



Intro to JavaScript Week 5 Coding Assignment

Points possible: 75

URL to Your GitHub Repository:

<https://github.com/nmgolz/week5.git>

Instructions: In VS Code, or an IDE of your choice, write the code that accomplishes the objectives listed below. Ensure that the code compiles and runs as directed. Take screenshots of the code and of the running program (make sure to get screenshots of all required functionality) and paste them in this document where instructed below. Create a new repository on GitHub for this week's assignments and push this document, with your JavaScript project code, to the repository. Add the URL for this week's repository to this document where instructed and submit this document to your instructor when complete.

Coding Steps:

1. Create a menu app as seen in this week's video. What you create is up to you as long as it meets the following requirements.
 - a. Use at least one array.
 - b. Use at least two classes.
 - c. Your menu should have the options to create, view, and delete elements.



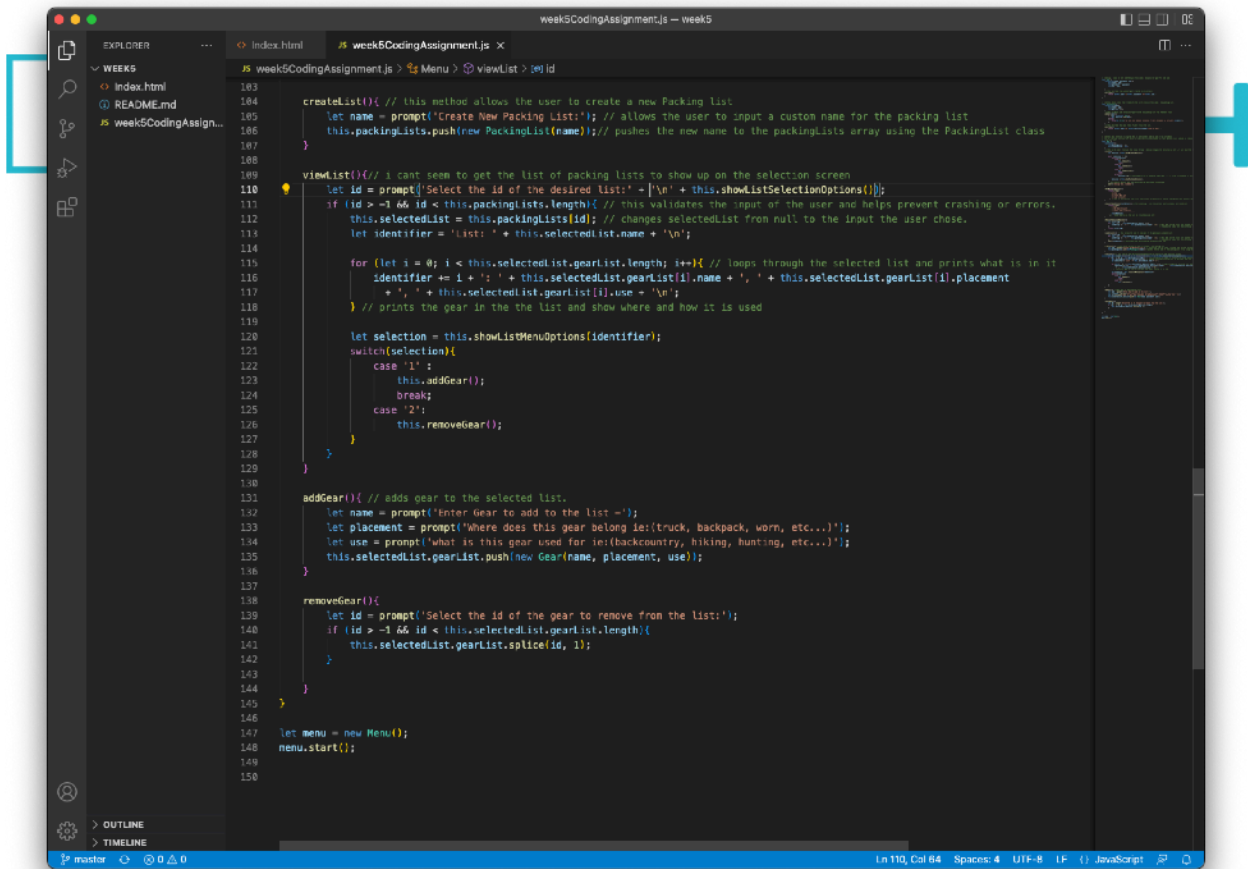
Screenshots of Code:

The first screenshot shows the initial class definitions and the start of the menu function in `week5CodingAssignment.js`.

