UNIT
 Ch. 1
 1
 2
 3
 4
 5
 6
 7
 8
 9
 Ch. 2
 10
 11
 12
 C
 O

 2
 13
 14
 D
 E

## **Unit 2 - Web Development**

In this unit, students are empowered to create and share the content on their own web pages. They begin by thinking about the role of the web, and how it can be used as a medium for creative expression. As students develop their pages and begin to see themselves as programmers, they are encouraged think critically about the impact of sharing information online and how to be more critical content consumers. They are also introduced to problem solving as it relates to programming, as they learn valuable skills such as debugging, commenting, and structure of language. At the conclusion of the unit, students compile their work to create a personal website they can publish and share.

## **Chapter 1: Web Content and HTML**

#### **Big Questions**

- Why do people create websites?
- How can text communicate content and structure on a web page?
- How can I incorporate content I find online into my own webpage?
- What strategies can I use when coding to find and fix issues?

## Week 1

## Lesson 1: Exploring Websites

#### Unplugged

This lesson covers the purposes that a website might serve, both for the users and the creators. The class explores a handful of the most-used websites in the United States and discusses how each of those sites is useful for users and how it might also serve its creators.

## Lesson 2: Websites for Expression

#### Unplugged

This lesson introduces websites as a means of personal expression. The class first discusses different ways that people express and share their interests and ideas, then looks at a few exemplar websites made by students from a previous course. Finally everyone brainstorms and shares a list of topics and interests to include, creating a resource for developing a personal website in the rest of the unit.

## Lesson 3: Intro to HTML

#### Web Lab

This lesson introduces to 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. The class looks at an HTML page in Web Lab and discusses how HTML tags help solve this problem, then uses HTML to write the first web pages of the unit.

## Week 2

## Lesson 4: Headings

Web Lab

This lesson continues the introduction to HTML tags, this time with headers. The class practices using header tags to create page and section titles and learns how the different header elements are displayed by default. Next, the class plans how to organize their content on the personal web pages that will be built across the unit and begins the first page of the project.

## Lesson 5: Digital Footprint

#### Unplugged

This lesson takes a step back from creating the personal website to talk about personal information people choose to share digitally. The class begins by discussing what types of information are good to share with other people, then looks at several sample social media pages to see what types of personal information could be shared intentionally or unintentionally. Finally, the class comes up with a set of guidelines to follow when putting information online.

## Week 3

## Lesson 6: Lists

#### Web Lab

This lesson introduces ordered and unordered lists and the associated <ul&gt;, <ol&gt;, and &lt;li&gt; HTML tags. The class practices using the tags, then goes back to the personal web page project to add a new HTML page that includes the new tags.

## **Lesson 7: Intellectual Property and Images**

#### Unplugged | Web Lab

This lesson covers how to use media such as images, video, or music created by others a website while respecting the rights of the creator of that media. After first studying Creative Commons licensing, the class learns how to add images to web pages, and how to give proper attribution when doing so.

## Lesson 8: Clean Code and Debugging

#### Web Lab

In this lesson covers common issues that arise when designing web pages in HTML. The class will correct errors in a sequence of increasingly complex web pages found on Code Studio and learn the importance of comments, whitespace, and indentation as tools for making web pages easier to read.

## Week 4

## Lesson 9: Project - Multi-Page Websites

Web Lab | Project

This lesson covers hyperlinks, which allow web developers to connect pages together into one website. The class will link together all the previous pages into one project, and create navigation bars for each page before publishing the entire site to the Web.

## **Chapter Commentary**

Students use computing as a form of self expression as they design and develop basic web pages. Focusing on the tags, keywords, and syntax used to communicate instructions to the computer, students use HTML to structure the content of a web page. They also explore the privacy and intellectual property implications of publishing their work online.

## **Chapter 2: Styling and CSS**

#### **Big Questions**

- How do I modify the appearance and style of my web pages?
- How do I safely and appropriately make use of the content published on the Internet?

## Week 5

## Lesson 10: Styling Text with CSS

#### Web Lab

This lesson introduces CSS as a way to style elements on the page. The class learns the basic syntax for CSS rule-sets and then explores properties that impact HTML text elements. Finally, everyone applies text styles to their personal websites.

## Lesson 11: Styling Elements with CSS

#### Web Lab

This lesson continues the introduction to CSS style properties, this time focusing more on non-text elements. The class begins by investigating and modifying the new CSS styles on a Desserts of the World page. Afterwards, everyone applies this new knowledge to their personal websites.

## **Lesson 12: Sources and Search Engines**

After first completing a web search scavenger hunt, the class learns about the inner workings of search engines and has an opportunity to flex their analytical skills in a search for strange and unlikely animals.

## Lesson 13: RGB Colors and Classes

Web Lab

This lesson covers classes and custom colors. The class first learns how to specify custom colors using RGB (red, green, blue) values, then applies these colors to a new Four Seasons web page, which uses CSS classes. Using classes, the class adds more styles to the Four Seasons web page, then use them to style their personal websites.

## Week 6

## Lesson 14: Project - Personal Portfolio Website

Web Lab | Project

In the last few days of the unit, the class finalizes their personal websites, working with peers to get feedback, review the rubric, and put the finishing touches on the site. To cap off the unit, everyone shares their projects and how they were developed.

## **Chapter Commentary**

After covering the basics of HTML, students dive into improving their websites with CSS. They learn to use colors, fonts and margins to create a unique style of their own design. At the end of the chapter students publish a personal portfolio website that demonstrates use of HTML for content and CSS for a personalized.



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

# Ch. 1 1 2 3 4 5 6 7 8 9 Ch. 2 10 11 13 14



## **Lesson 1: Exploring Websites**

## Overview

Every website has a purpose, a reason someone created it and others use it. In this lesson, students will start to consider the purposes a website might serve, both for the users and the creators. Students will explore a handful of the most-used websites in the United States and try to figure out how each of those sites is useful for users and how they might also serve their creators.

## Purpose

Unit 2 is all about the World Wide Web, with a focus on creating websites for self expression. To start students thinking about why they might want to create a website, they must begin to consider how websites are useful tools for both users and creators. The lesson starts with popular websites because they all serve clear purposes for a large number of users, but eventually students are asked to narrow their vision to simpler sites that might serve the needs of a smaller user group.

## Agenda

Warm Up (10 minutes) The Last Website you Visited Activity (45 minutes) Websites We Use Frequently

Finding the Purpose Wrap Up (5 min)

Web Development Goals

# View on Code Studio Objectives

Students will be able to:

- Identify the reasons someone might visit a given website
- Identify the reasons someone might create a given website

## Preparation

Print a copy of The Purpose of WebsitesActivity Guide for each student.

## Links

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

For the Teachers

• The Purpose of Websites - Exemplar

For the Students

• The Purpose of Websites - Activity Guide Make a Copy -

## Vocabulary

• Website - A collection of interlinked web pages on the World Wide Web

## Warm Up (10 minutes)

#### The Last Website you Visited

**Set Up:** Have students take out their journal. In their journal have students answer the following prompt.

Ŷ

**Prompt:** What was the last website you visited? Why did you go to that website?

Q

**Discuss:** Have students share out the last website they visited. Make sure students share why they went to that website and what information they were looking for. In addition you can ask students whether or not the website provided the information they wanted.

## Activity (45 minutes)

#### 😵 Teaching Tip

**Websites vs. Apps:** You may find that students don't *think* that they actually visit websites very frequently, because they use apps instead. If you encounter this, encourage students to consider whether those apps might actually just a streamlined way to visit a website. Facebook, for example, might be used more frequently from an app, but it's actually a website underneath it all.

#### 🥺 Discussion Goal

**Goal:** The goal of having this discussion is to pull out that websites share information and that people go to websites for a particular purpose. You only stay on that website if it fulfills your needs.

#### Websites We Use Frequently

**Prompt:** With a neighbor, come up with a list of what you think are the top 5 most frequently visited websites. These can include websites that you visit often, but also consider which pages the other people in your life use frequently.

Share: Have groups share out their top 5 websites lists, keeping track of unique sites on the board.

#### Q

**Discuss:** Why do you think these websites are so commonly used? What purpose do they serve, or what problems do they solve?

#### Finding the Purpose

#### 🎍 Remarks

We came up with a lot of reasons why people might use some of the most visited websites, but that's

only one side of the picture. The people who make websites also have their own reasons for doing so. In the next activity, you're going to start thinking about the purposes that various websites serve for both the user and the creator.

Group: Put students in pairs.

Distribute: Pass out a copy of The Purpose of Websites - Activity Guide to each student.

Direct students to the "Top Websites" level on Code Studio.

## 🖵 Code Studio levels

- Top Websites 🖺
- Student Overview

#### Discussion Goal

**Goal:** The intent of this discussion is to get students thinking about both why people make websites, and why they use websites. Push students to give detailed purposes for some of the websites that the class generated.

## **Top Websites in the US**

The following are the most frequently visited websites in the United States. Each includes a brief description of the website's core function.

- 1. Google.com Search engine for websites, pictures, videos, and other media
- 2. Youtube.com User-submitted video site with ratings and comments
- 3. Facebook.com Social media community that allows users to share information with friends
- 4. Amazon.com Online shopping sites for all kinds of goods
- 5. Yahoo.com A search engine, email provider, and news source
- 6. Wikipedia.org A free encyclopedia with content generated by users

#### **The Purpose of Websites**

#### Ŷ

#### Top Websites

This first page of this activity guide asks students to consider what purposes some of the most visited websites might serve. Using a list of the top ten websites in the US, student pairs select three sites that interest them and for each discuss:

- Why people might use that site (what purpose does it serve users?)
- Why the creators might have made that site (what purpose does it serve the creators?)

Encourage students to be as detailed as possible when brainstorming the purposes of these sites.

**Share:** Once everyone has had a chance to brainstorm about their sites, give the class a chance to share out their thoughts.

