

Lesson

Weekend

# Introduction to Programming

## (/introduction-to-programming)

### / Git, HTML and CSS (/introduction-to-programming/git-html-and-css)

### / Git, HTML, and CSS Objectives

Text

In this section we will be learning the building blocks of making basic web pages: HTML and CSS. We will also focus on using the command line and Git to create a daily workflow for our coding projects.

## Daily Goals

Everyone learns at a different pace. Understanding is the *most* important goal every day. **If you feel like you will not be able to complete all of the lessons in a given day, talk to a teacher about ways to ensure that you are covering the day's objectives.** And *always* focus on understanding the concepts at hand, *not* finishing the work as quickly as possible.

Lessons that have "Homework" in the title indicate that the work is expected to be completed at home and not in class.

## Plan

We will begin this section by practicing the command line, Git, Github, and exploring basic HTML. We'll create READMEs to include extra information with our projects on Github. We'll also learn a formatting language called Markdown to format READMEs. Then, we'll explore basic CSS to add styling to our web pages, and actually publish our sites online! Later in the section we will dive deeper into CSS by using divs and spans, floats, cascading, classes, ids, and the box model. Finally, we'll explore a tool called Bootstrap that offers even more powerful styling capability. You're in for a great time!

At the end of this section, you will be able to:

- Navigate the command line to access and create directories and files
- Use Git and GitHub to track the creation of static webpages
- Create a static webpage using HTML
- Style a static webpage using custom CSS rules in a linked stylesheet
- Create a project README.md in Markdown
- Apply Bootstrap rules and classes
- Adapt a webpage for responsive design

## Independent Project Objectives

---

At the end of this section you will complete an independent project. Your instructor will review this project for the following objectives:

- The following HTML tags are *all* used: `p`, `h`, `ul`, `ol`, `li`, `em`, `strong`, `a`, `img`, `div`, and `span`.
- Project includes a custom-made stylesheet that uses typography styles, the box model (including margin, border, and padding), and floats.
- Code is clean, well-refactored, and easy-to-read. This includes correct indentation, spacing, and including only necessary comments and debugging tools.
- Commits are made regularly with clear messages that *finish* the phrase "It will..."

- The project repo contains a README that includes:
  - application name
  - names of contributors
  - description of application
  - link to site on GitHub Pages
  - a "technologies used" section
  - application setup instructions
  - link to site on GitHub Pages
  - a "known bugs" section
  - copyright, date, and license information
- Site is successfully deployed to GitHub pages.
- Project is in a polished, portfolio-quality state.
- The prompt's required functionality and baseline project requirements are in place by the deadline.

## What is a polished, portfolio-quality state?

When a project is both polished and in a portfolio-quality state, this means:

- You've reviewed your project and your README prior to submitting it to make sure there are no errors or missing information and you are consistent in your indentation, spacing, and code structure.
- You are following the best practices and coding conventions we teach. Make sure that your:
  - Code is clean, well-refactored, and easy-to-read. This includes correct indentation, spacing, and including only necessary comments and debugging tools.
  - Variable names are descriptive and use lower camel case (e.g. `myVariableExample`).
  - Commits are made regularly with clear messages that finish the phrase "It will...".

## What are the baseline project requirements?

All independent coding projects at Epicodus have these baseline requirements:

- A complete and informative README
- The project's commit history demonstrates that the project's required work schedule and hours have been met:
  - 8 hours completed on Friday is required for full-time students
  - 4 hours completed over the weekend is required for part-time students
- Completion of the project based on the prompt *and* objectives. The prompt contains details on the project's theme and features that are not always detailed in the objective. Carefully read through the prompt towards the end of your work session to make sure that you are not missing anything.

Next (/introduction-to-programming/git-html-and-css/monday-schedule-and-expectations)

Lesson 1 of 64

Last updated February 28, 2023

disable dark mode



© 2023 Epicodus (http://www.epicodus.com/), Inc.