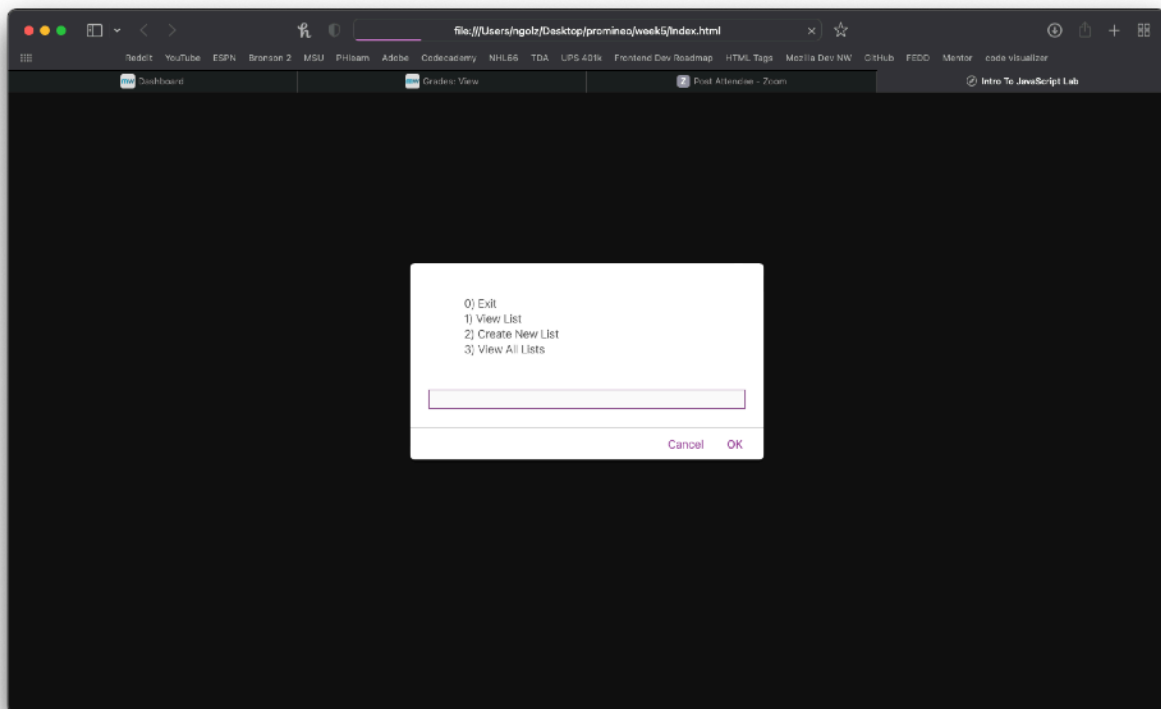
```
1 // creating a class to hold something in this case a new piece of gear for some use.
2 class Gear {
3   constructor(name, placement, use) {
4     this.name = name;
5     this.placement = placement;
6     this.use = use;
7   }
8   // this describes the piece of gear created in an instance
9   describe() {
10    return `${this.name} is ${this.placement} for ${this.use}`;
11  }
12 }
13
14 // create a second class that relates to the first class in this case a new packing list.
15 class PackingList {
16   constructor(name) {
17     this.name = name;
18     this.gearList = [];
19   }
20   // this is adding and instance of gear to the new packing list from the Gear class
21   addGear(gear) {
22     if (gear instanceof Gear) {
23       this.gearList.push(gear);
24     } else {
25       throw new Error('You can only add and instance of Gear. Argument is not gear: ${gear}');
26     }
27   }
28   // this describes how many items of gear are in the list.
29   describe() {
30     return `${this.name} has ${this.gearList.length} items of gear.`;
31   }
32 }
33
34 // create a menu function or program that is interactable and has one or two variables.
35 // this is the menu function that holds the code that is interactable to create packing lists and add or remove gear
36 // from said list
37 class Menu {
38   constructor() {
39     this.packingLists = [];
40     this.selectedList = null;
41   }
42   // this is the start function that loops through a menu as long as the selection is not 0, 0 will exit the menu
43   start() {
44     let selection = this.showMainMenuOptions();
45     while (selection != 0) {
46       switch(selection) {
47         case '1':
48           this.viewList();
49         case '2':
50           this.createList();
51         case '3':
52           this.viewAllLists();
53         default:
54           selection = 0; // this makes it so if anything other than 1, 2, 3, or 4 are selected it will exit the menu
55       }
56       selection = this.showMainMenuOptions();
57     }
58     // this alerts the use that they have left the menu after selecting 0
59     alert('Leaving Menu. Goodbye!');
60   }
61   showMainMenuOptions() {
62     return prompt('
63     0) Exit
64     1) View List
65     2) Create New List
66     3) View All Lists
67   '); // this is the function that first starts when the menu starts showing the options that the user can select
68   }
69   showListMenuOptions(listDetails) { // after selecting a list this allows you to interact with said list
70     return prompt('
71     0: Exit
72     1: Add Gear to List
73     2: Remove Gear From List
74   ');
75     // shows what is in the list we are interacting with
76     console.log(listDetails);
77   }
78   showListSelectionOptions() {
79     let listString = '';
80     for (let i = 0; i < this.packingLists.length; i++) {
81       listString += i + ': ' + this.packingLists[i].name + '\n'; // this adds all the lists into another list and numbers
82     } // then by the index into the listString variable
83     return listString;
84   }
85   viewAllLists() { // this allows the user to view all of the previously created lists
86     let listString = '';
87     for (let i = 0; i < this.packingLists.length; i++) {
88       listString += i + ': ' + this.packingLists[i].name + '\n'; // this adds all the lists into another list and numbers
89     } // then by the index into the listString variable
90     alert(listString); // this shows the new listString variable to the user.
91   }
92 }
```

