



## Intro to JavaScript Week 5 Coding Assignment

**Points possible:** 75

**URL to Your GitHub Repository:**

**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.

**GitHub URL:**

<https://github.com/acasia7/week5.git>

**Screenshots of Code:**



# PROMINEO TECH

```
JS index.js U X index.html U
Coding Assign Wk5 > JS index.js > Author > addBook
1 class Book { // created a class named Book
2   constructor(name, year) { // class will take the book name and year written
3     this.name = name;
4     this.year = year;
5   }
6   describe() {
7     return `${this.book} was written ${this.year}.`; // class will return the entered book name and year written
8   }
9 }
10
11 class Author { // created a class named Author
12   constructor(name) { //class will take the name of book author
13     this.name = name;
14     this.books = [];
15   }
16
17   addBook(book) { // method to take a book
18     if (book instanceof Book) { // check to see if a book is an instance of our book class
19       this.books.push(book); // push a new book to the book array
20     } else {
21       throw new Error('You can only add an instance of Book. Argument is not a book: ${book}');
22     }
23   }
24
25   describe() {
26     return `${this.name} has ${this.books.length} books.` // print out the name of author and how many of their books we have
27   }
28 }
29
30 class Menu {
31   constructor() {
32     this.authors = []; // array to collect multiple authors
33     this.selectedAuthor = null; // show a selected author from array
34   }
35
36   start () {
37     let selection = this.showMainMenuOptions(); // return the selection the user gives us
38
39     while (selection != 0) { // get users input
40
41       switch (selection) { // display different selections for the user
42         case '1':
43           this.createAuthor();
44           break;
45         case '2':
46           this.viewAuthor();
47           break;
48         case '3':
49           this.deleteAuthor();
50           break;
51         case '4':
52           this.displayAuthors();
53           break;
54         default:
55           selection = 0;
56       }
57       selection = this.showMainMenuOptions();
58     }
59
60     alert('Adios!');
61   }
62
63   showMainMenuOptions() {
64     return prompt(
65       0) exit
66       1) create new author
67       2) view author
68       3) delete author
69       4) display all authors
70     `);
71   }
72
73   showAuthorMenuOptions (authorInfo) {
74     return prompt(
75       0) back
76       1) create book
77       2) delete book
78     `${authorInfo}
79     `);
80   }
81 }
```



# PROMINEO TECH

JS index.js U X index.html U

Coding Assign Wk5 > JS index.js > Author > addBook

```
79     `);
80   }
81
82   displayAuthors() {
83     let authorString = ''; // create a blank string
84     for (let i= 0; i < this.authors.length; i++) { // iterate through authors
85       authorString += i + ' ' + this.authors[i].name + '\n'; // get the name of each author numbered on a new line
86     }
87     alert (authorString); // see all the authors
88   }
89
90   createAuthor () { // method to create author
91     let name = prompt ('Enter name for new author:'); // prompt the user to enter the name of an author
92     this.authors.push(new Author(name)); // push new author to authors array
93   }
94
95   viewAuthor() { // method to view author
96     let index = prompt('Enter the index of the author you wish to view:'); // prompt user to enter index to view selected author
97     if (index > -1 && index < this.authors.length) {
98       this.selectedAuthor = this.authors[index]; // set selected author class property to the author input by user
99       let description = 'Author:' + this.selectedAuthor.name + '\n';
100
101       for (let i = 0; i < this.selectedAuthor.books.length; i++) { // for loop to iterate through the selected authors books array
102         description += i + ' ' + this.selectedAuthor.books[i].name + ' - ' + this.selectedAuthor.books[i].year + '\n';
103       }
104
105       let selection = this.showAuthorMenuOptions(description) // pass in description to display author and show options for author
106       switch (selection) { // created options under author - create or delete book
107         case '1':
108           this.createBook();
109           break;
110         case '2':
111           this.deleteBook();
112       }
113     }
114   }
115
116   deleteAuthor() { //method to delete author
117     let index = prompt('Enter the index of the author you want to delete:'); //prompt user to enter the index of author they want to delete
```