#### ♀ Teaching Tip

The goal of this activity to get students to consider why different websites were created. They don't need to necessarily get it "right" for each site, but they should demonstrate that they have considered the potential purposes that a website might serve for the creator. If students struggle to identify the purpose a site serves its creator, consider asking the following prompts:

- Does the web site sell goods or services?
- Do users contribute anything (images, information, reviews) that might be useful to the creator?
- Does the site include any paid advertising?

#### Personal Websites

The second page of this activity guide encourages students to transition from thinking about huge sites that serve the needs of millions of people, to the much smaller kinds of sites they could create, which can still effectively serve the needs of a narrower group of users.

Read through the personal website description with the class:

**Website Description:** Julia wants to become a better home cook, so she started a blog where she can post about the recipes she tries. Each week Julia attempts a new recipe and adds a page about it to her blog. Sometimes the recipes go really well and her blog readers leave her encouraging comments. Occasionally her recipe attempts don't go so well, but she still posts and usually gets some helpful advice from her readers. After starting to blog about the meals she makes, Julia has started to cook more often and is attempting dishes that she never would have tried before.

**Discuss:** Give students an opportunity to ask questions about this example site. Make sure they understand how a small site that is primarily a tool for individual self expression can also be a useful site for other people.

The final questions on the activity guide ask students to come up with potential purposes this website serves both for its creator and its users.

Share: Ask for volunteers to share the purposes they came up with.

## Wrap Up (5 min)

Web Development Goals

**Journal:** At this point we've just scratched the surface of what websites really are, and we haven't even begun to create websites of our own. In your journal, come up with three goals for yourself this unit. These could be related to improving on one of our class practices, learning how to make a specific kind of web site, or creating a site that serves some purpose for you or others.

## **Standards Alignment**

CSTA K-12 Computer Science Standards (2017)

► IC - Impacts of Computing



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

## **Lesson 2: Websites for Expression**

Ch. 1 (1) (2) (3) (4) (5) (6) (7) (8) (9) Ch. 2

## Overview

In this lesson students investigate ways to use websites as a means of personal expression and develop a list of topics and interests that they would want to include on a personal website. To begin the lesson students brainstorm different ways that people express and share their interests and ideas. Students then look at a few exemplar websites made by students from a previous CS Discoveries course to identify ways they are expressing their ideas. Finally students brainstorm and share a list of topics and interests they might want to include on a personal website which they can reference for ideas as they progress through the unit.

## Purpose

This lesson introduces the central story of Unit 2, namely that web development, and by extension computer science, is an avenue for self expression. The warm up situates web development as another means of self-expression alongside visual art, dance, music, fashion, social media, and many other ways students are likely already expressing themselves. Seeing student exemplars helps scope students expectations for their skills by the end of the unit. Finally the brainstorm of content for their website provides students an opportunity to consider what messages they want to express as they build their websites.

## Agenda

Warm Up Express Yourself Activity Personal Website Planning Wrap Up (5 mins)

# View on Code Studio Objectives

Students will be able to:

 Identify websites as a form of personal expression

## Preparation

Prepare journals or optionally print copies of Personal Website Sketch -Activity Guide

## Links

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

#### For the Teachers

• Personal Website Sketch - Exemplar

#### For the Students

• Personal Website Sketch - Activity Guide

## Vocabulary

• Website Content - the text and images on a website

## Warm Up

#### **Express Yourself**

**Prompt:** What are the ways that you or your friends express yourselves? Think about different ways of communicating, activities, or spaces that you consider a form of self-expression.

#### Q

**Discuss:** Students should brainstorm ideas independently, then share with a neighbor, and finally share with the class at large.

#### 🎍 Remarks

We share our thoughts, feelings, and ideas in many different ways. Some people might express their thoughts online, others might do so in the way they dress or by making a song or drawing a picture. Self-expression is a really important part of our lives, and as we're going to see, making websites is another way we can express our ideas, interests, and feelings.

## Activity

#### Personal Website Planning

#### Personal Website Planning Guide

**Distribute:** Have students open a journal or optionally distribute copies of **Personal Website Sketch - Activity Guide** 

Ŷ

**Brainstorm Content:** Read through this section of the activity guide. Then give students several minutes to silently brainstorm content they might want to include in a personal website.

**Share:** Have students share their content ideas with a neighbor.

#### 🎐 Remarks

When sharing ideas with other people an important thing to consider isn't just what you want to say, but how you want to say it. You just created a list

#### Discussion Goal

**Goal:** This warm-up should generate a list of ways students express their ideas. If they need help getting started offer them a couple examples from the list below. The goal of this warm-up is to frame web development as another form of self-expression in this list.

Draw a picture, write a song, build something, dance, clothing and fashion, write and share something, talking with friends, posting something online

#### **?** Teaching Tip

**Use Journals:** The activity for today's lesson can easily be completed in student journals. Project or otherwise share the prompts and let students complete their work in their journals.

	Content Brainstorm
	*Valleyball *Valleyball * Parking and A dong my Dures * Coeffing and A dong my Dures
	Website Sketch
199999	The sent sent to be the wide wind Rules History Proto
66666	
9999	

of ideas you might want to share on your personal websites. Let's go look at some personal websites other students have made and think about not only what content they're sharing, but how they are presenting that message.

**View Personal Websites:** Send students to Code.org and have them look through the exemplar websites in pairs. Use these websites as inspiration for their own personal websites. They were created in Web Lab by other CS Discoveries students.

🖵 Code Studio levels

#### Exemplar Personal Websites 🗎

**Student Overview** 

#### Ŷ

**Website Sketch:** Give students some time to sketch one page from their website. Remind students that they should be thinking of ways to effectively share the content on that page. Use the student exemplars students looked at to help scope the complexity of web pages they should be considering.

## Wrap Up (5 mins)

**Share:** Give students a chance to share their website sketches with a neighbor.

#### 😵 Teaching Tip

Limit Time: Sketching a website can easily grow to take an entire class period if student expectations aren't scoped. Explain to students that they'll have plenty of time to refine their designs once they've developed more programming skills. Roughly 10 minutes should be plenty to get a reasonable sketch. Emphasize that their topic lists is much more important at this point but both will be updated throughout the unit.

#### Remarks

This unit we'll be working to bring many of your ideas to reality. Along the way you're going to learn some new computer languages and explore the way the websites you use are created. The sketches you made or even the content you want to share might change as we go through the unit, but this goal of using websites for self-expression will be there throughout.

**Collect:** If students did not put their website sketches in a journal collect them. They will explicitly be referenced again in Lesson 4.

## **Standards Alignment**

CSTA K-12 Computer Science Standards (2017)

- ► AP Algorithms & Programming
- ▶ IC Impacts of Computing



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



## Lesson 3: Intro to HTML

## 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

# View on Code Studio 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 Teachers

• Exemplar Text Website - 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 text and images on a website
- Website Structure how the content of a website is organized

## Introduced Code

- •
- <html></html>
- <head></head></head>
- <!DOCTYPE>
- <body></body></body>

## Warm Up (10 mins)

#### The Need for HTML

Display: Show the image inside the Exemplar Text Website - 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.

#### Q

**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.

#### 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.

- 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.

#### Ŷ

**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.

#### 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) Video: Intro to Web Lab - Part 1 Student Overview Explore HTML Teacher Overview Student Overview View on Code Studio C 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.



## Wrap Up (5 minutes)

Reflection

P

**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?

#### Q

**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 <u>Website</u> <u>Content</u> <u>Website Structure</u>

#### 🖢 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.

## **Standards Alignment**

CSTA K-12 Computer Science Standards (2017)

► AP - Algorithms & Programming

#### 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 stucture 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.



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

# $\begin{array}{c} \begin{array}{c} \text{UNIT} \\ \begin{array}{c} \text{Ch. 1} \\ 1 \end{array} \end{array} \begin{array}{c} 1 \end{array} \begin{array}{c} 2 \end{array} \begin{array}{c} 3 \end{array} \begin{array}{c} 4 \end{array} \begin{array}{c} 5 \end{array} \begin{array}{c} 6 \end{array} \begin{array}{c} 7 \end{array} \begin{array}{c} 8 \end{array} \begin{array}{c} 9 \end{array} \begin{array}{c} \text{Ch. 2} \end{array} \begin{array}{c} 10 \end{array} \begin{array}{c} 11 \end{array} \begin{array}{c} 12 \end{array} \end{array}$



## **Lesson 4: Headings**

## Overview

In this lesson, students continue to use HTML to structure text on web pages, this time with headings. Students learn how the different heading elements are displayed by default and practice using them to create page and section titles. Students then start to decide how they will organize their content on their own personal web pages. In the last level, students begin the project that they will continue to work on throughout the unit.

## Purpose

This lesson introduces the heading tags that students will use in their pages for the rest of the unit. It also reinforces the general structure of HTML (opening tag / content / closing tag) so that students are ready to begin adding their own content in the project.

The personal web page that students work on at the end of the lesson will follow them throughout the unit. As they learn more HTML and CSS, they will improve the page, adding images, colors, and different fonts. This lesson is a chance for them to start the page, knowing that they will continue to improve it over the next few weeks.

## Agenda

Warm Up (5 minutes) Tags Poster Activity (45 minutes) Pair Programming Web Lab: Headings Wrap Up (5 minutes) Journal

# View on Code Studio Objectives

Students will be able to:

- Use heading tags to change the appearance of text on a web page.
- Structure content into headings, subheadings, and paragraphs.

## Preparation

 Have student journals (or project sketches) ready to give back.
 If you will not use journals to track HTML tags, prepare poster paper to do so as a whole class

## Vocabulary

• Heading - A title or summary for a document or section of a document.

## Introduced Code

• <h1></h1>

## Warm Up (5 minutes)

#### Tags Poster

#### Ŷ

**Group:** Put students into groups of three to four students.

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

**Prompt:** Yesterday, you learned about HTML, the language of the World Wide Web. HTML uses tags to structure content on web pages. In your groups, 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.

#### Q

**Discuss:** Groups should discuss with one another and record their ideas in their journals (or alternately prepare to share on the shared class poster). Afterwards quickly share across the room to make sure all groups have listed all tags.

#### 🎍 Remarks

As we learn more HTML tags, we'll be keeping track of them so we'll have a reference as we make our web pages.

