

Exercise 4: Client-side JavaScript

This lab continues to build JavaScript skills.

You may find it useful to refer to an online JavaScript references and tutorials, such as:

- [MDN Web Docs on JavaScript](#)
- [W3Schools JavaScript Tutorial](#)

W3Schools

Continue through the [W3Schools JavaScript tutorial](#), completing the [Objects](#), [Functions](#), [DOM](#), and [BOM](#) topics.

Creating a Wordle App



Update your Wordle function from the last lab so that rather than printing out which letters are correct or in the wrong place, it returns an integer array of length 5, where:

- a 0 in cell i means the i th letter is not in the word,
- a 1 in cell i means the i th letter is in the word, but in a different place,
- a 2 in cell i means the i th letter is in the word and in the right place.

For example, if the secret word is CRAZE, and you guess TRACK, the result should be [0,2,2,1,0].

BEWARE: Wordle treats duplicate letters in a specific way. If you have two duplicate letters in the guessed word, and only one in the secret word, only one of the letters will be yellow (or green). For example, if the secret word is CRAZE, and you guess TREES, it should return [0,2,1,0,0].

Once you have constructed this function and thoroughly tested it, you can use it to create the Wordle game using the DOM. The secret word can be a const variable in the page's JavaScript, and the user enters their guess into a text box (or you can use JS events to type directly into the wordle image). Then their guess is rendered into the HTML using the traditional Wordle colouring.

You should also add messages for when the user guesses the word or runs out of guesses.

Generating Pages Dynamically using the DOM

The aim of this exercise is to create the components of the page by directly constructing the DOM tree.

Read [Introduction to the DOM](#)

In the lectures, several properties and methods that are useful for traversing the DOM tree were discussed, such as: `previousSibling`, `nextSibling`, `firstChild`, `lastChild` and `parentNode`.

Several methods for manipulating the tree were also discussed, including: `insertBefore`, `appendChild`, `removeChild` and `replaceChild`.

Some other useful properties and methods include the `body` property and the methods `createElement` and `createTextNode`.

An online reference for these (and other) properties and methods can be found in the [Gecko DOM Reference](#) at the Mozilla Developer Center. Look up each of the above methods in the Gecko DOM Reference. Note that some of the methods belong to `document`, and some belong to `element`.

1. Using JavaScript and the previously described methods, fill an empty page with various components, such as: a header, some text, and an image.
2. Assume you are writing a system that allows users to prototype web pages using your template and print them off. Previously you created pages by manually copying the template to a new file and adding the content (and title) with an editor. This of course requires access to the filesystem, which is not an option for end users.

Write methods that use [pop up dialogue boxes](#) to ask the user to enter:

- a. The page title
- b. The heading (of the content section)
- c. A paragraph of text
- d. The URL of a picture

The page should then be populated with this information.

Searching

Design a JS function (`addFinder`) that, when called, adds a textfield to the top of the page. When this textfield is updated, it should search all text elements of the given webpage, looking for matches to the textfield. Upon finding one, it should highlight the match using CSS.

Hint: Use span tags with a class.

RegEx: A better way of searching

Read [this tutorial](#), which gives a basic overview of Regular Expressions, and how to use them. Go through some of the exercises [here](#) to challenge yourself.

As an additional challenge, edit your addFinder to take RegEx expressions.

Optional for fun! Using the Canvas

There is a [good online tutorial](#) for using the canvas from W3Schools. Go through this tutorial and have a go at adding a canvas with an interesting design to your webpage.

Getting started with jQuery

Read the [W3Schools JQuery tutorial](#). Try using jQuery to rewrite some of your DOM manipulation code from earlier.