confirm it works.

Submission Instructions

There is one deliverable for this assignment.

GitHub Repository URL – e.g., https://github.com/yourusername/portfolio-site

GitHub Pages URL – e.g., https://yourusername.github.io/portfolio-site/

Overview for coding one

Total Points: 100

Instructors and peer reviewers should use the criteria below to assign points. Each criterion includes specific directions and what to look for when verifying the student submission.

Why Peer Grading Matters

Peer grading is a valuable part of the learning process because it encourages you to think critically about code quality, structure, and presentation—just like a real developer reviewing a teammate's work. By evaluating others, you reinforce your own understanding, recognize common mistakes, and develop a stronger eye for clean, functional, and well-documented code. It also builds essential skills in collaboration, feedback, and professionalism, all of which are crucial in today’s team-based development environments.

Detailed Grading Rubric

✔️ Local Folder Created and Properly Set Up (10 points)

✅ Full credit: Folder named portfolio-site in Documents with index.html and style.css files inside.

❌ −5: Wrong folder name or misplaced files.

❌ −5: Missing either HTML or CSS file.

✔️ HTML Page Includes All Required Elements (20 points)

✅ Includes: name, career goals paragraph, image or placeholder, and at least one contact link.

❌ −5 per missing element (up to −20).

✔️ CSS Styling Applied and Linked Correctly (20 points)

✅ CSS file is created and properly linked with at least three style elements: background color, font customization, and layout adjustments.

❌ −10: No link to CSS file or broken path.

❌ −5: Styling minimal or ineffective.

✔️ GitHub Repo Created and Code Pushed via VS Code GUI (20 points)

✅ Public repo named portfolio-site, initial commit labeled “Initial portfolio page,” and VS Code Source Control used.

❌ −10: Repo not public or misnamed.

❌ −5: Commit message missing or clearly not made in VS Code.

✔️ GitHub Pages Live Site Published and Accessible (20 points)

✅ Site is accessible at the GitHub Pages link and reflects local content.

❌ −10: Site not published or link broken.

❌ −5: Page loads but doesn’t match repository files.

✔️ Clear Organization and Professional Presentation (10 points)

✅ Layout is readable, clean, and reflects effort (spacing, font size, no overlapping elements).

❌ −5: Disorganized or difficult to read.

❌ −5: Template untouched or no personalization.

Coding Challenge 02

Customer Records Tracker: Managing Clients with Arrays and Objects

In this challenge, you will build a system for managing customer records using JavaScript arrays and objects. This small business-focused project will help you practice creating and manipulating compound data structures that store and organize information about customers.

You will use arrays to manage a list of customers and objects to store details about each customer including their name, email, and purchase history.

Important Note

Be sure to test your code using console.log() frequently. Follow clean coding practices with readable formatting and descriptive variable names.

Instructions

This challenge is intended to take approximately 1-2 hours.

Challenge Instructions

✅ Step 1: Create a New JavaScript File

Create a folder named: customer-tracker

Inside the folder, create a file named: cc\_2.js

✅ Step 2: Create Customer Records

Create an array called customers.

Inside this array, create at least three customer objects.

Each customer object should have the following properties:

name (string)

email (string)

purchases (array of strings)

✅ Step 3: Add and Remove Data

Add a new customer to the customers array using .push().

Remove the first customer using .shift().

✅ Step 4: Update Customer Info

Update the email of one existing customer.

Add a new purchase to a customer’s purchases array using .push().

✅ Step 5: Display Customer Information

Loop through the customers array using .forEach().

Log each customer’s name, email, and total number of purchases using template literals.

Submission Instructions

Submit the following deliverables via GitHub:

GitHub Repository URL – e.g., https://github.com/yourusername/customer-tracker

Ensure your repository contains the cc\_2.js file and has a clear commit history documenting your progress.