**Lesson 6** Mad Lib Mini-Project

**How can we make a Mad Lib using buttons and text inputs?**

| **Overview** | |
| --- | --- |
| In this mini-project, students will create their own Mad Lib game applying their knowledge of buttons and text input fields. | |
| **Lesson Objectives** | |
| Students will be able to   * Create paragraphs, text input fields, and buttons using the p5 DOM library * Define and use callback functions * Modify the HTML of a web page using values from text inputs | |
| **Suggested Duration** | |
| One or two periods (45 - 90 minutes) | |
| **Blueprint Foundations Student Outcomes (**https://blueprint.cs4all.nyc/outcomes/) | |
| Abstraction  Prototype | **Explain why** I chose to include the specific components of my prototype over others. |
| Algorithms  Prototype | **Demonstrate** the benefit of using an event, conditional, or loop in my prototype. |
| Networks Prototype | **Explain how** I used at least three different markup tags to build a website. |
| **Vocabulary** | |
| * N/A | |
| **Planning Notes** | |
| * For the **Do Now**, it may be helpful to print out cheat sheets for parts-of-speech like [this one](https://i.pinimg.com/originals/59/71/43/597143e33b755933ba516dca8ae60121.jpg) or [this one](https://i.pinimg.com/originals/83/7b/9e/837b9e41469899afc15f8a24b4173489.jpg). Determine which parts of speech to use in the prompt based whichever Mad Lib you choose to complete as a class. * For the **Discussion**, pick a Mad Lib in advance to complete together as a class. [This example](http://www.eduplace.com/tales/content/wwt_042.html) is called “Today’s Horoscopes”. * For the **Student Activity**, consider assigning the three-sentence story as homework before this lesson so more in-class time can be spent coding. | |
| **Resources** | |
| * [RiTa.js library](http://rednoise.org/rita/index.php) (for extension) | |
| **Assessments** | |
| * Assess the **Student Activity**. Check for the ability to:   + Create paragraphs, text input fields, and buttons using the p5 DOM library   + Define and use callback functions   + Modify the HTML of a web page using values from text inputs * Assess the **Wrap Up**. Check for the ability to:   + Describe debugging strategies   + Give constructive feedback to peers | |

| **Do Now:** |
| --- |
| * **[Design Journal]** Write at least one example of the following parts of speech:   + Adjective   + Verb   + Noun * Note: Depending on the Mad Lib you choose to complete with students during the **Discussion**, you may wish to add more specific categories to the list above. For example, this [Mad Lib](http://www.eduplace.com/tales/content/wwt_042.html) (which is also linked in the planning section) asks for an animal and a number in addition to more general nouns. |
| **Discussion:** |
| * Explain that today, students will be making a “Mad Lib” game where users fill in the blanks with words to create a wacky story. * Practice as a class using the Mad Lib you’ve selected and the words that students wrote down in the **Do Now**. |
| **Teacher Demo:** |
| * Ask students how to do the following as a refresher before they begin the activity. Write syntax up on the board for them to reference while they code. If students can’t remember certain syntax, model how to search for functions in the p5 reference:   + Create an input field *Answer: textBox = createInput(“optional default text goes here”)*   + Capture the text typed inside an input field *Answer: textBox.value()*   + Create a button *Answer: button = createButton(“button text goes here”)*   + Call a function when a button is pressed *Answer: Create a custom callback function and attach it to the button variable. button.mousePressed(nameOfCallback)*   + Create a paragraph *Answer: p = createP(“paragraph text goes here”)*   + Modify the html of a paragraph *Answer: p.html(“new paragraph text goes here”)* * It may also be helpful to show them a simple p5 Mad Lib without displaying the code. Here is an [example](https://editor.p5js.org/mparker/present/mNTXXyOKT). |
| **Student Activity:** |
| Students should follow these instructions to complete the Mad Lib:   1. Write a simple three-sentence story, and choose between **one and three words** that will be left blank for users to fill in themselves. 2. Create a **header** with the title of your Mad Lib. 3. Declare **variables** to hold your text input(s), your button, and your story. 4. Inside setup(), create a **text input** for each blank word, a **button** to reveal the Mad Lib, and a **paragraph** that will be modified with the text of the story. 5. Make a custom callback function that captures the **values** of the text inputs, and then modifies the paragraph using those values. 6. Call the custom function by attaching it to the button variable with mousePressed()   Here are some extensions for early finishers:   * Challenge: Write a conditional inside the callback function to check if the input fields are blank or if the default text has not been changed before generating the story. In either case, your sketch should display a message telling the user to type their own words in the boxes. * Super Challenge: Use the language library [RiTa.js](http://rednoise.org/rita/index.php) to check if the user is typing the correct part of speech before generating the story, and if not, prompting them to use the correct part of speech! For example, [here is the syntax](http://rednoise.org/rita/reference/RiTa/RiTa.isNoun/index.php) to check if a word is a noun. You’ll need to add this line of code to the <head> of your index.html file to use RiTa.js:   <script src="https://cdnjs.cloudflare.com/ajax/libs/rita/1.3.94/rita-full.js"></script> |
| **Wrap Up** |
| * [**Design Journal**] First ask students to reflect on the following:   + Briefly describe how your sketch works.   + When something was challenging, what strategies did you use to get unstuck?   + How might you change or add onto this project in the future? * Invite a few students to come up and share their process:   + Fill in the blanks with suggestions from the class and read out your Mad Lib!   + Share your responses to the reflection questions. * Non-presenting students should give feedback by sharing glows and grows. |
| **Extensions:** |
|  |