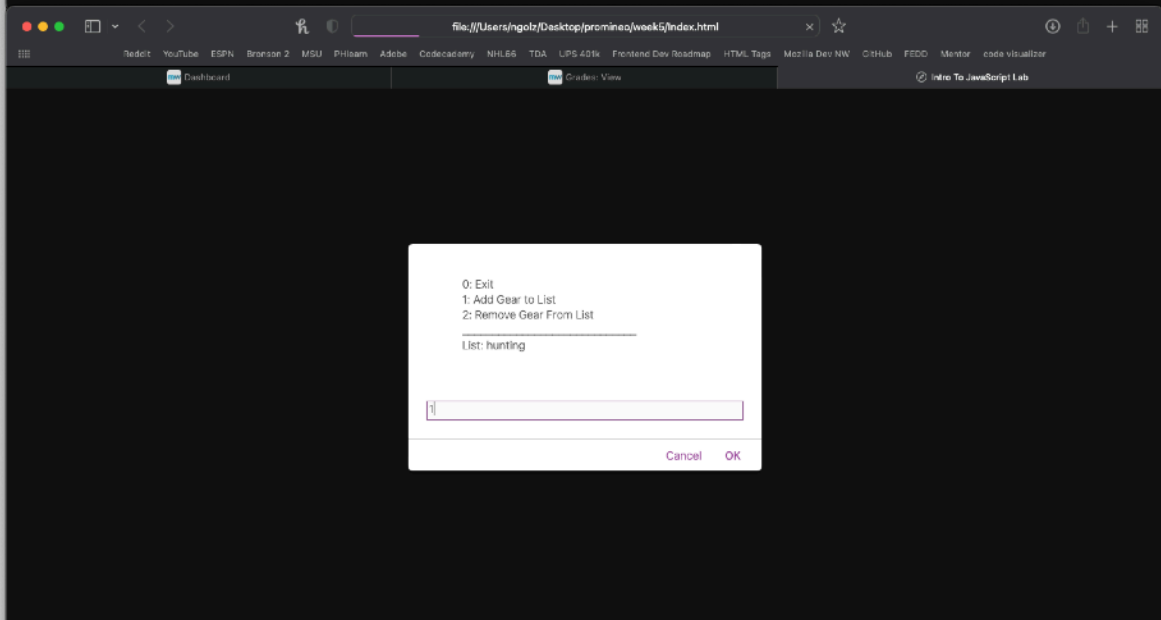
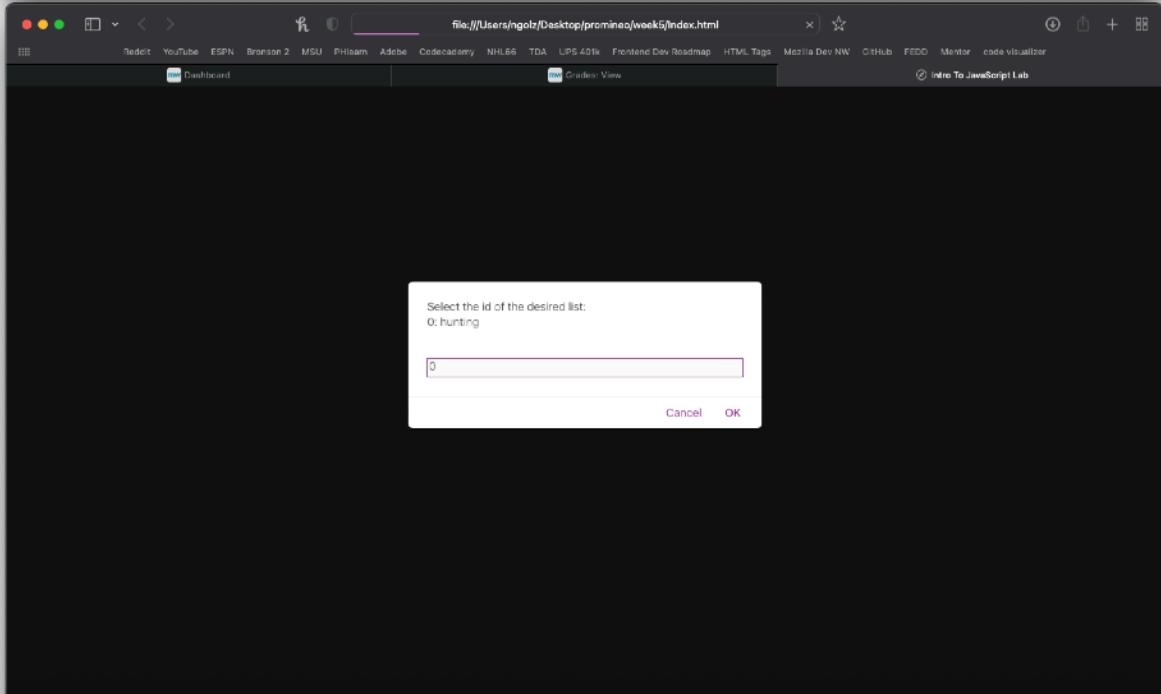
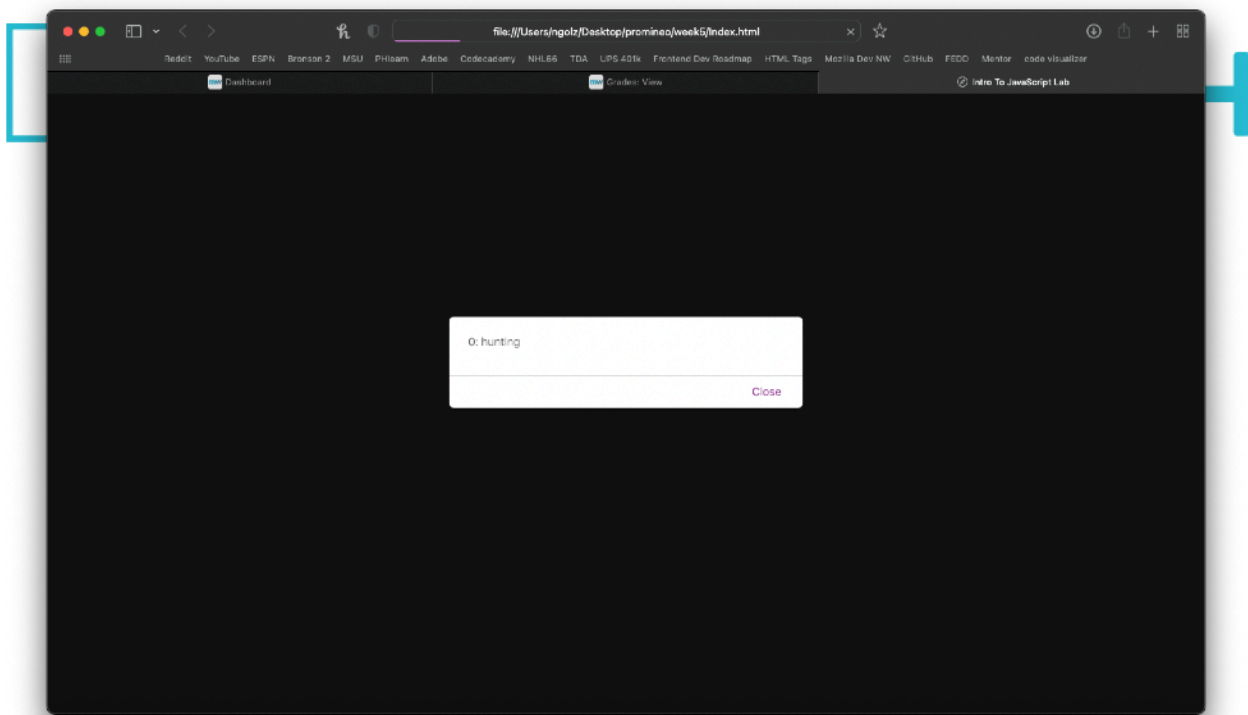
The second screenshot continues the code from the first, showing the implementation of the menu options.