## Activity (45 minutes)

#### Pair Programming

Group: students into pairs.

#### Ŷ

#### 🎍 Remarks

We're going to be working on Web Lab again today, but this time we'll be using **pair programming**. Pair progamming helps people make better programs by working together, but there are some rules we have to follow to make sure it goes well.

Display: Show the Pair Programming video.

**Review:** Ensure that students understand the rules for pair programming:

#### • There is only one computer.

- The driver is the only one to touch the keyboard/mouse.
- The navigator should look for problems in the code and keep track of the high-level plan.
- Both driver and navigator should be communicating constantly.
- Driver and navigator must switch when the teacher indicates, typically every couple minutes.

#### **?** Teaching Tip

#### Journal or Poster?

Throughout this unit students will be keeping track of the tags they learn. This warm up descibes one option for students to record the HTML tags that they learn by writing them in their journals. If you wish you can keep track of the same information on a shared class poster that you update after each lesson. Prompts throughout the unit



will tell you when students should update their journals or the class poster.

#### 喿 Discussion Goal

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

- <!DOCTYPE html> Tells the computer that this is a document written in HTML
- <html> Indicates the beginning of your code written in HTML
- <head> Contains information (sometimes called "metadata") about your web page
- <body> Contains all the main content of your web page
- Defines a paragraph

#### **§** Teaching Tip

**Pair Programming:** Pair programming is practiced in education and in industry. Students who pair program are more confident in their abilities and are more likely to continue to study computer science. This practice is most effective if it is introduced early and the rules for switching partners are enforced by the teacher. You can read more about Pair Programming in the **Computer Science Discoveries Curriculum Guide** 

#### Web Lab: Headings

Transition: Have pairs go to Code Studio and both log in using the "Pair Programming" feature.

**Prompt:** students to switch driver and navigator every three minutes.



Prompt: Have students update their "HTML Tags" log with the heading tags they learned in this lesson.

**Prompt:** Have students reflect on their development of the **Five Practices of CS Discoveries** (Problem Solving, Persistence, Creativity, Collaboration, Communication). Choose one of the following prompts as you deem appropriate.

- Choose one of the five practices in which you believe you demonstrated growth in this lesson. Write something you did that exemplified this practice.
- Choose one practice you think you can continue to grow in. What's one thing you'd like to do better?

• Choose one practice you thought was especially important for the activity we completed today. What made it so important?

## **Standards Alignment**

CSTA K-12 Computer Science Standards (2017)

► AP - Algorithms & Programming



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



## **Lesson 5: Digital Footprint**

## Overview

As students have recently spent some time thinking about the actual content that will go into their personal website, this lesson takes a step back from the unit-long project (publication of a personal website) to help students articulate what personal information they choose to share digitally and with whom. It also reinforces the notion that much of the information that they choose to share digitally falls largely out of their control the moment it is released.

Students begin by individually identifying appropriate audiences with whom they would be comfortable sharing given pieces of personal information. They then look at several social media pages to determine what sorts of information people are sharing about themselves or one another. Last, students reflect on what guidelines they think are appropriate for posting information online.

The ultimate point of this lesson is not to scare students, but rather to experientially bring students to realizing precisely what level of control they don't have in releasing information into the web.

## Purpose

Now that students are beginning to share information publicly, it's crucial that we instill in them an understanding of the potential consequences of sharing personal information online.

The first activity engages students in considering specific pieces of personal information and the audiences with whom they would or would not want that information to remain private. The goal of this activity is for students to understand that for any given piece of personal information if there is *any* audience they would like to maintain privacy with they should consider not posting that information online.

In a follow up activity students look through several example social media profiles for fake students. Individually these pages contain relatively innocuous information, but when students cross reference information across multiple profiles they see how a detailed picture (or <u>digital footprint</u>) can start to form. Through this activity students see that even information that they may not see as private on its own can share more information than intended when combined with other small details.

## Agenda

Warm Up (10 min) Your Digital Footprint

# View on Code Studio Objectives

Students will be able to:

- Understand and explain reasons that it is difficult to control who sees information published online.
- Understand and justify guidelines for safely publishing information online.

## Preparation

Print out copies of When Does YourPrivacy Matter? - Activity Guide for each student

Print out copies of Social Sleuth -Activity Guide for each student (or one for each group if grouping)

## Links

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

For the Teachers

- When Does Your Privacy Matter? Exemplar
- Social Sleuth Exemplar

#### For the Students

- When Does Your Privacy Matter? Activity Guide Make a Copy -
- Social Sleuth Activity Guide Make a Copy -

## Vocabulary

• **Digital Footprint** - The collected information about an individual across multiple websites on the Internet.

#### Activity (30-40 min)

How Much Do You Care About Privacy? Investigating Social Profiles

#### Wrap Up (5 minutes)

Shaping Your Digital Footprint

## Warm Up (10 min)

#### Your Digital Footprint

**Prompt:** Ask the students to consider what websites they have given personal information to. In small groups, ask students to come up with a list of websites that they have accounts on (even if they use sign in with Facebook, Google, or similar).

**Discuss:** Have students silently brainstorm or journal, then share with a partner, and finally share as a full class. Create a comprehensive list of all of the websites that may have their personal information.

**Discuss:** Ask the class about what kind of information they think these websites might have about them. You can frame it from a few different angles:

#### Discussion Goal

**Goal:** This discussion is not about coming up with a comprehensive list of personal information students may have shared with websites, nor is it to frighten them out of sharing information. The goal is for students to start thinking more critically about when and where they share information about themselves.

This will prime them for the main activity of the day, looking at specific pieces of information and audiences to consider when their privacy matters, and whether they are unknowingly giving up their privacy.

- What information do you know you've given to a website (eg. your email address)
- What information might you have unknowingly given to a website (eg. a picture with your home address)
- What information might other people have shared about you without your knowledge or permission (eg. tagging you on Facebook)

## Activity (30-40 min)

#### How Much Do You Care About Privacy?

Distribute: When Does Your Privacy Matter? - Activity Guide

#### **When Does Your Privacy Matter?**

**Step 1:** Individually, have students complete the grid on the front side of the activity guide. Encourage them to think critically about the consequences of each audience having access to each piece of personal information. What might seem like an innocuous piece of information might have far reaching consequences depending on who has access to it.

Q

•

**Step 2:** Once students have completed the privacy grid, have them come together into pairs or small groups. They should look for cells in each grid where their partner marked a different answer than they did and discuss their choices. Encourage students to respectfully debate when they disagree on an issue. While there are no right or wrong answers here, it's useful for students to hear how others may have considered consequences (both positive and negative) that they didn't think of.

**Step 3:** After the small group discussion, students should complete the reflection questions on the next page of the activity guide.

#### 오 Discussion Goal

In the pair discussion on the Activity Guide, the goal is not to focus on what the actual private information is (like who happens to be "the favorite teacher" or the "secret crush" of a student), but rather about identifying commonalities and differences in what students deem appropriate to share with different audiences.

Students may be tempted to think about specific examples that they would or would not be comfortable with others knowing, but you want to push them towards thinking more generally of the types of information and audiences presented. **Prompt:** Were there any places on the grid where your discussion with your partner changed your perspective? If so, what were they?

**Discuss:** Have students share places where they changed their mind. Focus particularly on blind spots, assumptions, or misconceptions that students had about access to their personal information that may have been revealed through discussion with a partner.

#### Investigating Social Profiles

**Goal:** Explore how small pieces of personal information spread across multiple sites on the Internet can produce a fairly detailed picture of a person, known as a <u>digital footprint</u>.

Group: This activity can be done individually, but is better when students are in groups of 2-3

#### **Distribute: Social Sleuth - Activity Guide**

Transition: Send students to Code Studio.

#### 🖵 Code Studio levels

Lesson Overview	Teacher Overv	view Student Over	view	
<ul> <li>When Does Your Privacy Matter? - Activity Guide Exemplar (PDF   DOCX)</li> <li>View on Code Studio ☑</li> <li>Social Sleuth - Activity Guide (PDF   DOCX)</li> </ul>				
Social Sleuth 🖹	Teacher Overview	Student Overview		

Social Sleuth

View on Code Studio 🗹

**Step 1:** Students look through all of the provided social media pages on Code Studio. The pages represent several unique individuals, each of whom has an account on three social media platforms. They are distributed as follows (do not share this with students):

	FaceSpace	Chirpr	instantframe
John Thomas	John T	@johnsnow	johnsnow
Haley Gutierrez	Haley G	@flyinghail	desertrider
Lizzie Dell	Lizzie Dell	@lillizard	photolizzie

Tip:

Students are asked to choose two users from the available profiles, but if you are short on time, you can ask

**Step 2:** Once they've looked through the pages, students will attempt to figure out who the two users are and answer some detailed questions about them. The footprint questions are designed to push students towards combining details from multiple social platforms and understanding that together the profiles represent more detailed information than the users probably intend to reveal about themselves. Some questions may not be answerable for all users, or may have different levels of details (full street address for one user, but only a city and state for another).

Share: Have the class share the digital footprints that they developed through the activity.

#### Discuss:

- Which information was most difficult to find?
- Which details were innocuous on their own, but revealed private information when combined with other details?

## Wrap Up (5 minutes)

**Goal:** Develop a common understanding of how our choices to publish information online contribute to a digital footprint.

#### Shaping Your Digital Footprint

#### Remarks

The activities today focused on how social media websites contribute to a digital footprint. As we prepare to publish our first web pages, you'll need to think about how those contribute to your digital footprint.

#### Q

**Prompt:** With your elbow partner, come up with a checklist that you can use to determine what should, and shouldn't, be posted online.

Share: Allow groups to share out their checklists,

using the responses to develop a class-wide web

#### Discussion Goal

**Goal:** Use this discussion to create a lasting checklist of best practices for when publishing information online. Possible checks include:

- Could someone identify me with this information?: for example name, address, phone number, etc.
- Do I want everyone to see this?: Don't publish anything you don't want to possibly be published to the world.
- Do I want this to be permanent?: You lose control of information once it's published and it could be around for your entire life.

