

Lesson 3: Intro to HTML

Web Lab

Overview

In this lesson students are introduced HTML as a solution to the problem of how to communicate both the content and structure of a website to a computer. The lesson begins with a brief unplugged activity demonstrating the challenges of effectively communicating the structure of a web page. Students then look at an exemplar HTML page in Web Lab and discuss with their classmates how HTML tags help solve this problem. Students then write their first HTML. A wrap-up discussion helps to solidify the understanding of content vs. structure that was developed throughout the lesson.

Purpose

This lesson introduces many new concepts and tools to students. They are introduced to HTML, the Web Lab tool, and how to navigate lesson resources on Code.org in general. While the understanding of HTML as a way to communicate the structure of a web page is a critical learning objective, this lesson has students do minimal programming since there are many other new ideas and tools to grasp. In the next lesson students will have more time to spend programming in HTML.

Agenda

Warm Up (10 mins)

The Need for HTML

Activity (30 minutes)

Exploring HTML

Wrap Up (5 minutes)

Reflection

Objectives

Students will be able to:

- Explain that HTML allows a programmer to communicate the way content should be structured on a web page
- Write a simple HTML document that uses opening and closing tags to structure content
- Understand how to use lesson resources provided in Web Lab

Preparation

- Review the Code Studio levels

Links

Heads Up! Please make a copy of any documents you plan to share with students.

For the Teacher

- **Exemplar Text Website**

Vocabulary

- **HTML** - Hypertext Markup Language, a language used to create web pages
- **HTML Element** - A piece of a website, marked by a start tag and often closed with an end tag
- **HTML Tag** - The special set of characters that indicates the start and end of an HTML element and that element's type
- **Website Content** - The raw text, images, and other elements included in a web page
- **Website Structure** - The purpose of different pieces of content in a web page, used to help the computer determine how that content should be displayed

Introduced Code

- `<p></p>`
- `<html></html>`
- `<head></head>`
- `<!DOCTYPE>`
- `<body></body>`

Teaching Guide

Warm Up (10 mins)

The Need for HTML

Display: Show the image inside the **Exemplar Text Website**

Prompt: Imagine you wanted to explain to a person over the phone how to draw this web page. Write down as clear instructions so that what they would draw would perfectly match this image.

Discuss: Once students have written their instructions have them briefly share their instructions with a neighbor.

Demo: Run a quick demo using the instructions below.

- Pick one student to verbally share one of their instructions with you.
- The teacher should act as the person on the phone trying to draw the web page
- Publicly “draw” the website exactly as the students instructions say. For example, if told to "Write bigger", write the word "bigger" on the page. If they don't indicate where text goes then place text in random locations.
- As the student gives you instructions have them tell you if you have drawn it correctly. If you have not drawn it correctly, have them make their directions more specific until you can draw it correctly.
- Change students after a couple instructions to get more students involved.
- Keep track of the instructions students give and the improvements they make to the instructions somewhere visible as well.
- Repeat this process until you have recreated most of the web page.

Discuss: Once you have finished drawing the site, quickly create a list of all the different kinds of information they needed to account for in their instructions. For example, location, size, font, etc.

Remarks

There's a lot of information that we need to communicate if we want to create web pages. It's not enough to just know what content you want to put on your page, like the actual words or images. You need to know where things should be and how they should look. Today we're going to start learning the languages used on the web to represent this additional information.

Activity (30 minutes)

Exploring HTML

Remarks

Today we are going to start working with a lot of resources. As we discover each type of resource, we'll add it to the list here at the front of the room, and at the end of the lesson we'll review how each is used.

Display: At the front of the room, write the heading "Resources" on the board or blank poster paper, leaving space to list the various resources as they appear in the lesson.

Discussion Goal

Goal: Activities like this one are often used in CS courses to help highlight just how much precision is needed to communicate instructions to a computer. In this instance the goal is similar. You want to highlight the challenge of differentiating the actual content on the page and instructions indicating how it should be structured. This demonstration helps justify the creation of HTML in order to tag pieces of content to help the computer understand what they are and hence how they should look.



Teaching Tip

Using Resources: Below you can find recommendations for using the many resources students are introduced to in the lesson. Wait until after students have seen all of these resources to review at the end of the lesson, but add them to the list and model the correct usage as they appear.

