

Lesson 3: Headings and Lists

45 minutes

Overview

This lesson introduces the core practices of pair programming and debugging. In this lesson, students continue to use HTML to structure text on web pages, this time in pairs, with a focus on working together and debugging problems with their sites. Students learn how to use the different heading and list elements and practice using them to give their web pages more structure.

Question of the Day: How can we work together to fix problems with our websites?

Standards

Full Course Alignment

CSTA K-12 Computer Science Standards (2017)

► **AP** - Algorithms & Programming

Agenda

Warm Up (5 minutes)

Tags Poster

Activity (35 minutes)

Web Lab: Headings

Web Lab: Lists

Wrap Up (5 minutes)

Journal

Objectives

Students will be able to:

- Structure content into headings, subheadings, lists and paragraphs.
- Use a structured practice to collaboratively create a digital artifact.
- Use heading and list tags to change the appearance of text on a web page.

Preparation

- Have student journals ready to give back.
- If you want to use an anchor chart, prepare poster paper to do so as a whole class.
- Check the "**Teacher's Lounge**" forum for verified teachers to find additional strategies or resources shared by fellow teachers
- If you are teaching virtually, consider checking our **Virtual Lesson Modifications**

Links

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

For the teachers

- **Formatting HTML** - Resource
- **Headings and Paragraphs** - Resource
- **Lists** - Resource

For the students

- **Video: Debugging** - Video (**Download**)

- [Video: Pair Programming](#) - Video
([Download](#))

Vocabulary

- **Heading** - A title or summary for a document or section of a document.
- **List** - allows web developers to group a set of related items

Introduced Code

- `<h1></h1>`
- `<li></li>`
- `<ol></ol>`
- `<ul></ul>`

Teaching Guide

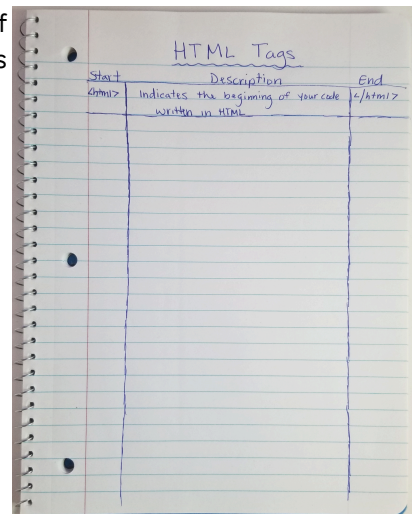
Warm Up (5 minutes)

Tags Poster

Journal: Have students make a three-column chart on a blank page in their journals and label the top "HTML Tags"

Teaching Tip

Anchor Chart / Poster: Throughout this unit students will be keeping track of the tags they learn. This warm-up prompts students to record the HTML tags that they learn by writing them in their journals. If you wish, you can also keep track of the same information on a shared class poster or anchor chart that you update after each lesson. Prompts throughout the unit will tell you when students should update their journals, and updating the anchor chart may provide reinforcement of that process.



Start	Description	End
<code><html></code>	Indicates the beginning of your code written in HTML	<code></html></code>

Prompt: Yesterday, you learned about HTML, the language of the World Wide Web. HTML uses tags to structure content on web pages. Individually, think of as many tags as you remember and what they do.

Give students a few minutes to think of as many tags as they can.

 **Remarks**

Now that you've had some time to think of your own, share your lists with a partner and see whether there's anything else that you can add.

Discuss: Pairs should discuss and share with one another the HTML tags they can remember and start record their ideas on the HTML Tag Chart.

Discussion Goal: The goal of this discussion is to review the tags that students saw in the previous lesson. These include:

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

Afterward, ask students to share the tags they came up with along with each tag's description.

 **Display:** Use the slide with the HTML Tag Chart table to help students ensure they documented all tags needed at this point along with their descriptions.

