

Create your personal web page

Using HTML and CSS, create your own portfolio.

Constraints

- Your project must contain several HTML documents, with a full navigation system between all pages.
- Your project must have at least the following pages:
 - `index.html` - the main page
 - `blog/index.html` - the "main page" for your "blog". It must have links to all posts (see below).
 - `blog/post1.html` - blog post #1. It must have a link to the *next post* (#2) (and navigation links).
 - `blog/post2.html` - blog post #2. It must have a link to the *previous and next posts* (#1 and #3) (and navigation links).
 - `blog/post3.html` - blog post #2. It must have a link to the *previous post* (#2) (and navigation links).
- Your project must use images.
- Your project must have at least one CSS file used by all pages, with a consistent design throughout.
- Your project files must be neatly laid out (all images in the same folder, all CSS files in the same folder, etc).

Main page

This is the main page for your site. It must contain at least:

- your name and contact details (you can use fake information)
- a profile picture
- your current job title (you can use "Student", "Junior web developer", etc)
- a section with your previous job experiences
- a section with your past education (if relevant)
- a section with your current education
 - it must include a list with at least 3 courses that you are taking at BCIT this semester
 - each course must have the course name, number, and a short summary (you can use the course outline page)
- anything you feel relevant or interesting

Challenge yourself

The contents of your pages do not matter that much - you can use fake information and pictures. However, you should challenge yourself and experiment with all HTML and CSS attributes / properties you have seen so far.

- Make sure your HTML and CSS are valid using online validators.
 - [W3C validation service](#)
 - [CSS validation service](#)
 - [Contrast checker \(for WCAG / accessibility\)](#)
- Use semantic elements to structure your page (`<main>`, `<aside>`, `<section>`, etc).
- Use generic classes to build your own set of "styled elements" to keep a consistent design.

This is your practice for the coming exams.

Resources

- Image placeholders
 - <https://placeholder.co/>
 - <https://picsum.photos/>
- Text generators: <https://loremipsum.io/ultimate-list-of-lorem-ipsum-generators/> (don't overuse them!)