JS index.js U X index.html U

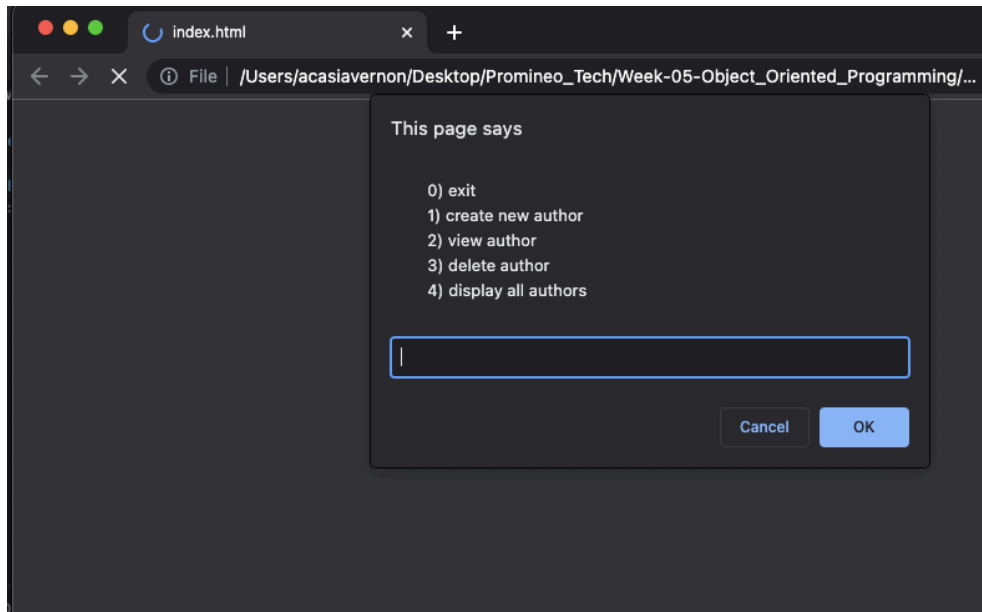
Coding Assign Wk5 > JS index.js > Author > addBook

```
112   }
113 }
114 }
115
116 deleteAuthor() { //method to delete author
117   let index = prompt('Enter the index of the author you want to delete:'); //prompt user to enter the index of author they want to delete
118   if (index > -1 && index < this.authors.length) {
119     this.authors.splice(index, 1); //delete the author under entered index
120   }
121 }
122
123 createBook() { //method to create a book
124   let name = prompt('Enter name for new book:'); // prompt user to enter the name and year of book
125   let year = prompt('Enter year for new book:');
126   this.selectedAuthor.books.push(new Book(name, year)); // push the new book, with its name and year, to the selected author's books array
127 }
128
129 deleteBook() { // method to delete a book
130   let index = prompt('Enter the index of the book you want to delete:'); //prompt user to enter index of book they want to delete
131   if (index > -1 && index < this.selectedAuthor.books.length) {
132     this.selectedAuthor.books.splice(index, 1); // delete book under user entered index from the selected author's books array
133   }
134 }
135 }
136
137 let menu = new Menu ();
138 menu.start();
```