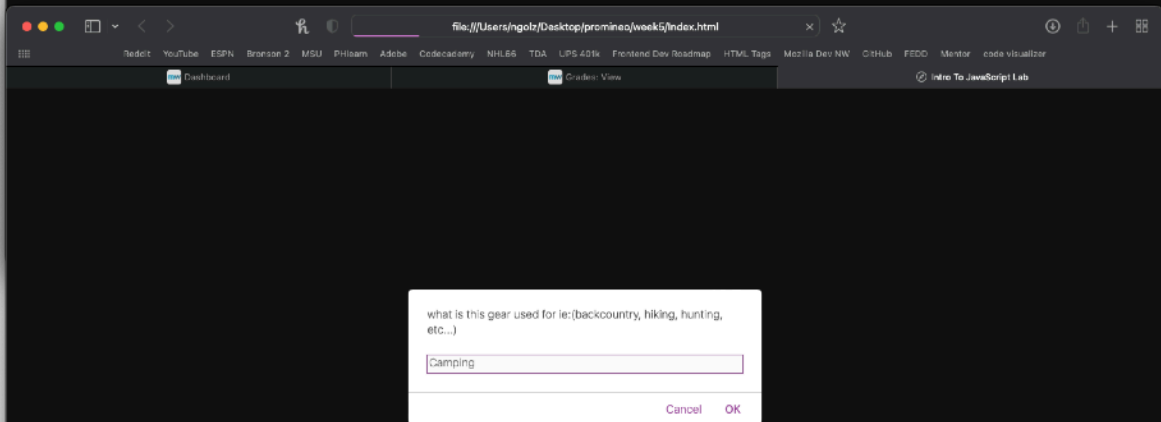
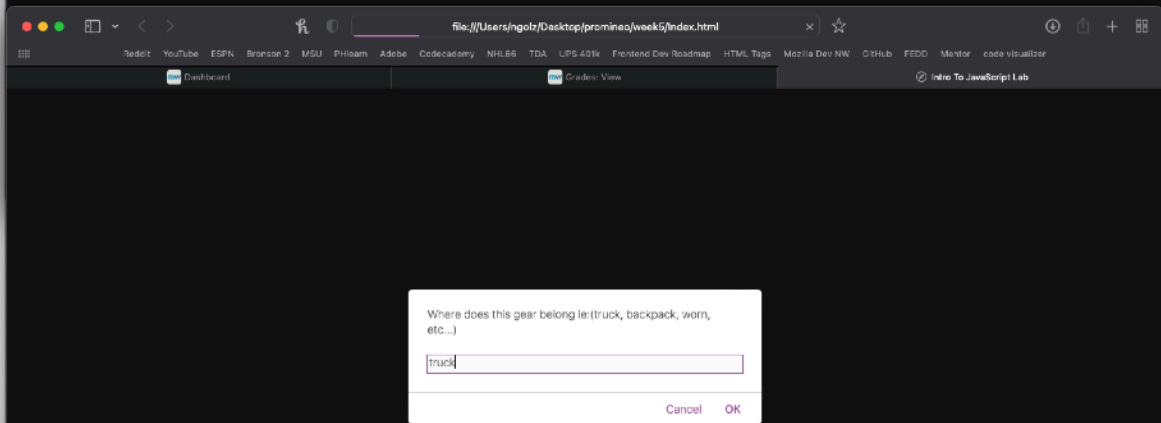
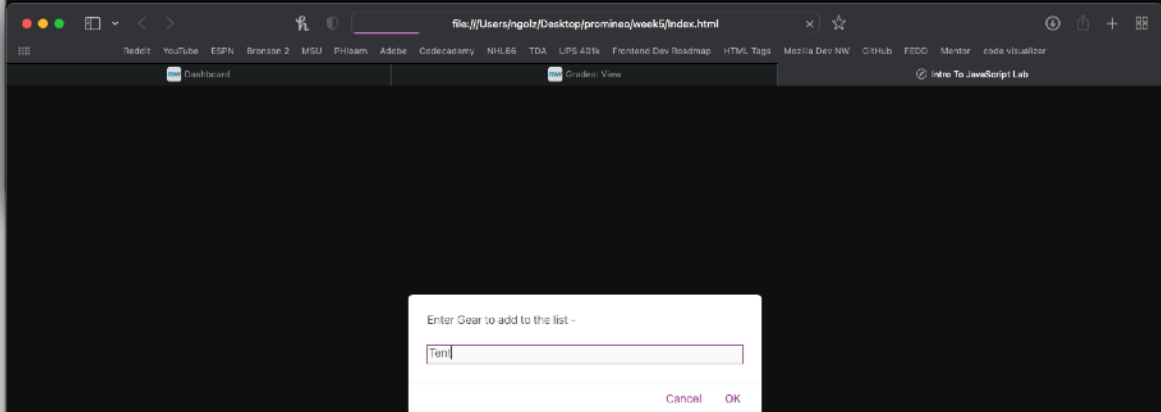
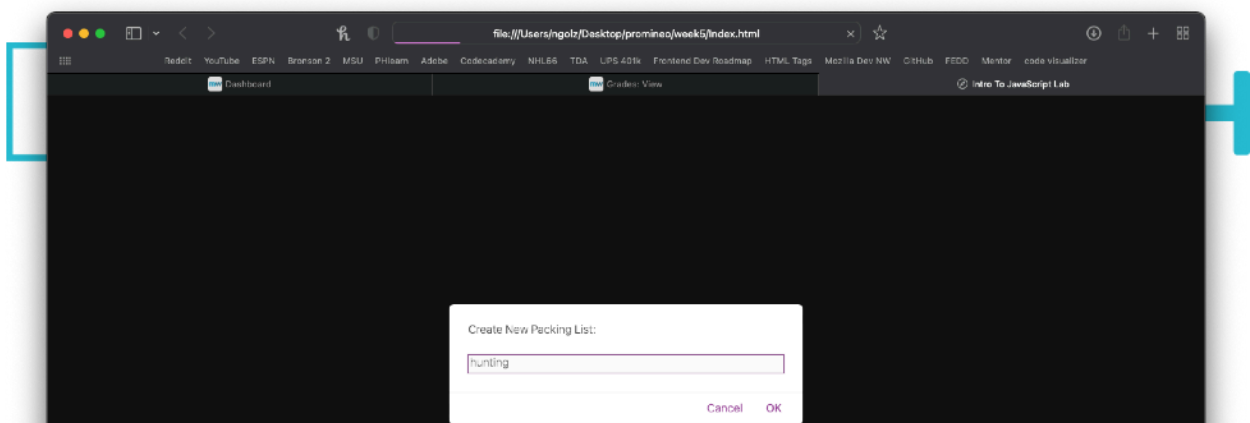
```
51 // this makes it so if anything other than 1, 2, 3, or 4 are selected it will exit the menu
52 }
53 // this alerts the use that they have left the menu after selecting 0
54 alert('Leaving Menu. Goodbye!');
55
56 showMainMenuOptions() {
57   return prompt('
58   0) Exit
59   1) View List
60   2) Create New List
61   3) View All Lists
62 '); // this is the function that first starts when the menu starts showing the options that the user can select
63 }
64
65 showListMenuOptions(listDetails) { // after selecting a list this allows you to interact with said list
66   return prompt('
67   0: Exit
68   1: Add Gear to List
69   2: Remove Gear From List
70 ');
71   // shows what is in the list we are interacting with
72   console.log(listDetails);
73 }
74
75 showListSelectionOptions() {
76   let listString = '';
77   for (let i = 0; i < this.packingLists.length; i++) {
78     listString += i + ': ' + this.packingLists[i].name + '\n'; // this adds all the lists into another list and numbers
79   } // then by the index into the listString variable
80   return listString;
81 }
82
83 viewAllLists() { // this allows the user to view all of the previously created lists
84   let listString = '';
85   for (let i = 0; i < this.packingLists.length; i++) {
86     listString += i + ': ' + this.packingLists[i].name + '\n'; // this adds all the lists into another list and numbers
87   } // then by the index into the listString variable
88   alert(listString); // this shows the new listString variable to the user.
89 }
90 }
```