Students' lists may be different but should cover these principles and additional ones they saw in today's lesson.

publishing checklist. Consider making a poster of your class-wide checklist that you can refer back to throughout the unit.

## **Standards Alignment**

CSTA K-12 Computer Science Standards (2017)

- ▶ IC Impacts of Computing
- ▶ NI Networks & the Internet



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

## 



## Lesson 6: Lists

## Overview

Students are introduced to ordered and unordered lists in HTML and work through a few levels in which they use the , , and tags. They then go back to their project, where they add a new HTML page. Inside the new page, they write the HTML to display a recipe, top ten list, or any other content that uses the new tags that they have learned.

## Purpose

This lesson introduces list elements, which are different from the previous elements in that they involve nested tags. Students should understand that the list item elements (<li></li>) go inside the list elements (<li></ul> or <ol></ol>).

Students also continue to work on their projects, adding a new HTML page. Students are free to choose the content of this page, but it should include some type of list. Right now there is no way for users to navigate from one project page to another, but students will learn more about how to link the pages together in Lesson 9, right before they publish their project.

## Agenda

Warm Up (10 minutes) Quick Share Activity (40 minutes) Web Lab: Lists Wrap Up (10 minutes) Journal

# View on Code Studio **Objectives**

Students will be able to:

- Use the , , and tags to create ordered and unordered lists in an HTML page.
- Create and name a new HTML page.

## Preparation

Have student journals (or project sketches) ready if they will use them to generate ideas for their lists page.

## **Introduced Code**

- •
- •
- •

## Warm Up (10 minutes)

#### Quick Share

#### 🎍 Remarks

In the last lesson, we looked at different types of content that we would or would not want to share with the world. Today you'll create an entirely new page for your project, and you'll need to think about the content of that page, but you'll also be using a new structure: lists.

**Prompt:** Take a few minutes to brainstorm some content that you would like to share with the world that you think could be structured as a list.

#### Q

**Discuss:** Students should brainstorm individually, then share with partners, then finally share with the whole class.

**Prompt:** Sometimes we use numbers to order our lists. Which of these lists do you think should be numbered, and which should not be numbered?

**Discuss:** Allow students to share their ideas for when numbering is appropriate.

#### Remarks

In HTML, there are two types of lists, ordered lists, which have numbers, and unordered lists, which have bullets. We're going to learn how to make both types of lists today so that you can use this structure in your project.

## Activity (40 minutes)

#### Web Lab: Lists

#### Ŷ

Group: Place students into pairs.

**Circulate:** Helps students as they work through the first set of levels. Keep track of when pairs make it to Level 5.

#### Discussion Goal

**Goal:** The goal of this discussion is to allow students some time to think about how lists are used, and to think about the difference between ordered and unordered lists before seeing them in HTML. Some possible lists students might come up with might be lists of their favorite things, top 10 lists, a shopping list, etc.

When students are discussing whether lists should be ordered or unordered use examples to highlight the difference. For example a list of finishers in a race should probably be ordered, while a grocery list probably does not need to be. Return to this discussion later in the lesson if need be.

#### **§** Teaching Tip

**Pair Programming:** More on Pair Programming can be found in the **Computer Science Discoveries Curriculum Guide**.



## Review

Expanding Your Website	昏 7	(click tabs to see student view)		
Expanding Personal Websi	te	8 (click tabs to see student view)		

## Wrap Up (10 minutes)

#### Journal

**Prompt:** Have students update the "HTML Tags" page in their journals with the tags they learned in this lesson.

**Prompt:** Have students reflect on their development of the **five practices of CS Discoveries** (Problem Solving, Persistence, Creativity, Collaboration, Communication). Choose one of the following prompts as you deem appropriate.

- Choose one of the five practices in which you believe you demonstrated growth in this lesson. Write something you did that exemplified this practice.
- Choose one practice you think you can continue to grow in. What's one thing you'd like to do better?
- Choose one practice you thought was especially important for the activity we completed today. What made it so important?

## **Standards Alignment**

CSTA K-12 Computer Science Standards (2017)

► AP - Algorithms & Programming



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

## **Lesson 7: Intellectual Property and Images**

## Overview

Starting with an discussion of their personal opinions on how others should be allowed to use their work, the class explores the purpose and role of copyright for both creators and users of creative content. They then move on to an activity exploring the various Creative Commons licenses as a solution to the difficulty in dealing with copyright.

Finally, with a common understanding of the restrictions of various Creative Commons licenses, students learn how to add images to their web pages using the <img> tag.

## Purpose

Until this point the only content that students have used on their web pages is their own, but it's common, and quite useful, to be able to integrate information and media created by others. Before learning about the technical ways to do this, however, we need to step back and gain a common understanding of the restrictions and purpose of copyright. We use the Creative Commons license as a clearer alternative to the more restrictive standard copyright and guide students through searching for, using, and properly citing Creative Commons licensed media.

## Agenda

Warm Up (10 min) Use of Your Work Activity 1 (20 min) The Creative Commons Solution Activity 2 (20 min) Adding Images Wrap Up (5 min) Your Own CC License

# View on Code Studio Objectives

Students will be able to:

- Explain the purpose of copyright.
- Identify the rights and restrictions granted by various Creative Commons licenses
- Add an image to a web page

## Preparation

Preview Creative Commons Overview -Video. You may need to download it before school if Youtube is blocked.

## Links

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

For the Teachers

• Licensing Your Work - Exemplar

For the Students

- Creative Commons Overview Video
- Licensing Your Work Activity Guide Make a Copy -

## Vocabulary

- **Citation** A quotation from or reference to a book, paper, or author, especially in a scholarly work.
- **Copyright** the exclusive legal right to print, publish, perform, film, or record literary, artistic, or musical material, and to authorize others to do the same
- Creative Commons A collection of public copyright licenses that enable the free distribution of an otherwise copyrighted work, used when an author wants to give people the right to share, use, and build upon a work that they have created
- Intellectual Property A work or invention that is the result of creativity, such as a piece of writing or a design, to which one

has rights and for which one may apply for a patent, copyright, trademark, etc.

## Introduced Code

• <img >

## Warm Up (10 min)

#### Use of Your Work

#### 🎍 Remarks

All of you have been working hard to create a new website that you're going to publish to the world. When you publish something, though, it can be hard to control what other people do with it. Sometimes people use our work in ways that don't seem fair to us. I'm going to describe a few situations for you. If you think what happened was fair, stand up. If you think it was unfair, sit down.

Ŷ

- 1. You take an awesome picture and someone puts it on their social media account and with your name beside it.
- 2. You write a story and someone else publishes it and says that they wrote it.
- 3. You write a song and someone sings it to her friends.
- 4. You write a song and someone sings it at a concert and makes a lot of money.
- 5. You take a picture and someone else Photoshops it and puts the new version on his web site.

#### Teaching Tip

These prompts aim to relate broader questions of copyright to students' own experiences creating and sharing content online. To make this hit close to home you may want to change the prompts to pick particular forms of social media that students are interested in. If you know students don't use a lot of social media, then alter the prompt to reflect another area where your students might think their own creative work was used inappropriately.

#### Remarks

It's okay if not everyone agrees how they want their work to be used. Copyright law says that whoever creates new content, such as a picture, a story, or a song, gets to decide how other people are allowed to use it.

Prompt: What rules would you make for people who want to use your creative work?

**Discuss:** Have students journal individually, then share with a neighbor, and finally discuss as a whole class.

## Activity 1 (20 min)

#### The Creative Commons Solution

#### Remarks

Copyright is granted the moment something is created, so unless explicitly told otherwise, we have to assume any picture, video, or other media we find online is fully covered by copyright law, which means that no one else can make copies or post it online without permission.

Sometimes, though, we want people to share our work so that more people can see it. The Creative Commons (CC) license was developed to help content creators have more specific control over how other people can use their work.

Group: Pair students.

Distribute: Give each student a copy of Licensing Your Work - Activity Guide

#### S Licensing Your Work

#### License Components

Ŷ

The first page of this activity covers the components of a Creative Commons license and asks students to brainstorm reasons why a content creator might want to include each component in their license. Push students to think critically here about both the value each component adds to the creator, but also the ways it might limit

the cultural exchange of ideas.

#### **Choosing the Right License**

The second page of this activity provides students with six CC licenses and three content creator scenarios. For each scenario, students are asked to evaluate which of the six provided licenses is the *least restrictive* but still addresses the concerns and needs of the content creator.

Q

**Discuss:** Ask several students to share out their responses to the *Choosing the Right License* scenarios. Encourage discussion and debate if students identified different licenses for the same scenarios.

## Activity 2 (20 min)

Code Studio levels

#### Adding Images

**Transition**: Send students to Code Studio, where they will learn about the <img> tag and have an opportunity to add images to the pages they've already created.

#### **§** Teaching Tip

**Is it Derivative?:** Students may struggle with whether the use of a work is derivative or not. The rule of thumb is to consider whether your use of a licensed work would be considered a *new* copyrightable work.

Converting an image from digital to print, or a song from CD to mp3 would not create a derivative work, but animating a drawing or adding new lyrics to a song *would* be creating derivative works.

#### Discussion Goal

**Goal:** While there may not be a specific "right" license for each of these scenarios, encourage students to always consider whether the license they chose is more restrictive than it strictly needs to be. One of the design goals of the Creative Commons license is to increase the amount of creative material available to the general public, promoting the sharing of intellectual property for the common good. With this in mind we should only be adding the minimal restrictions needed to meet the content creator's wishes.

Lesson Overview 🗎	Teacher Overv	view Stu	udent Overvie	w		
• Licensing Your Work - Activity Guide Exemplar View on Code Studio ☑						
Images in HTML 🕅 Student Overview						
Adding Images 🖵 3	<b>4</b>	) (click t	abs to see st	udent view)		
Creative Commons Search 🕅 Teacher Overview Student Overview						
View on Code Studio C Recommended Search Engines						
Because we can't know which sites might be blocked in your district, we've avoided pointing students to a specific search engine. Not all search engines make it easy to set Creative Commons filters - some of the easiest include:						

- Creative Commons Search
- Google Image Advanced Search
- Wikimedia Commons
- Flickr Creative Commons

As with any site with crowdsourced content, search engine results will change from day to day, and some of those results may be inappropriate for the classroom. You may want to check these sites shortly before displaying them to the class, and decide what is best for your classroom.

