Project Instructions

Project 9: Capstone Portfolio

Before you start:

1. Have a GitHub account and know how to create a new repository and upload files to it.

2. Have at least 4 portfolio-worthy projects for your portfolio.
   - You can use projects you created in this program or other ones you've built on your own and show off your technical skills with HTML, CSS, and JavaScript.

3. Place your finished portfolio and project files online using either:
   - a web hosting company
   - or, free hosting on GitHub pages, as described in the Publishing Websites Using GitHub Pages workshop (See the Project Resources for the link).

Instructions:

1. **Mobile first design.**
   - Create a mobile-first site that works well on mobile phones and desktop computers.
   - The design should be easy to use on smaller screen sizes as well as on the desktop.
   - 320px is the smallest screen size you will need to test your design on.
   - Make sure the HTML files include the viewport meta tag in the head of the document, see the project resources for links to "Using the viewport meta tag" and "Configuring the Viewport".

2. **Images that link to at least 4 other projects.**
   - The images can be images of your projects, icons, or whatever other visual represents the project.
   - You need to link to live, functional versions of other web projects. You can use some (or all) of the projects you've completed in this program. You can include the files for those projects along with the portfolio pages.

3. **A short description of each project and the skills you used to build it.**
   - Employers want to know what you can do, so let them know what the project was and what you used to build it.
   - For example, if you include Project #7 -- the SVG Web Site Update -- you could describe what you did (replaced raster images with scalable vector graphics) and the skills you used -- HTML, CSS and an understanding of SVGs.

4. **Contact information**
   - You'll want employers to be able to find and contact you.
   - You should include information like email, Twitter, and LinkedIn profile links for example. If you don't want to share this information for this project, no problem, just make these up.

5. **At least one example of JavaScript interactivity.**
   - For example, you could use the pop-up effect from Project #5, the Interactive Photo Gallery. Or, use JavaScript or jQuery to trigger CSS transition effects or animations.
   - Have fun with this -- JavaScript is an important part of front end web development, so providing multiple examples of your JavaScript abilities will impress employers.

6. **Put your project in a new GitHub repository on your GitHub account:**
   - See the workshop Share your Projects with GitHub Desktop (link in the Project Resources). The GitHub Desktop application can automatically create a new repository for you.
   - Make sure that you're only putting the files for this project in that repository.

7. **Publish your portfolio on GitHub pages, or with a web hosting company, so that the code reviewers can see your finished work.**
   - Make sure you add a link to your project’s URL in the comments section when you submit your project for review.

8. **Make sure to check your code is valid by running it through an HTML and CSS validator.**
   - Links to the validators can be found in the Project Resources. This will help you spot any errors that might be in your code.
   - There are a few exceptions that you don’t need to fix:
     - Don’t worry about any warnings, we just need you to check any errors that might be there.
     - If CSS validator flags use of calc, vendor prefixes or pseudo-elements/pseudo-classes these errors should be ignored.
     - If HTML validator flags use of pipe (‘|’) in Google font links/URLs this error can also be ignored.

9. **You should also check for issues with your JavaScript code using JSHint, linked in the Project Resources.**
   - JSHint may show you some warnings, so make sure to check through those for any potential problems.
   - You do not need to fix every warning listed but reviewing them can be useful.

NOTE: A good practice is to check your project for cross browser compatibility. Making sure that it looks and functions correctly in multiple (at least three) browsers is an important part of being a top-notch developer. If you want, leave a comment to the project reviewer about which browser(s) the project was checked to ensure they are seeing things as you have designed them.

Some browser options:

- Google Chrome
- Mozilla Firefox
- Internet Explorer/Edge
- Safari

EXTRA CREDIT

- Create a breakpoint for tablet displays.
- Link to more than 4 projects.
- Use SVG images for contact information icons.
- Multiple JavaScript interactions.