[Debugging Commands](https://youtu.be/68wO-sl5vXg):

1. Open dev tools: Ctrl+Shift+J/cmd + Option + i
2. Sources Tab - Where you debug JavaScript
3. [Breakpoint Types](https://developer.chrome.com/docs/devtools/javascript/breakpoints/#loc): Allows you to set:
   1. **Event Listener Breakpoints** - Allows you to pause the code when the event listener executes.
      1. In Dev tools
         1. Go to sources tab.
            1. Select the Javascript file.
         2. Go to Event Listener Breakpoints.
         3. Open the Mouse dropdown
            1. Click Checkbox: Dev Tools will pause whenever a click event executes.
         4. Now when clicking a button, the breakpoint will stop program execution.
         5. Click **resume script execution** icon to go to the next line of code: 
            1. **Resume Script Execution** - lets you execute your code line by line. Each click will send you to the next line in the code.
         6. Click the “**Step into next Function call**” - To inspect a function further.
            1. When positioned on a function call, causes the cursor to jump to the function definition.
         7. Click the “**Step out of current function**”  - When stepping through your code, if you land in an irrelevant part of the code. You can execute that function and return to the function that called it by clicking the “step out of the current function” button.
         8. Click “**Step over next function call**”  - The function executes to completion, and you will pause on the last line of the function that you are currently stepping through.
      2. Code Stepping Controls:
         1. Step into code
   2. [**Line of Code Breakpoint**](https://developer.chrome.com/docs/devtools/javascript/breakpoints/#loc) - Set a breakpoint at a specific Line of Code.
      1. Click the line number next to the code.
      2. Click **Resume script execution** - Dev tools reruns all the code up until that line, and pauses before the line executes.
      3. **Scope Section:**
         1. Allows you to view local and global variable values, while stepping through the program execution.
      4. **Watch Section**:
         1. Allows you to watch the values of variables as they change over time.
         2. Allows you to add any javascript expression here.
            1. Click add expression: 
            2. Type a variable name or expression and click Enter.
      5. Console Section:
         1. When paused on a line of code, the console has access to all the variables currently in scope.
         2. Type a variable name within the function’s scope and hit enter to get the value.
         3. Test any possible bug fixes in the Dev Tools console.
      6. Testing a bug Fix: Once you have a possible fix, by testing in the console.
         1. Click **resume script execution**
         2. Make the change in the javascript file, Click Ctrl+S to save
         3. Click **deactivate break points**, to run the program normally: 
         4. Click the button to see if you get the right result
   3. [DOM Change Breakpoint](https://developer.chrome.com/docs/devtools/javascript/breakpoints/#dom) - Dom Node Changes