Attribution 7 🖵 **B** (click tabs to see student view)

## Wrap Up (5 min)

#### Your Own CC License

Journal: Have students update their "HTML Tags" list with the tags they learned in this lesson.

Journal: Think of all the personally created items that you've put on your website so far, and those that you may add in the future. Without a clear license all of those of things (and your web pages themselves) and covered under the fully restrictive copyright. Which Creative Commons license would you rather use (if any) and why?

If there is time you may want to consider having students return to their site and add the Creative Commons license they decided they would like their website to have.

Check out the Creative Commons license chooser

Students can add a footer at the bottom of each page with their Creative Commons License.

## **Standards Alignment**

CSTA K-12 Computer Science Standards (2017)

- AP Algorithms & Programming
- ▶ IC Impacts of Computing



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



## Lesson 8: Clean Code and Debugging

## Overview

Students deal with common issues that arise when designing web pages in HTML. Students will correct errors in a sequence of increasingly complex web pages. In the process they will learn the importance of comments, whitespace, and indentation as tools for making web pages easier to read. At the end of the lesson students create a list of strategies for debugging web pages and ensuring they are easy to read and maintain.

## Purpose

<u>Bugs</u> in HTML are more forgiving than programming languages such as JavaScript (the language used in Unit 3). However <u>debugging</u> is an explicit problem solving process that students will use repeatedly when working with any language on the computer. When problem solving there are different strategies that a computer scientist can use to find the source of the issue.

In addition to the strategies to fix bugs once they have occurred there are certain styles of writing HTML code that help prevent bugs or make it easier to find bugs. The three main style conventions used are <u>comment</u>, <u>whitespace</u>, and <u>indentation</u>. To motivate students to consider using these conventions in the future, the debugging levels demonstrate that it is easier to debug a program that is written with these style conventions.

## Agenda

Warm Up (10 minutes) Previous Experience with Bugs Activity (35 minutes) Web Lab: Smash Those Bugs! Wrap Up (10 minutes) Coding Style Conventions

# Vlew on Code Studio Objectives

Students will be able to:

- Describe why using whitespace, indentation, and comments makes your code easier to maintain.
- Develop a set of techniques for preventing bugs in HTML code and finding them when they occur

## Preparation

Prepare poster paper, sticky notes, and markers

## Vocabulary

- **Bug** Part of a program that does not work correctly.
- **Comment** A note in the source code of a computer program that helps explain the code to people who read it
- **Debugging** Finding and fixing problems in an algorithm or program.
- Indentation The placement of text farther to the right or left of the surrounding text, making it easier to understand the program's structure
- Whitespace Any character that shows up as a blank space on the screen, such as a space, a tab, or a new line; helps separate different parts of the document to make it easier to read

## Introduced Code

• <!--->

## Warm Up (10 minutes)

#### Previous Experience with Bugs

**Set up:** Put an poster up on the wall where all students can get to it. Write the title "Class Bugs" at the top but wait to explain the meaning of the term until it is introduced below.

Group: Place students in groups of 3-4.

**Distribute:** Give each group a handful of sticky notes

**Prompt:** With your group:

- Come up with at least three specific problems you have encountered while trying to write web pages in HTML.
- What project were you working on?
- How did you ultimately track down and fix the problem?

#### Q

**Discuss:** Have students share out the bugs they have faced and strategies they have used for finding them.

**Vocabulary:** Introduce the concept of <u>bug</u> and <u>debugging</u> explaining them in the context of the problems and strategies students shared.

Ŷ

**Share:** Ask each group to use the sticky notes they were given to write down at least three of the bugs they've encountered and how they solved them, using one sticky per bug. Each sticky should have:

- 1. A brief description of the bug
- 2. Steps taken to solve it
- 3. Name of student who solved it

Once groups have written down their bugs, have them stick them up on the class poster.

#### 🎍 Remarks

The problems you had when your HTML code did not work correctly are bugs. The process of fixing bugs and strategies used to fix them is called debugging. Today we're going to be working on our debugging skills and learning some strategies to keep our code clean to help avoid them.

## Activity (35 minutes)

#### Web Lab: Smash Those Bugs!

Group: Put students in pairs to work on these Code Studio levels.

#### Discussion Goal

**Goal:** Students do not need to get all the bugs or the strategies in this first discussion. They will add to this list throughout class.

Bugs students might share could include:

- Closing and opening tag switched
- Not closing a tag
- Not putting quotes around attributes values
- Spelling a tag name wrong
- Not putting a list item inside a type of list element
- Using the wrong header tag
- Wrong file path for an image
- Forgetting the equals sign between attribute name and value
- Closing tags out of order
- Not putting content inside a tag

Strategies students might share for debugging could include:

- Guess and Check
- Taking out sections to see which section is causing problems
- Looking for missing tags
- Figuring out what section of the code has the problem (Making the problem smaller)
- Asking a friend for help
- Thinking about when the code last worked and what you have added since then

#### ♀ Teaching Tip

**Using the Bugs Poster:** Encourage students to use this bugs poster as a resource through the rest of the unit. Whenever a student successfully squashes a new bug, have them put it on the poster. When students are get stuck, they can check the bug poster for someone who may have encountered the same issue before.

**Use Journals:** If you like you can have sutdents individually track their bugs on a new page in their journals, just like the "HTML Tags" page that they have been updating.


### Wrap Up (10 minutes)

### Coding Style Conventions

**Prompt:** What made it harder or easier to debug the web pages?

#### Q

**Discuss:** Have students share out things that made it easier or harder to debug the web pages they encounter.

#### 🖢 Remarks

Code needs to be useful for both people and computers. Code that your computer can run might still be really hard for someone (or even you!) to read and make changes to. From now on it's important that we use these practices to ensure our code is easy to read for people, not just good enough for a computer to use.

#### Discussion Goal

**Goal:** Students answers will vary but should hopefully include the following:

- Number of bugs
- Use of comments (text between the characters <!-</li>
   and -->)
- Separating things onto separate lines (whitespace)
- Grouping together things that are one idea such as a list (whitespace)
- Indenting elements that are inside other elements

If students don't mention some of these things ask them to compare two sites that have different uses of these elements.

**Setting Expectations:** Use this discussion to motivate the need for making readable code. Highlight that you will expect them to follow the norms they learned today from now onwards, in particular on their projects for this unit.

### **Standards Alignment**

CSTA K-12 Computer Science Standards (2017)

► AP - Algorithms & Programming



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



# Lesson 9: Project - Multi-Page Websites

### Overview

After learning about how to link web pages to one another, students are finally able to publish the website they have been working on. In this lesson, they link together all the previous pages they have created into one project, create a new page, and add navigation between the pages before publishing the entire site to the Web.

### Purpose

In several of the previous lessons, students created web pages that will be included in their websites. In this lesson, they will learn to link all of these pages together and publish them to the Internet.

### Agenda

#### Warm Up (15 minutes)

**Content Brainstorm** 

#### Activity

Multi-Page Websites Peer Review

#### Wrap Up (10 minutes)

Journal: Personal Website

# Vlew on Code Studio **Objectives**

Students will be able to:

• Connect multiple web pages into one website using hyperlinks.

### Links

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

#### For the Teachers

• Personal Website - Exemplar

#### For the Students

- Personal Website Project Guide Make a Copy -
- Personal Website Peer Review Make a Copy -
- Personal Website Rubric Make a Copy -

### Vocabulary

 Hyperlink - A link from a HTML file to another location or file, typically activated by clicking on a highlighted word or image on the screen.

### **Introduced Code**

• <a></a>

### Warm Up (15 minutes)

### **Content Brainstorm**

#### 🎍 Remarks

In this lesson, you'll be publishing your website to the Internet so that anyone with the address can see it. Before you do that, though, you're going to have a chance to add anything that you think might be missing from your site.

Prompt: What are some things you want on your site that you haven't had a chance to add yet?

Give students a few minutes to brainstorm some ideas, then allow them to share with the class.

#### Remarks

In a moment, you'll be able to create new pages using any of these ideas. Before we do, though, we need to make sure that our users will be able to get to all of the pages on our sites. To do that, we'll need to use a <u>hyperlink</u>.

### Activity

### Multi-Page Websites

#### Ŷ

**Distribute** the **Personal Website - Project Guide** and review the first page as a class.

**Circulate:** Support students as they complete the first two pages of the Project Guide. When students are finished designing their pages, have them transition to Code Studio.

#### 😵 Teaching Tip

Adding Multiple Pages: Students who decide to add more than one web page will need extra paper to complete their sketches. Alternatively, you may choose to have students complete their sketches and image tables in their journals.

### 🖵 Code Studio levels

- Lesson Overview 🖹
- Teacher Overview
- Student Overview
- Personal Website Project Guide Exemplar (PDF | DOCX)
- Personal Website Peer Review Exemplar (PDF | DOCX)

# Overview

This lesson covers hyperlinks, which allow web developers to connect pages together into one website. The class will link together all the previous pages into one project, and create navigation bars for each page before publishing the entire site to the Web.

### Vocabulary

• **Hyperlink** - A link from a HTML file to another location or file, typically activated by clicking on a highlighted word or image on the screen.

#### View on Code Studio 🗹

### Introduced Code

• Hyperlink - <a></a>

### Resources

- Personal Website Project Guide
- Personal Website Peer Review
- Personal Web Site Rubric
- Hyperlinks 🗎
- Student Overview
- Levels
- 🖵 3
- 🖵 4

### Student Instructions

# **Making a Multi-page Site**

A web site should include more than one web page, and they should connect to each other. Here's an example of a project with multiple pages that are connected by hyperlinks (the blue underlined words that take you to another page when you click on them).

