## Overview

In this activity, you will use conditionals to determine what content to display when a user clicks certain tabs.

Conditionals are a key element of JavaScript. The ability to run certain code if a condition is met is essential knowledge for any web designer-developer.

## Instructions

1. Open index.html, located in the .zip file you downloaded in preparation for class (in the Activity folder).
   * **Note:** You’ll see that there’s structure in place for two tabs, but there’s no interactivity when clicking between tabs (yet!).
2. Open index.js from the .zip file.

**Step 1: Write if-else Syntax**

1. You already have an event listener in index.js that listens for a click event in any <div> with a class of .tab. You also have the skeleton of an if statement in the file.
   * Take a moment to think about the following problem: There are two <div>s with a class of .tab. How can you know which one was clicked?
     + Within a click listener, the this keyword refers to the element that was clicked.
     + The this keyword is a tricky subject in JavaScript. For now, all you need to remember is that it refers to the event that was triggered by the click event. You will surely encounter more instances of this in your future career, so you can read more about this [here](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/this).
2. Add the beginning of your if statement within the click event listener:

| if () {  } |
| --- |

1. Next, add the else if statement:

| else if () {  } |
| --- |

1. Your if-else statement should look like the below:

| if () {  } else if () {  } |
| --- |

**Step 2: Add Conditionals**

1. Now that you have the syntax for your if-else statement, you can add the conditions that need to be checked and the code that runs when a conditional is true.
2. Add $(this).hasClass("tab1") to the beginning of your if-else statement.
   * **Tip!** Do you know what this conditional is checking for? It is checking whether the element that was clicked on (this) has a class of tab1.
3. Your code should now look like this:

| if ($(this).hasClass("tab1")) {  } |
| --- |

1. Plan for a user who clicks a <div> with a class of tab2 by targeting tab2 in the else if statement.
   * **Hint:** Your code should look pretty similar to the code block above.

**Step 3: Add Interactivity**

1. It’s time to add interactivity to our page. Add the following in the if block of your if...else statement:

| $(".tab1").addClass("active"); $(".tab2").removeClass("active"); |
| --- |

1. Take a moment to consider what this code will be doing to our HTML elements. Does what you think match the description below?
   * **If** the element clicked has the tab1 class, you are adding the active class to anything with the tab1 class and removing the active class from anything with the tab2 class.
2. Next, add the following within the if block of your if-else statement:

| $("#tabContent1").css("display", "block"); $("#tabContent2").css("display", "none"); |
| --- |

1. Once again, consider what this code is doing. Does what you think match the description below?
   * **If** the element clicked has the tab1 class, you are setting the CSS property display to block on an element with the ID tabContent1. Next, we set the CSS property display class to none on the element with the ID tabContent2.
2. Now, do the same for when tab2 is clicked—i.e., write code that will add interactivity to the else if statement. (**Hint:** It will look similar to the if statement.)
3. Save your changes and reload index.html.
4. Check out your code in action!