`<!DOCTYPE html>` - Tells the computer that this is a document written in HTML - No closing tag

`<html>` - Indicates the beginning of your code written in HTML - `</html>`

`<head>` - Contains information (sometimes called "metadata") about your web page - `</head>`

`<body>` - Contains all the main content of your web page - `</body>`

`<p>` - Defines a paragraph - `</p>`

As you go through the tags, highlight how working together allowed the students to make their lists more comprehensive.

Teaching Tip

If you choose to make a poster or anchor chart, you can use this time to update the chart at this point.

Remarks

Usually we are able to solve problems better when we work with someone else. That's true in programming our web sites, too. Today, we're going to look at some ways that we can work together to solve different problems that our websites might have.

Question of the Day: How can we work together to fix problems with our websites?

Activity (35 minutes)

Teaching Tip

Guide to Programming Levels: Additional guidance for programming levels is provided in the [CSD Guide to Programming Levels](#). This document includes strategies and best-practices for facilitating programming levels with students.

Web Lab: Headings

Group: Group students into pairs

Transition: Have pairs go to Code Studio and both log in using the "Pair Programming" feature, just like yesterday

Do This: Remind students to switch driver and navigator every three minutes. You may want to project a digital timer at the front of the room.

 1

Exploration

Circulate: Give students time to explore the headings and discuss the questions on the level with their partner.

Prompt: When would you use each of these headings? For example, which heading would you use for a web page title ... when might you use the H6 heading?

Discussion Goal: Students should notice the H1 is the largest heading tag and should therefore be used for web page titles. The remaining heading tags are used for sub-titles and section/subsection titles.

Code Studio: Have students continue to the next level in Code Studio

 2-3

Skill Building

2

3

 4



Quick Check

Teaching Tip



[pull-right]



[/pull-right]

Level Types: This multiple choice question that asks students to choose how HTML headings will display on a web page. You can see which students have answered the question correctly by going to that level and pulling out the Teacher Panel from the right hand side. Students who have successfully answered the question will have a green bubble. Students have as many chances as they like to answer correctly, so guessing and checking can also lead to a correct answer. [Click Here to learn more about using the teacher panel with students.](#)

Assessment Opportunity



This level can be used to assess student understanding of using heading tags to change the appearance of text on a web page.

Transition: Have your students pause programming and bring the class back together at this point in order to add heading tags to their HTML Tag Chart.

 **Prompt:** What are the different heading tags? Which one would we use for web page titles? What are the others for?

Discussion Goal: Students should be able to identify six different heading tags, from `<h1>` to `<h6>`. In addition, students should understand that the `<h1>` tag is the largest heading tag, typically used for the main web page title while the others are used for subtitles throughout a web page.

 **Display:** Use the slides and the remarks below to help students add heading tags to their HTML Tag Chart.

`<h1>` - Largest heading tag, typically used for web page titles - `</h1>`

`<h2>` to `<h6>` - Heading tags, typically used for subtitles - `</h2>` to `</h6>`

Remarks

 It is important to remember and get down in our HTML chart that heading tags, `<h1>` through `<h6>` are used any time we have a title or a subtitle. `<h1>` is normally used for the webpage title, while `<h2>` to `<h6>` are used for subtitles on webpages.

Web Lab: Lists

 **Prompt:** When might someone use a list?

- *Think about times when you have made lists ... Why did you make the list? What was the list for? What type of stuff was in your list(s)?*

Keep track of the different types of lists students identify.

Discussion Goal: There are many possible answers. Examples could include a **bulleted** list used to make a grocery list and a **numbered** list used to list your favorite movies or the steps in a recipe.

 **Display:** Show the first example of an HTML list and the result. Ask the students "What do you notice? What do you wonder?"

Discussion Goals: Student responses to the first prompt, *What do you notice*, may vary but could include:

- There are two new tags - `<ul>` and `<li>`
- The result is a bulleted list
- The closing `<ul>` tag comes after all of the `<li>` tags
- The `<li>` tags are indented

Student responses to the second prompt, *What do you wonder*, may also vary but could include:

- What do "ul" and "li" mean?
- Why are there multiple `<li>` elements?
- Why does the `<ul>` tag not seem to do anything?
- Why are the `<li>` tags indented?