# **Do This**

- Use the inspector tool to see the code that creates the hyperlinks on the top of index.html.
- Discuss with your partner how you think that code works.
- Click on recipe.html in the files list on the left hand side. Add a hyperlink to the top of this page that will take you back to index.html.

### Student Instructions

# **Create a Navigation Bar**

Here's your project from before. You'll be linking the two pages by creating a **navigation bar** at the top of the page. Websites usually have the same navigation bar across all of their pages so you can get back and forth between things on the site. You can see an example of a navigation bar on **this page** 

# **Do This**

- In your home page (index.html), add two hyperlinks at the top of the page, one to your lists page, and one to the home page.
- Even though it might seem strange to link a page to itself, it's less confusing if the navigation bar is exactly the same on every page.
- Copy the code for the navigation bar, then add it to the top of your lists page.
- Test your links in the preview area by clicking on them.

View on Code Studio 🗹

View on Code Studio 🗹

- What are hyperlinks?
- How do I add a hyperlink?

- What are hyperlinks?
- How do I add a hyperlink?

- Project Guide 🖺
- Teacher Overview
- Student Overview

# **Planning Checkpoint**

Check over student plans before allowing them to continue on the rest of the levels.

# **Your Personal Website**

In the next few levels, you'll be completing and publishing your personal website.

You've already done a lot of work on your website, but this is your chance to get creative and add a new page of something that you care about.

Here are some ideas for your new page, but you can do anything approved by your teacher.

- A movie, television, or book review
- Description of a favorite holiday tradition
- A "cause" page that explains an issue that is important to you

If you have more than one idea, you can add multiple pages to your site.

Your Project Guide will help you remember everything you need to do. Make sure you have your teacher's approval before moving to the next level.

### Examples



- Levels
- 🖵 6
- 🖵 7
- 🖵 8
- 🖵 9
- 🖵 10
- 🖵 11

### Student Instructions

#### View on Code Studio 🗗

# **Create and Name Your New Page**

Before you start coding, you should make sure that you have all the pages that you need in your project.



View on Code Studio 🗹

# Cats Cats Cats Cats You Should Adopt Reasons to Adopt a Cat How to Care for a Cat

- Create a new HTML page and rename it. (Look at Lesson 6 Level 8 if you've forgotten how.)
- Repeat this process for any other pages you want to add.

Student Instructions

**Upload Images** 

Next, you'll need to upload all of your images into your project.

# **Do This**

- Download all the images you want onto your computer. Visit Lesson 7 Level 5 for help if you need it.
- Make sure you have the right to use these images.
- Write down all of the attribution information that you need.
- Upload all of the images into your project. Visit Lesson 7 Level 5 if you've forgotten how.

### Student Instructions

View on Code Studio 🗹

View on Code Studio 🗹

# Add HTML

Next, you should add the HTML to your pages.

# **Do This**

- For each of the new pages that you've created, add HTML to make the page you sketched in your project guide.
- You may also add more HTML to the pages you've already created.

### Student Instructions

#### View on Code Studio 🗹

# **Navigation Bar and Footer**

Each of your pages needs a navigation bar and a footer.

# **Do This**

- For each page, make sure it has a navigation bar that links to other pages in your site. For a review of the navigation bar, go to Lesson 9 Level 4.
- For each page, make sure it has a footer at the bottom that explains the copyright information for your page. (Go to **Creative Commons** for help in creating this HTML.)

### Student Instructions

#### View on Code Studio 🗹

## **Review Your Project**

You're almost ready to publish your project, but you should check one more time that you have fulfilled all the requirements for this project.

# **Do This**

- Check that you are comfortable sharing all the information that's on your site.
- Check that all of your images are correctly attributed, and that you have the right to put them on your site.
- Check that your project includes everything included in the project rubric.
- Use comments and whitespace to make your website readable.

• Ask your partner to help you find any potential problems with your site.

#### **Student Instructions**

# **Share Your Project**

It's time to share your project with the world!

### **Do This**

- Click the Share button at the top of this page.
- Copy the URL it gives you, and email it to the people you want to see your page.

#### **Peer Review**

Group: Place students in pairs.

#### Distribute: Personal Website - Peer Review

**Support:** Help students as they complete the Peer Review Guide, then allow students to make any final revisions to their websites.

Collect: Project Guides, Peer Review Guides, and student website addresses.

### Wrap Up (10 minutes)

#### Journal: Personal Website

#### Q

**Prompt:** After the first day of pulling together your personal website reflect on your experience.

- How did you use the problem solving process to in creating your site?
- What do you like about your site so far?
- What do you still want to know how to do?

#### Send students to Code Studio to complete their

reflection on their attitudes toward computer science. Although their answers are anonymous, the aggregated data will be available to you once at least five students have completed the survey.

go.

Discussion Goal

Goal: Students should reflect on the process so far and

keep track of the fact that this is an iterative process

where they are constantly improving things as they

### 🖵 Code Studio levels

- Levels
- 🛇 Ξ 12

#### **Student Instructions**

This level is an assessment or survey with multiple questions. To view this level click the "View on Code Studio" link.

Discuss: Have students share out the things they still want to learn how to do.

Keep this list and cross it off as they learn more things in the unit.

Journal: Have students add the new tags they learned to the "HTML Tags" section of their journal.

### **Standards Alignment**

- ► AP Algorithms & Programming
- ► IC Impacts of Computing



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

# $\begin{array}{c} \begin{array}{c} \text{UNIT} \\ \hline 2 \\ \hline 13 \\ \hline 13 \\ \hline 14 \end{array} \end{array}$



# Lesson 10: Styling Text with CSS

### Overview

This lesson introduces CSS as a way to style elements on the page. Students learn the basic syntax for CSS rule-sets and then explore properties that impact HTML text elements. They work on a HTML page about Guinness World Record holders, adding their own style to the provided page. In the last level, students apply what they have learned about styles for text elements to their personal web page.

### Purpose

While there are many CSS properties, the properties highlighted in the levels are simple properties that can style text elements. Students will use these properties often. More properties are covered in subsequent lessons.

### Agenda

Warm Up (5 minutes) Journal: HTML Appearance Activity (40 minutes) Web Lab: Introduction to CSS Wrap Up (10 minutes) Recording CSS Properties

# View on Code Studio **Objectives**

Students will be able to:

- Use CSS selectors to style HTML text elements.
- Create and link to an external style sheet.
- Explain the differences between HTML and CSS in both use and syntax.

### Preparation

Create a new poster titled **CSS Properties** if your students will not be tracking new CSS properties in their journals

### Vocabulary

- **CSS** Cascading Style Sheets; a language used to describe how HTML elements should be styled
- **CSS Selector** the part of a CSS rule-set that defines which HTML elements the style should be applied to

### Introduced Code

- text-decoration: value
- text-align: value
- font-size: value
- font-family: value
- color: value

### Warm Up (5 minutes)

### Journal: HTML Appearance

**Prompt:** In the past few lessons, we've been using HTML to write our web pages. HTML lets us use tags to define the structure of a page. With your partner, come up with a list of all the different HTML tags as you can recall, and what they mean.

#### Q

**Discuss:** Choose one of the tags that describes text on the screen ( , , <hl>, etc.) and have the students describe how the contents of that tag appear on a web page. Ask students whether they

#### 오 Discussion Goal

**Goal:** Students may mention that the <h1> tag has large font, or that the tag puts a number or bullet before the text. Some other options that they may not consider are that all the text appears in black, and uses the same font. None of the text is underlined or bolded. Prompt students to realize that there are other possible styles that they see on web pages that it not the default appearance when HTML is used.

would *always* want those elements to appear in that exact way. For example, always means that the content is a paragraph, but should paragraphs always look the same on every page and on every site?

#### 🎍 Remarks

So far we have only made web pages where we control the content and structure, such as which parts of the pages are headers, lists, or paragraphs. We've been using HTML as the language to specify the content and structure of the pages. While HTML allows us some control over how the page looks, but doesn't give developers much control over the specific the look and style of each element. To do that, we need a language to express sty/e. Web developers use CSS to specify the style of a page. We use different languages for structure and style because it makes it easier to differentiate between the code for style and the code for structure.

### Activity (40 minutes)

Web Lab: Introduction to CSS

Transition: Send students to Code Studio.

🖵 Code Studio levels

Lesson Overview 🗎

Student Overview

Exploration: Adding Style 
Teacher Overview
Student Overview

Have students explore this level with a partner. Afterwards lead a short share out so **View on Code Studio** that partners can share with the class their responses to the three questions in the instructions. The subsequent video should help reinforce what students discovered so there's no need to lead a lengthy debrief conversation.

Video: Intro to CSS 🛤

**Student Overview** 

Styling Text 🗣 4 🗣 5 (clic

(click tabs to see student view)

**Student Overview** 

View on Code Studio 🗹

## Introducing Content-Structure-Style

This map level, and in particular the first section, introduce the Content-Structure-Style paradigm that students will use to help understand the differences between HTML and CSS. While they have previously been using HTML tags to label and categorize their content, they're now learning how to control the style of the different elements in their page using CSS. Students may have some misconceptions at this point since their browser has used default styles for their different kinds of content (e.g. <h1> tags by default make text larger). As they move through these lessons they should come to see that tagging content really just allows the browser's default styles, or the styles they define using CSS, to be applied to the correct pieces of content in their web pages.

Fonts and Style 7 🖵 9 **9 —** 10 (click tabs to see student view) CSS Text Properties 🚺 **Student Overview Teacher Overview** View on Code Studio 📝 **Differences in Browsers and Operating Systems** As mentioned in this map, you may find that different combinations of computers and browsers render fonts differently. While there is a common specification for how HTML and CSS should be read and displayed on the screen, each browser does so in a slightly different way, which can lead to the same page looking different on different computers. If your students encounter this, you can reassure them that it's a common challenge that web developers face.

In the case of font-family: fantasy, you can actually specify a list of fonts, from most to least specific. That way, if a user's computer has the exact font you specified, that's what they'll see, but if not the browser will try the next font on your list. To try and ensure that users see the font in the example image, you could write the css as font-family: Papyrus, fantasy. That way, even if a user's browser has a different default font for "fantasy", it will try to load the specific font "Papyrus" first.