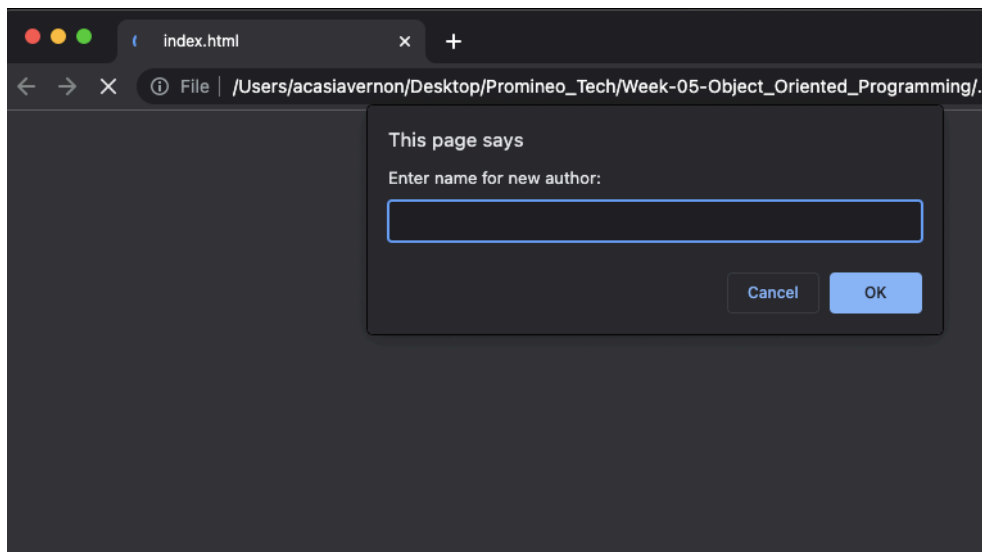


# PROMINEO TECH

## Screenshots of Running Application:



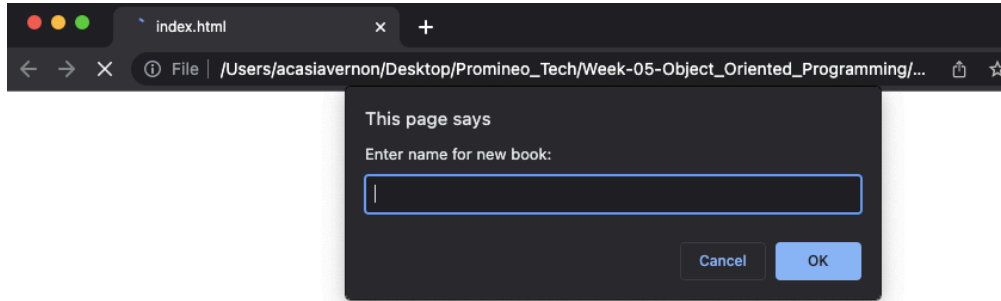
## Created an author:



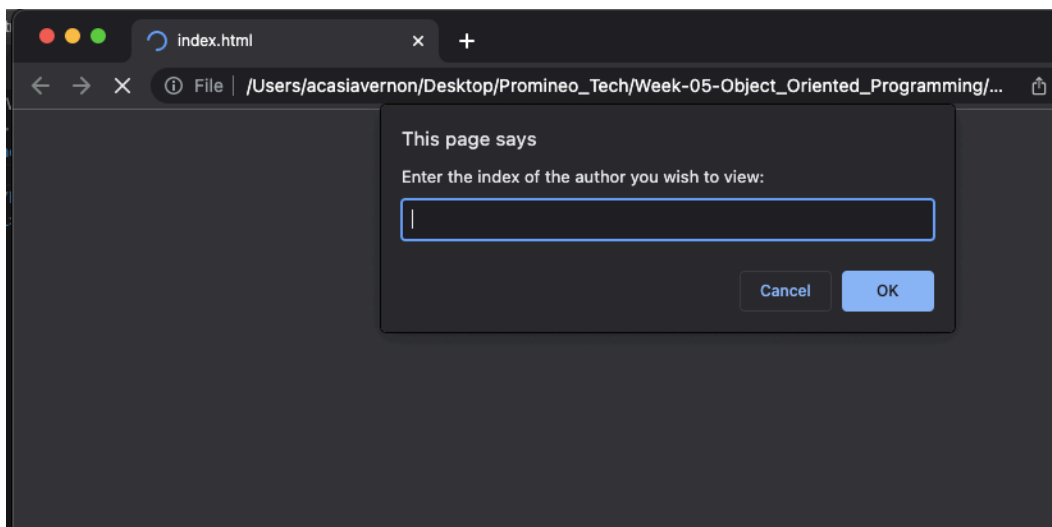


# PROMINEO TECH

## Created books

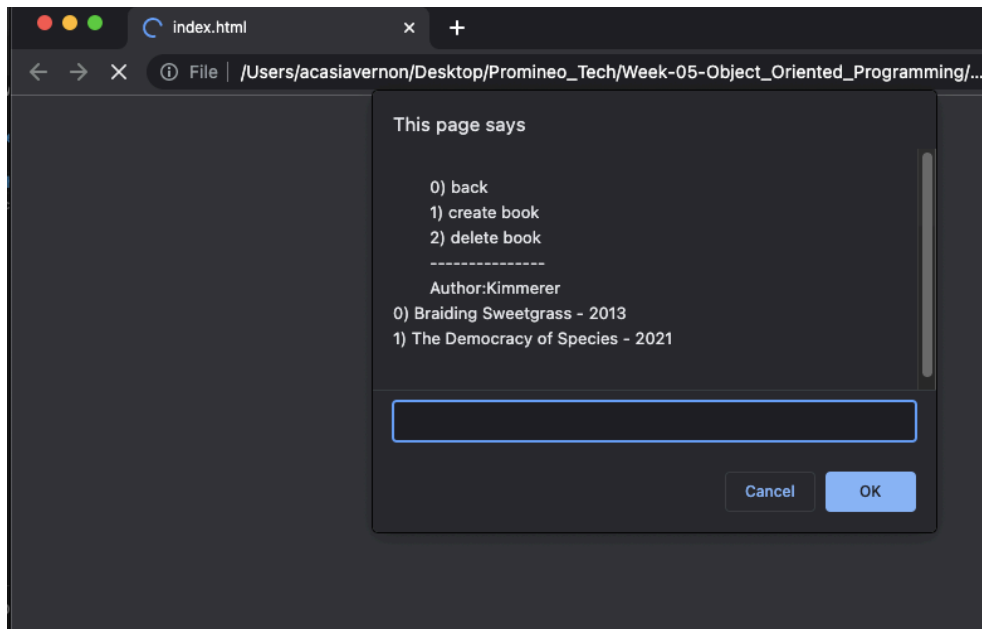


## View Author with books displayed

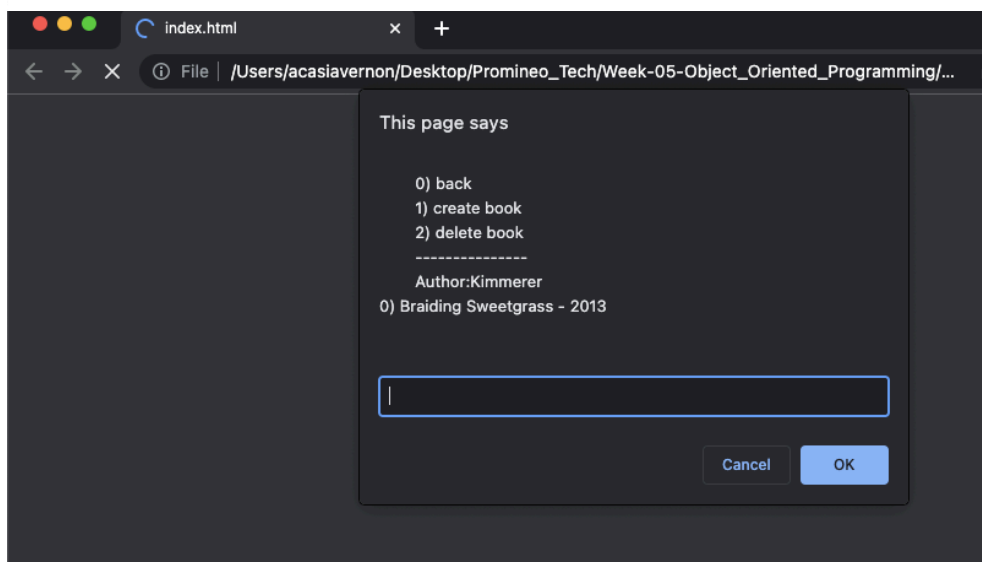
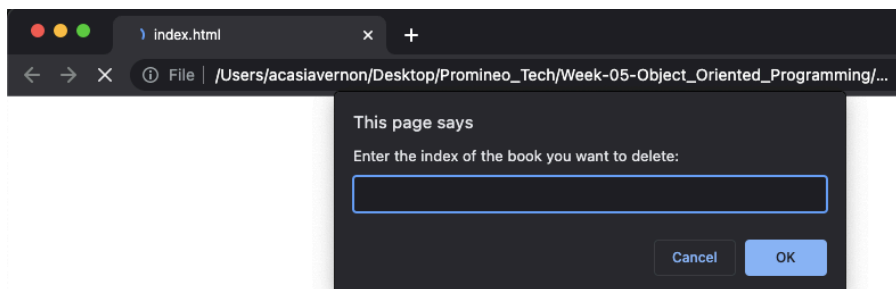