It is important to note that this prompt is purposefully open-ended which could result in unexpected student responses, which is okay.

Do This: Move on to the second example before answering any questions. Allow their *What do you wonder* questions to linger.

 **Display:** Show the second example of an HTML list and the result. Again, ask the students *What do you notice? What do you wonder?*

Discussion Goals: Student responses for the first prompt, *What do you notice*, may again vary but could include:

- The `<li>` tag is being used again
- `<ol>` is being used instead of `<ul>`
- This time a numbered list was made
- There is no content next to the starting `<ol>` tag, similar to the `<ul>` start tag in the first example
 - This last observation is important to draw out of your students. You may consider sharing with students that it is something you noticed if there are no students who volunteer that information.

Student responses to the second prompt, *What do you wonder*, for the second example may again vary but could include:

- What does "ol" mean?
- Why did this example make a numbered list?
- Why are both examples using the `<li>` tag?
- Why does making a list involve two different HTML tags?

Discussion Goal: Use student responses from both rounds of *What do you notice?* *What do you wonder?* to guide your discussion, making sure to cover the following key points:

- To make an HTML list, they need both tags - the first tag indicating the **type** of list they want and the second tag indicating a list item.
- The structure of the HTML list element is unlike other HTML elements they have used so far since there is no content that goes right next to the start `<ul>` or `<ol>` tag.
- We use indentation, or *whitespace*, in front of the `<li>` tags to indicate that they are "children" nested within the type of list tag.
- Once they have finished listing their items using the `<li>` tag, that is when they will finally use the closing `</ul>` or `</ol>` tag.
- If they want their list to have a title, they need to use one of the heading tags they just learned about.

 **Display:** Use the slide to help students add list tags to their HTML Tag Chart.

`<ol>` - Starts an Ordered List - `</ol>`

`<ul>` - Starts an Unordered List - `</ul>`

`<li>` - Actually creates the list item - `</li>`

Have students continue Pair Programming through the List Skill Building levels.



5-7

Skill Building

5

6

7

 **Transition:** Have your students pause programming and bring the class back together at this point in order to watch the **Video: Debugging** video together.

Questions to consider with the video:

- What is debugging?
- What are the four steps to debugging?

Discussion Goal: Students should identify debugging as the process of finding and fixing problems in their code. The four steps of the debugging process are describing the bug, hunting for the bug, trying out small solutions (or changing your code), and documenting what you have learned.

Normalizing Mistakes and Supporting Debugging: As programming levels become more complex, students may find themselves with bugs in their code that they need to untangle. If this happens frequently, this can be a demoralizing experience for students and can affect their self-perception of how capable they are in class.

To counter this, we recommend normalizing bugs and mistakes as something that happens to everyone - it's just part of the process. Additionally, consider displaying the **Student Guide to Debugging** for students to reference throughout the unit and having **Bug Report Quarter-Sheets** available for students to use.

💡 Teaching Tip ▲

To encourage active engagement and reflection, use one or more of the strategies discussed in the **Guide to Curriculum Videos**.

Do This: Have students continue Pair Programming through the remainder of the levels.

💻 8

Practice

💻 9-10

Assessment



✔ Assessment Opportunity ▲

Formative Assessment: Levels 9 & 10 can be used as a formative assessment.

A rubric is provided in level 9, and written feedback can be given to students. **[Click here to learn more about giving feedback to students.](#)**

Level 10 is a free response reflection that can be used as an opportunity to gain insight into how the students completed the assessment level and the choices they made with the HTML tags. In addition, this reflection can be used to identify any misconceptions shared among students that should be cleared up.

👥 11

Challenges

Wrap Up (5 minutes)

Journal

Question of the Day: How can we work together to fix problems with our websites?

Key Vocabulary:

- **heading** - a title or summary for a document or section of a document
- **list** - allows web developers to group a set of related items

📋 Prompt: Today, you learned a lot about debugging, which is an important skill for programmers. What is one way working with a partner helped you to debug today?

Check that students are describing effective forms of collaboration.



This work 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**.