Video:	Intro	to	CSS	Part	2	

Student Overview

**Styling your Website** 

🖵 14 (click tabs

🖵 13

(click tabs to see student view)

### Wrap Up (10 minutes)

**Recording CSS Properties** 

**Review:** Briefly review the "Content-Structure-Style" paradigm introduced in the map level during the lesson. Aim to distinguish between how HTML indicates the structure of a document and CSS now allows students to set the styles.

**Set Up:** Have students create a new T-chart in their journals called CSS Properties where students can easily get to it.

**Group:** Place students in groups of two to five - you'll need at least one group for each of the properties introduced in this lesson.

#### **?** Teaching Tip

**Journal or Poster?** Just as with the "HTML Tags" page in their journals, you may choose to have your class keep track of CSS Properties in a shared class poster.

**Jigsaw:** Assign each group one of the properties introduced today. Each group needs to come up with the a description and example for their property.

Share: Have groups add their properties to their journals or to the class "CSS Properties" poster.

### **Standards Alignment**

CSTA K-12 Computer Science Standards (2017)

► AP - Algorithms & Programming



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



# **Lesson 11: Styling Elements with CSS**

### Overview

This lesson continues the introduction to CSS style properties, this time focusing more on non-text elements. Students begin this lesson by looking at a website about Desserts of the World. They investigate and modify the new CSS styles on this website, adding their own styles to the page. After working on the Desserts page, students apply their knowledge of new CSS properties to do more styling of their personal websites.

### Purpose

This lesson builds on what students previously learned about CSS properties, this time looking at properties that can be used all elements. These new properties impact the visual look of the web page beyond its text and give students more precise control over the layout of their pages. As students expand their knowledge of style properties, many may have questions about how they can do even more with CSS. Curious students can explore more properties at **W3 Schools**.

### Agenda

Warm Up (5 minutes) Discuss: What Styles Do You Want? Activity (40 minutes) Web Lab: Styling Elements with CSS Wrap Up (5 minutes)

# View on Code Studio Objectives

Students will be able to:

- Use CSS properties to change the size, position, and borders of elements.
- Create a CSS rule-set for the body element that impacts all elements on the page.

### Introduced Code

- width: value
- margin: value
- height: value
- border-width: value
- border-style: value
- border-color: value
- background-color: value
- float: value

### Warm Up (5 minutes)

### Discuss: What Styles Do You Want?

Prompt: Yesterday we styled text elements. What other ways do you wish you could style your page?

#### Q

**Discuss:** Have students share styles they would like to be able to add their page.

#### 🎍 Remarks

Today we are going to learn more properties we can use to style our web pages. Hopefully some of them allow you to add some of the styles we listed in this discussion.

### Activity (40 minutes)

#### Discussion Goal

Goal: Students will likely bring up:

- Adding background colors to things
- Being able to move things around on the page (center, left, right)
- Being able to add borders to things
- Being able to control the size of objects

♀ Teaching Tip						
g pair programming articular as students						
esson Overview Student Overview						
<b>Exploring more CSS</b> $\square 2$ $\square 3$ (click tabs to see student view)						
The Body Element M Student Overview						
Layout with CSS 🖵 5 🖵 6 🖵 7 🖵 8 🖵 9 🖵 10 (click tabs to see student view)						
Layout Properties M Student Overview						
Styling your Website - 12 (click tabs to see student view)						

### Wrap Up (5 minutes)

**Journal:** Have students add the new properties they learned to the CSS Properties page in their journal or the class poster.

**Prompt:** Have students reflect on their development of the **five practices of CS Discoveries** (Problem Solving, Persistence, Creativity, Collaboration, Communication). Choose one of the following prompts as you deem appropriate.

- Choose one of the five practices in which you believe you demonstrated growth in this lesson. Write something you did that exemplified this practice.
- Choose one practice you think you can continue to grow in. What's one thing you'd like to do better?
- Choose one practice you thought was especially important for the activity we completed today. What made it so important?

### **Standards Alignment**

CSTA K-12 Computer Science Standards (2017)

► AP - Algorithms & Programming



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



# **Lesson 12: Sources and Search Engines**

### Overview

This lesson encourages students to think more critically about how web searches work and how to find relevant and trustworthy information online. After viewing and discussion a video about how search engines work, students will search for information about several unlikely animals. They'll need to analyze the sites they find for reliability in order to identify which of the animals is actually a hoax.

### Purpose

Students use search engines all the time. They take for granted how hard it is to produce search engine results in seconds. In order to give them a deeper understanding of the algorithms that control modern search, students will develop their own algorithms.

This lesson connects back to Unit 1, which is explicitly called out in the Wrap Up. This a problem that humans and computers solve differently. In fact it's a problem that computers solve better than humans, as a human would take too much time to search through huge amounts of data. Humans, however, still have a major role to play in searching effectively, as computers are unable to reliably tell us which websites contain trustworthy information and which are scams. Through this lesson, students will collaboratively create a class checklist that they can use to vet websites they come across for relevance and trustworthiness.

### Agenda

Warm Up (10 minutes)

Internet Scavenger Hunt

#### Activity (30 minutes)

Relevant and Trustworthy Sites The Trustworthiness Checklist

#### Wrap Up (5 minutes)

Human versus Computer Searching

# View on Code Studio Objectives

Students will be able to:

- Use basic web searching techniques to find relevant information online
- Identify elements that contribute to a website's trustworthiness or untrustworthiness

### Preparation

 Print a copy of Internet Scavenger Hunt -Activity Guide for each pair of students.
 Print a copy of Strange Yet True Animals
 Activity Guide for each pair of students.

Create a blank poster titled **Trustworthiness Checklist** and place it on the wall.

### Links

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

For the Teachers

• Strange Yet True Animals - Key

For the Students

- How Search Works Video (download)
- Internet Scavenger Hunt Activity Guide
- Strange Yet True Animals Activity Guide Make a Copy -

### Vocabulary

- Algorithm A list of steps to finish a task.
- Relevant closely connected to a topic
- Search Engine A program that searches for items on the World Wide Web.
- Trustworthy reliable, honest, and truthful

### Warm Up (10 minutes)

### Internet Scavenger Hunt

Group: Place students in pairs.

Distribute: Each group will need a copy of Internet Scavenger Hunt - Activity Guide

Transition: Send students to the computers to prepare for the scavenger hunt.

#### 🎍 Remarks

When I start the clock you will have exactly 7 minutes to complete as much of this scavenger hunt as you can. Your goal isn't to answer *every* question on this list, but to find as much *accurate* information as possible in the time you've been given.

Q

#### Prompt:

- Which things were hardest to find?
- Which were easiest to find?
- What strategies did you use to find things online?
- How did you know the information you found was correct?

#### Discussion Goal

**Goal:** The first three prompt questions are intended to help students identify any searching techniques that they currently use. If students *don't* have clear strategies, you may want to spend some time discussion basic search techniques. Keep track of search strategies on the board for reference later.

The last question gets to the focus of the activities in this lesson. It's likely that students didn't consider whether or not their information was accurate, but if they do come up with some ideas, make sure that you track them in a place where they can be referred to later in the lesson.

**Discuss:** Student pairs share out what they were able to discover about the scavenger hunt and what strategies worked well for finding information.

### Activity (30 minutes)

### Relevant and Trustworthy Sites

Display: Watch the How Search Works - Video with the class.

#### 🎍 Remarks

Now that we know a little bit more about how search engines work, we're going to search for some information about some of the world's strangest animals.

Group: Place students in pairs.

Distribute: Give each pair one copy of Strange Yet True Animals - Activity Guide

#### Inlikely Animals

**Search:** Give student pairs some time to search online and fill in the table on the activity guide. They should have little difficulty finding this information by just searching for the names of each animal.

**Share:** Ask student pairs to share out their answers for each of animals. If, during any point in the share out, others in the class disagree with an answer encourage them to discuss where and how they found their information.

**Vocab:** It's likely that students found websites that were <u>relevant</u> to their topics, but how do we know if they are <u>trustworthy</u>?

#### 🎍 Remarks

Ŷ

You did a great job finding websites that were <u>relevant</u> to your search. You found a lot of interesting information about those strange animals, but is any of it real? You and your partner have five minutes to decide which of these animals are real, and which are made up. For each of your decisions you'll need some supporting evidence that shows why your source websites are either <u>trustworthy</u>, or untrustworthy.

Ŷ

**Search:** Give students five more minutes to look through their source sites to try to figure out whether or not they're trustworthy.

**Share:** Allow each pair to share out their assessment of trustworthy and untrustworthy sites. Push students to give detailed reasons for why they trust a site or not, and keep track of them on the board. Once all of the pairs have had a chance to share, reveal the that the Tree Octopus was the only fake

#### ♀ Teaching Tip

Students may come upon sites that explain that the pacific northwest tree octopus is in fact a fictional animal. If so, you can ask them to think about why they believe some sites over others, and how they are sure that the other two animals are real. Have them continue filling out the worksheet with these questions in mind, and let them know that they will be able to share their thoughts later in the activity.

You can find more untrustworthy websites in **this** forum post.

#### Teaching Tip

**Just in Case:** The Pacific Northwest Tree Octopus is the fake animal.

animal. Depending on how accurate your students were, consider visiting the site as a class to explore some of the clues that it's not quite right.

### The Trustworthiness Checklist

Set Up: Start a poster on the wall labeled Trustworthiness Checklist.

Group: Place students in groups of 4-5.

**Prompt:** Now that we've identified some sites that are, and are not, trustworthy, work with your group to come up with a list of things to check for when trying to determine whether or not to believe a website.

#### Ŷ

**Share:** Have groups share out their lists. As a class, generate a class Trustworthiness Checklist on a poster. You can refer back to this in the future whenever asking students to find information online.