- **Videos:** Watched as a class, but students can always return to them.
- **Map Levels:** Contain text and diagrams explaining content. These are intended as helpful student resources, not class readings. They are a good place to go for review after learning content or when students get stuck in levels.
- **Level Instructions:** Instructions may introduce small pieces of new content. Each level features a "Do This" section explaining what students are supposed to do in that level. Set the expectation early that reading these instructions, not just the "Do This" section, is important.
- **Level Tips:** Students can click these tips in the instruction areas of lessons. Students should use these as a first place to check for help before talking with peers or a teacher.
- **Inspector Tool:** Highlights the code corresponding to a web page element when hovered over in the Preview Area.
- **Bubble Color:** Bubbles may turn green but there is no validation of correctness. Green only means a student clicked Continue or Finish for a level. Set the understanding early that this is more a tool for them than an indication of either completeness or correctness.

Code Studio levels

Lesson Overview

Student Overview

Experiment with Web Lab

2

(click tabs to see student view)

[View on Code Studio](#)

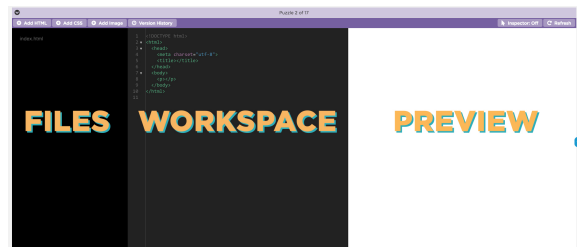
Student Instructions

Welcome to Web Lab!

The platform you are in is called Web Lab. There are three main parts of the screen in Web Lab.

Do This

- Try typing something in the workspace.
- Press to see it in the preview area!



Video: Intro to Web Lab - Part 1

Student Overview

[View on Code Studio](#)

[Download Video](#)

Explore HTML 

[Teacher Overview](#)

[Student Overview](#)

[View on Code Studio](#)

HTML Discussion

You should bring students back together once they've spent a couple minutes looking through this level. The discussion prompts listed in the level should be used in a standard Think-Pair-Share structure.

- What text is appearing in both the code and the web page?
- How is this language communicating extra information about the way to represent text?

The goal of the discussion is to call out the features of HTML that students are noticing. The two primary takeaways (reinforced in the subsequent video as well) are that HTML uses a system of tags to surround content and indicate what it is and how it should be displayed.

Video: Intro to Web Lab - Part 2 

[Student Overview](#)

HTML Tags 

[Student Overview](#)

Using HTML Tags

[7](#)

[8](#)

[9](#)

(click tabs to see student view)

Wrap Up (5 minutes)

Reflection

Prompt: In your own words, how does HTML help solve the problem of telling a computer what a web page looks like, not just what content is on it?

Discuss: Have students write or silently think out their ideas, then share with a partner, then share with the class.

Vocabulary: Introduce the following words [Website Content](#) [Website Structure](#)

Remarks

HTML uses tags to help the computer know what different pieces of content in the web page actually are. Right now we've only learned how to tell the computer that some text is a paragraph, or that part of your website is the body. We've already seen how that affects the way our web pages look and are structured. As we move forward we're going to learn more tags and see more examples of how this language helps us add structure to our webpages.

Review: Return to the list of lesson resources you wrote on the board and review as a class how they are supposed to be used. Refer to the teaching tip above for recommended uses.

Discussion Goal

Goal: Students answers will vary but will likely center around the fact that using tags helps the computer know what different pieces of content "are". Using these tags helps the computer know what the tags are supposed to look like. If this discussion needs to be returned to after students have seen more tags that's fine as well. In either case use this discussion to motivate the content vs. structure wrap-up point.

Content Corner

Content - Structure - Style: Once students have been through this lesson the definitions provided here should have some context. The content is the literal words that are being typed on the page. Using HTML students are providing structure to the page, explaining how those pieces of content should be interpreted. Later in the unit students will learn CSS, a language that allows them to individually style elements. For now, however, the styles being applied based on their HTML tags are just the default styles of their web browser. Students don't need to fully understand this difference at this point, as it will be much clearer once they learn CSS in chapter 2. The difference between content and structure, however, will be returned to many times in this chapter.

Standards Alignment

View full course alignment

CSTA K-12 Computer Science Standards

► **AP** - Algorithms & Programming



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

If you are interested in licensing Code.org materials for commercial purposes, **contact us**.