# PROMINEO TECH



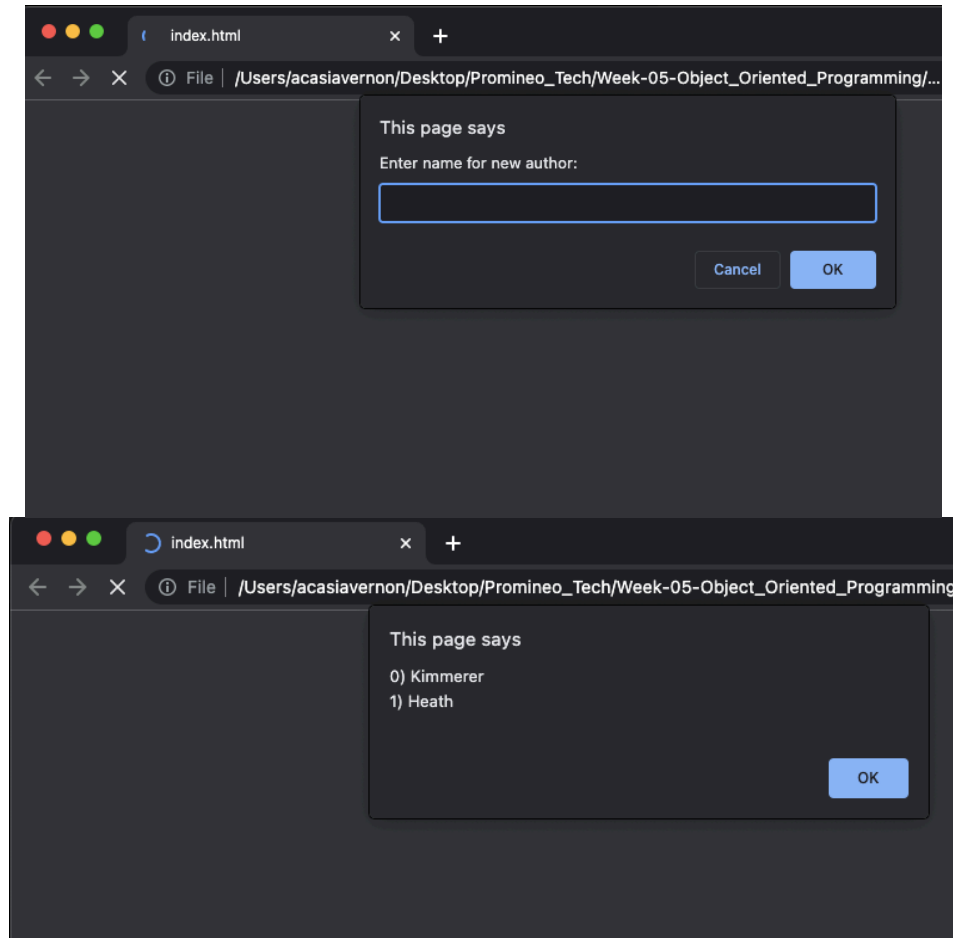
**Deleted a book:**





# PROMINEO TECH

**Added an author:**





# PROMINEO TECH

**Deleted an author:**