Discuss: Think about your own websites - do you think

#### Teaching Tip

**Prompting Conversation:** If your students are struggling to come up with ideas, consider prompting them with any of the following:

- What does the web address of a trustworthy or untrustworthy site look like?
- Is a well designed site more likely to be trustworthy, or untrustworthy?
- Would you expect a trustworthy site to have easily found contact information?
- Can you learn anything from the kinds of ads that a site displays?

people would find your site trustworthy or untrustworthy? What changes could you make to your site to encourage users to trust your content?

### Wrap Up (5 minutes)

#### Human versus Computer Searching

#### Q

**Journal:** In Unit 1, we looked at the differences between how computers solve problems and how humans solve problems. What are the differences between the way search engines work and how humans search for resources?

### **Standards Alignment**

CSTA K-12 Computer Science Standards (2017)

► IC - Impacts of Computing



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



# **Lesson 13: RGB Colors and Classes**

### Overview

In this lesson, students first learn how to specify custom colors using their RGB (red, green, blue) values. They then apply these colors to a new Four Seasons web page, which uses CSS classes. CSS classes allow web developers to treat groups of elements they want styled differently than other elements of the same type. Using classes, students add more styles to the Four Seasons web page, then use them to style their personal websites.

### Purpose

Up until this point, the only styling students have been able to do is styling by element, which means that every element of a particular kind has the same style. Classes allow for web developers to group together a set of elements they want to style. This means students can single out a certain element they want to style or group together elements from one or more types of elements. Once elements are in a class, the class can be used as a selector in a style rule.

Note: Single elements can also be selected by id, but this type of selection is possible with a class applied only to that single element. Because id selection does not add any extra functionality, it is not taught in this course.

### Agenda

Warm Up (5 minutes) RGB Colors Matching Activity (40 minutes) RGB and Classes Wrap Up (5 minutes) Journal: Reflecting on the Practices

# View on Code Studio Objectives

Students will be able to:

- Group elements using classes in order to create more specific styles on their website.
- Apply the rgb() color function to add custom colors to their website

### Preparation

Print a copy of RGB Colors - Activity Guide for every student.

### Links

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

#### For the Teachers

• RGB Colors - Key

#### For the Students

• RGB Colors - Activity Guide Make a Copy -

### Vocabulary

• CSS Class - An identifier that allows multiple elements in an HTML document to be styled in the same way

### **Introduced Code**

• rgb(red, green, blue)

### Warm Up (5 minutes)

### **RGB** Colors Matching

#### 🎍 Remarks

So far, you've been using color names to choose colors for your web pages. That means that you can only use the colors that someone else has already thought of. Today, you're going to learn how to mix new colors for your pages.

#### Distribute: Hand out the RGB Colors - Activity Guide.

**Transtion:** Send students to Code Studio to use the RGB widget at the beginning of this progression to complete the activity guide. You may want to answer one question together, then allow students to work in pairs to complete the first activity.

#### Remarks

On the computer screen, we mix colors by combining red, green, and blue. In the next few levels, you'll have a chance to use the color mixer to choose your own colors, then put them into a web page.

### Activity (40 minutes)

### **RGB** and Classes

**Transition:** Once students have completed the warm up, allow them to continue working through the remaining levels.

#### Code Studio levels

Lesson Overview 🖺	Teacher Overview	Student Overview						
• RGB Colors - Activity Guide (Pl	DF   DOCX)		View on Code Studio 🗗					
RGB Widget - Student Overview								
RGB Exploration	3 4 5	<b>G</b> (click tabs to see student view)						
CSS Classes N Student Overview								
Styling with Classes $\square 8$ $\square 9$ $\square 10$ $\square 11$ (click tabs to see student view)								

### Wrap Up (5 minutes)

Journal: Reflecting on the Practices

**Journal:** Have students update their "CSS Properties" page in their journal with the new information they've learned about classes.

**Journal:** Have students reflect on their development of the five practices of CS Discoveries (Problem Solving, Persistence, Creativity, Collaboration, Communication). Choose one practice you thought was especially important for the activity we completed today. What made it so important?

### **Standards Alignment**

CSTA K-12 Computer Science Standards (2017)

► AP - Algorithms & Programming



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



### Lesson 14: Project - Personal Portfolio Website

### Overview

Students have spent a lot of time throughout the unit working on their Personal Website. In the final couple of days students finalize their websites. They work with peers to get feedback, put the finishing touches on the websites, review the rubric and reflect on their process. To cap off the unit, they will share their projects and also a overview of the process they took to get to that final design.

### Purpose

This project emphasizes many of the core practices of this course. Students will need creativity, problem solving skills, and persistence to complete their websites. This final section of the project requires those practices as well as collaboration and communication. Peer review will encourage students to leverage their peers as resources. The final reflection and presentation allow students to practice communication about their work.

### Agenda

#### Day 1

Define: Rubric Review Plan: Finishing Personal Website Try: Finalize Personal Website

Day 2

Reflect: Peer Review Plan and Try: Final Touches Reflect: Final Reflection

#### Day 3

Showcase Set Up Student Website Showcase

#### View on Code Studio

### **Objectives**

Students will be able to:

- Apply CSS styles across an entire website
- Explain the design choices they made on their website to other people
- Prioritize and implement incremental improvements

### Preparation

Figure out a way to showcase all the students projects and get as many users as possible to check them out
 Plan out the day breakdown based on your class schedule
 Print a copy of Final Personal Website - Peer Review for each student.
 Print a copy of Final Personal Website - Project Guide for each student.
 Print a copy of Final Personal Website - Project Guide for each student.
 Print a copy of Final Personal Website - Rubric for each student.

### Links

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

#### For the Teachers

- Final Personal Website Project Guide Exemplar
- Final Personal Website Peer Review Exemplar

#### For the Students

- Final Personal Website Peer Review Make a Copy -
- Final Personal Website Project Guide Make a Copy -
- Final Personal Website Rubric Make a Copy -

### Day 1

#### Define: Rubric Review

#### 🎍 Remarks

Over the course of the whole unit you have been developing a personal website. Over the next couple of classes you will be creating a final version of that project. As we prepare to showcase your work, we will use the problem solving process as our guide to ensure we have the best quality product possible.

The first step of the problem solving process is to define the problem. To help define our problem we need to understand the expectations of the final product. The rubric for your Personal Website will help us. Let's go over it before starting.

Distribute: Give each student a copy of Final Personal Website - Project Guide.

Discuss: As a class, review the rubric portion of the Project Guide and clarify any questions students have.

### Plan: Finishing Personal Website

#### 🎍 Remarks

Next you will begin the planning stage of the problem solving process. Keep in mind you have limited time to complete this project so you will need to prioritize the work you need to.

Students should use **Final Personal Website - Project Guide** to plan out the work they will do to finish their Personal Website.

#### Try: Finalize Personal Website

#### Remarks

Now that you all have plans, it is time to begin implementing your plan. We will be doing a peer review as a class. You should work to make a polished product to share during the peer review.

Transition: Students log onto Code Studio and begin finalizing their websites.

### 🖵 Code Studio levels

- Lesson Overview 🖹
- Teacher Overview
- Student Overview

View on Code Studio to access answer key(s)

View on Code Studio 🗹

View on Code Studio 🗹

# Overview

In the last few days of the unit, the class finalizes their personal websites, working with peers to get feedback, review the rubric, and put the finishing touches on the site. To cap off the unit, everyone shares their projects and how they were developed.

### Resources

- Final Personal Website Peer Review
- Final Personal Website Project Guide
- Final Personal Website Rubric
- Final Personal Website 🖵

• Student Overview

# **Personal Website**

This is it! You have been working on your Personal Website for the whole unit. Put the finishing touches on it and submit it to your teacher! Remember to use the problem solving process as your guide in preparing your final product.

## **Do This**

- Define: Read the rubric so you know what is expected
- Plan: Decide what you still need to work on and put those things in priority order
- Try: Implement your improvements to your personal website.
- Reflect: Get feedback from a classmate and decide what of that feedback to act on.

Don't forget to take a screenshot of your home page and save it somewhere on your computer so you can find it later (*Need help with how to screenshot? Check out this* **website**.)

### Day 2

#### Remarks

You should now have a polished product you are proud of. An important part of any major project is to get feedback from people not working on that project with you. They may bring some perspective you might have missed. We will spend today giving and getting feedback. Then you will reflect on how to put this feedback into action.

#### **Reflect: Peer Review**

Distribute: One copy of Final Personal Website - Peer Review to each student.

Group: Pair students up.

#### *I* Peer Review Process

Students will:

- Open up their website projects in Web Lab.
- Fill in the top part of the worksheet, identifying what they would like feedback on.
- Trade places with their partner so their partner is now looking at their sheet and website.
- Give feedback on partners work.
- Switch back to their sheet and website to review feedback.
- Make a plan for implementing some of the feedback.

### Plan and Try: Final Touches

Transition: Students return to Code Studio and make any improvements that were identified in the peer review session. If they did not get any suggestions from the peer review, the teacher may want to give them some suggestions.

Students should also review the rubric as a final way to check their work.

### **Reflect: Final Reflection**

**Setup:** Students will need their journals and all of their screenshots of the progress on their home page. You may want printed versions if you are going to display them during the showcase.

#### Prompts:

- I am most proud of the following aspect(s) of my personal website:
- The biggest challenge that I overcame so far in making my personal website:

- Improvements I still want to make to my website are:
- If I knew at the beginning of this project what I know now, I would have started it differently by doing this:
- I would describe the level of effort I've invested into this project as:

Send students to Code Studio to complete their reflection on their attitudes toward computer science. Although their answers are anonymous, the aggregated data will be available to you once at least five students have completed the survey.



- Levels
- 🛛 🖽 3

### **Student Instructions**

#### View on Code Studio 🗹

This level is an assessment or survey with multiple questions. To view this level click the "View on Code Studio" link.

### Day 3

#### Showcase Set Up

Setup: Students need:

- A computer to display the website.
- A way to display their website progression screenshots

### Student Website Showcase

Students should stand next to their computers and talk to people attending the showcase about their work. If you can't get others to come visit your room for this activity, you can split the class in half and have one half present while the others circulate. Then they can switch.

### **Standards Alignment**

CSTA K-12 Computer Science Standards (2017)

► AP - Algorithms & Programming



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