Screenshots of Running Application:

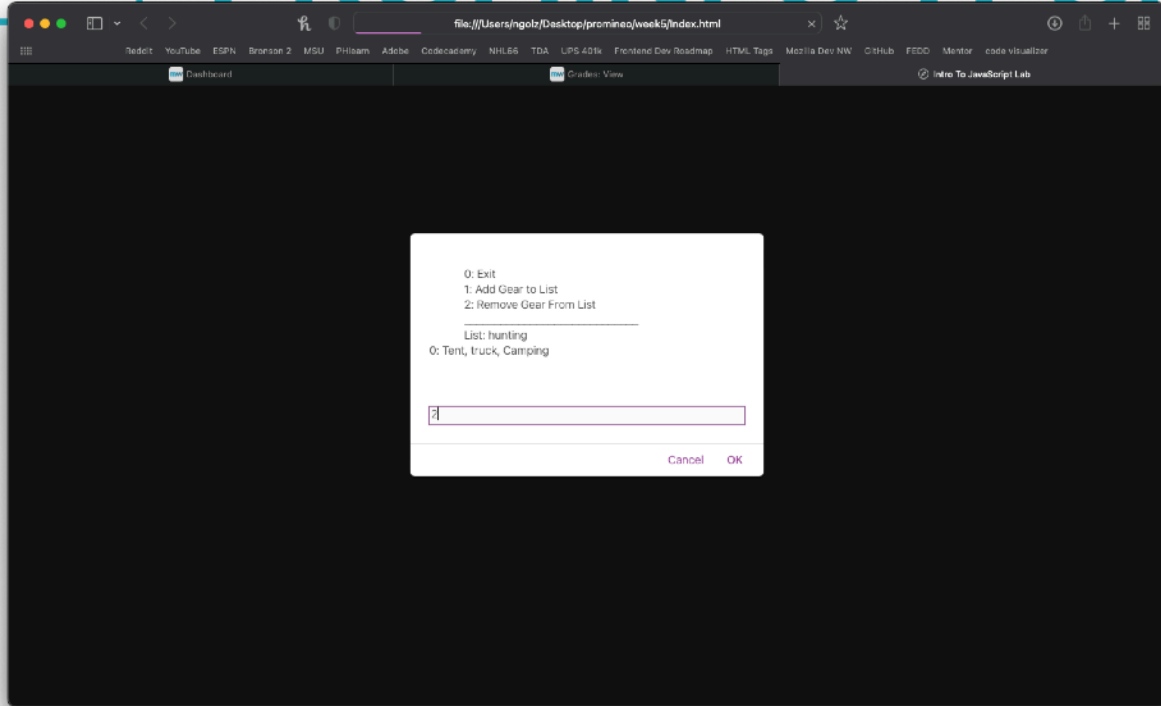








PROMINFO TECH





PROMINEO TECH



PROMINEO TECH