

### Intro to JavaScript Week 5 Coding Assignment

Points possible: 70

Category	Criteria	% of Grade
Functionality	Does the code work?	25
Organization	Is the code clean and organized? Proper use of white space, syntax, and consistency are utilized. Names and comments are concise and clear.	25
Creativity	Student solved the problems presented in the assignment using creativity and out of the box thinking.	25
Completeness	All requirements of the assignment are complete.	25

**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.

URL to GitHub Repository: https://github.com/sw-dev-lisa-s-nh/JavaScript-Week5

### **Screenshots of Code:**

```
class Instrument { // Like a Player
         constructor(name, section) {
            this.instrumentName = name;
             console.log(`${this.instrumentName} is in the ${this.section} Section`)
         constructor(name) {
             this.name = name:
             this.instruments = [];
          addInstrument(instrument) {
             if (instrument instanceof Instrument) {
                  this.instruments.push(instrument);
                  This argument is not an instrument: ${instrument}`):
             console.log(`${this.name} plays ${this.instruments.length} instruments.`);
             this.musicians = []; // array of musicians
this.selectedMusician = null; // manage one musician at a time
              let selection = this.showMainMenuOptions():
              while (selection != 0) {
                  switch(selection) {
                          this.createMusician();
                      case '2' :
    this.viewMusician();
                          this.deleteMusician();
                          this.displayMusicians();
                   selection = this.showMainMenuOptions();
69
               alert('End of Musician Program!');
          showMainMenuOptions() {
               return prompt(`
              Musician Main Menu:
                  2) View a Musician3) Delete a Musician
                  4) Display all Musicians
```



## **PROMINEO TECH**

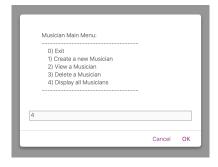
```
showMusicianMenuOptions(MusicianInfo) {
               Individual Musician Menu:
                   2) Delete an Instrument
           displayMusicians() {
               for (let i = 0; i < this.musicians.length; i++) {</pre>
                   musicianString += i+ ') ' + this.musicians[i].name + '\n';
103
104
               alert(musicianString);
               let name = prompt('Enter the name for new Musician: ');
this.musicians.push(new Musician(name));
110
111
          viewMusician() {
               let index = prompt("Enter the index of the Musician that you want to view:");
               if (index > -1 && index < this.musicians.length) {
                   this.selectedMusician = this.musicians[index];
                   let description = "\n Musician Name: " + this.selectedMusician.name + "\n";
                   for (let i = 0; i < this.selectedMusician.instruments.length; i++) {</pre>
                       let selection1 = this.showMusicianMenuOptions(description);
                   switch (selection1) {
                           this.createInstrument();
                           this.deleteInstrument();
           deleteMusician() {
               let index = prompt('Enter the index of the Musician that you wish to delete: ');
                if (index > -1 && index < this.musicians.length) {</pre>
                    this.musicians.splice(index,1);
                   throw new Error(`Index: ${index} is an invalid Musician's index!`);
               let name = prompt('Enter name for new instrument: ');
let section = prompt('Enter section for new instrument: ');
this.selectedMusician.instruments.push(new Instrument(name, section));
           deleteInstrument() {
               let index = prompt('Enter the index of the instrument that you wish to delete: ');
                if (index > -1 && index < this.selectedMusician.instruments.length) {
155
156
                   this.selectedMusician.instruments.splice(index,1);
               } else {
       let menu = new Menu();
       menu.start();
```



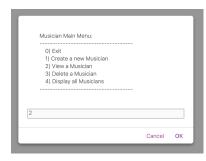
### **Screenshots of Running Application:**

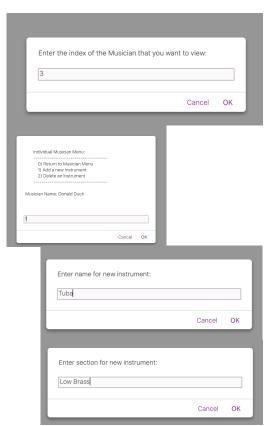














# PROMINEO TECH

