**Lesson 3** Intro to the DOM

**What is the Document Object Model?**

| **Overview** | |
| --- | --- |
| In this lesson, students will be introduced to the DOM and use the p5 DOM library to dynamically add and modify HTML elements on a web page using JavaScript. | |
| **Lesson Objectives** | |
| Students will be able to   * Describe the Document Object Model and explain why it is useful * Use the p5 DOM library * Add and modify HTML using JavaScript | |
| **Suggested Duration** | |
| One period (45 minutes) | |
| **Blueprint Foundations Student Outcomes (**https://blueprint.cs4all.nyc/outcomes/) | |
| Algorithms  Prototype | **Explain why** I used specific instructions to complete a task. |
| Programming Communicate | **Discuss** what can and cannot be done with a specific set of commands. |
| Networks Analyze | **Explain** what markup languages are and the role they play in creating websites. |
| **Vocabulary** | |
| * **Document Object Model (DOM)**: Defines the structure of an HTML document and allows JavaScript to manipulate HTML elements on a web page. | |
| **Planning Notes** | |
| * The **Do Now** asks students to answer prompts based on *The Simpsons* family tree, but feel free to modify using a different fictional or historical family that is more culturally relevant to your students. | |
| **Resources** | |
| * Coding Train Tutorial: [Creating HTML Elements with JavaScript](https://www.youtube.com/watch?v=lAtoaRz78I4&list=PLRqwX-V7Uu6bI1SlcCRfLH79HZrFAtBvX&index=2) * [How a browser builds and displays a web page](https://youtu.be/DuSURHrZG6I?t=344) (start at 5:44) * [p5.dom reference](https://p5js.org/reference/#/libraries/p5.dom) | |
| **Assessments** | |
| * Assess the **Student Activity**. Check for the ability to:   + Use the p5 DOM library   + Use the createP() function to create paragraph elements and html() to modify the content of these elements * Assess the **Wrap Up**. Check for the ability to:   + Describe the Document Object Model and explain why it is useful | |

| **Do Now:** |
| --- |
| * **[Design Journal]** Use this image of *The Simpsons* family tree to answer the following prompts:      * + Homer and Marge are Maggie’s \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (one word beginning with “p”)   + Bart and Lisa are Homer’s \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (one word beginning with “c”)   + Bart, Lisa, and Maggie are \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (one word beginning with “s”) |
| **Discussion:** |
| * Transition from the **Do Now** by explaining that similar to a family tree, the elements of a web page also have a hierarchical tree structure, starting with the <html> element:      * <html> is the **parent** element of the <head> and <body> elements. <h1>, <p>, and <canvas> are called **sibling** elements because they’re all **children** of <body>. * Ask: What other parent, sibling, and child elements do you see?   + For example, students might observe that <a> is a child of its parent element <p>, or that <script> and <link> are siblings because they’re both children of <head>. * Explain that this tree structure is also known as the DOM, or Document Object Model. It is created when a browser first reads an HTML file and interprets its structure.   + First the browser fetches the elements inside the <head>, like CSS files or JavaScript libraries. Then it looks at the elements that make up the <body>. Finally, the browser turns all these elements into **objects**, the “O” in DOM. * Explain to students that because the elements are represented as objects, this allows JavaScript to change any HTML in **response** to user interaction. **The DOM allows JavaScript to manipulate web pages**. |
| **Teacher Demo:** |
| * Open up the p5 reference and direct students’ attention to the top of the page:      * Ask: Which of these topics are you already familiar with?  *Answer: Color (fill and stroke functions), Events (mousePressed and keyPressed functions), Image (loadImage and image functions) and Shape* * Tell students that they will learn new p5 functions to help them use the DOM. Click on the topic to reveal those new DOM functions, and then click on createP():      * Ask a student to read the description out loud. * Turn and Talk: Based on the description, what does the createP() function do? What does the **parameter** [html] represent? * Expand on student responses by saying that createP() is a function that uses JavaScript to create a paragraph element on a web page. This paragraph element is the same as the element created using <p> tags in HTML, but is made using the DOM.   Demo 1: Creating Paragraph Elements   * Build out [this example](https://editor.p5js.org/mparker/sketches/HGk7mkyFf).   + Ask students what they notice. Why is it useful to write JavaScript for this example, instead of just HTML?   + Students may point out that createP() makes a new paragraph element every time it is called. Explain that in order to modify a single paragraph, you need to save it inside a variable, which you will demo in the next example.   Demo 2: Modifying a Paragraph Element   * Expand on the previous example so the code looks [like this](https://editor.p5js.org/mparker/sketches/QRDwATOCU).   + Explain that in order to change the **content** of an HTML element, you will use the html() function, also known as a “method”.     - Note: An object-specific function is called an object “method”. You can choose to introduce this new vocab term to your students or just use the blanket term “function”.   + Ask: What is different about the way that the html() function is being called?   + Explain that DOM objects (like a paragraph) have access to special functions, which are called “methods”. The html() method doesn’t work as a “standalone function”, but it can change the text of a paragraph element when it is attached to a variable using **dot syntax**. * Have students write down what the code on line 15 should be to make it so that instead of *only* showing the number, the paragraph also says “Your lucky number is…”   + Answer: paragraph.html(“Your lucky number is...” + luckyNum) |
| **Student Activity:** |
| * Show [this sketch](https://editor.p5js.org/mparker/present/bnuALjPhF) to students without showing them the code. Have them work in pairs to recreate the sketch by following these prompts:   + Create 3 global variables for each paragraph   + Create 3 paragraphs and store them inside the variables in setup()   + Change the background color arguments so that the red and blue values are controlled by mouseX and mouseY (keep green value at 0)   + Inside draw(), update the paragraphs with to show changing red and blue values * Scaffold: If students are unsure how to control the background color using mouseX and mouseY, show them [this sketch](https://editor.p5js.org/mparker/sketches/afDd94FB) for reference. * Extensions for early finishers:   + If the mouse moves off the canvas, the values displayed will be greater than 255. Add conditional statements to prevent the values from exceeding 255.   + Right now, the green value is fixed at 0. Add a new feature that lets users control the green value of the background, and also updates the paragraph to display that value. The green value can be controlled by mouse position, or by mouse or key presses! |
| **Wrap Up** |
| * **[Design Journal]** Students should answer the following:   + In your own words, explain what the DOM is and why it is useful. |
| **Extensions: N/A** |
|  |