PROMINEO TECH

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.

Screenshots of Code:

```javascript
1    const prompt = require("prompt-sync")( { sigint: true});
2
3    class Country {
4        constructor(name, capital) {
5            this.name = name;
6            this.capital = capital;
7        }
8
9        describe() {
10            return `The capital of ${name} is ${capital}.`
11        }
12    }
13
14    class Continent {
15        constructor(name) {
16            this.name = name;
17            this.countries = [];
18        }
19
20
21    addCountry(country) {
22        if (country instanceof Country) {
23            this.countries.push(country);
24        } else {
25            throw new Error(`You can only add an instance of Country. Argument is not a Country: ${country}`);
26        }
27      }
28
29        describe () {
30        return `${this.countries.length} countries.`;
31      }
32    }
33
```

```javascript
class Menu {
    constructor() {
        this.continents = [];
        this.selectedContinent = null;
    }

    start() {
        let selection = this.showMainMenuOptions();
        while (selection != 0) {
            switch (selection) {
                case '1':
                    this.createContinent();
                    break;
                case '2':
                    this.viewContinent();
                    break;
                case '3':
                    this.deleteContinent();
                    break;
                case '4':
                    this.displayContinents();
                    break;
                default:
                    selection = 0;
            }
            selection = this.showMainMenuOptions();
        }

        alert('Bye!');
    }
```

```
64    showMainMenuOptions() {
65      return prompt(`
66    0) exit
67    1) create new continent
68    2) view continent
69    3) delete continent
70    4) display all continents
71      `);
72    }
73
74    showContinentMenuOptions(continentInfo) {
75      return prompt(`
76    0) back
77    1) create country
78    2) delete country
79    ---------------------
80    ${continentInfo}
81      `);
82    }
83
84    displayContinents() {
85      let continentString = '';
86      for (let i=0; i < this.continents.length; i++) {
87          continentString += i + ') ' + this.continents[i].name + '\n';
88      }
89      alert(continentString);
90    }
91    createContinent() {
92      let name = prompt('Enter name for new continent:');
93      this.continents.push(new Continent(name));
94    }
```

```javascript
 95
 96    viewContinent() {
 97      let index = prompt('Enter the index of the continent you wish to view:');
 98      if (index > -1 && index < this.continents.length) {
 99          this.selectedContinent = this.continents[index];
100          let description = 'Continent Name: ' + this.selectedContinent.name + '\n';
101
102          for(let i = 0; i < this.selectedContinent.countries.length; i++) {
103              description += i + ') ' + this.selectedContinent.countries[i].name
104              + ' - ' + this.selectedContinent.countries[i].capital + '\n';
105          }
106
107          let selection = this.showContinentMenuOptions(description);
108          switch (selection) {
109              case '1':
110                  this.createCountry();
111                  break;
112              case '2':
113                  this.deleteCountry();
114          }
115      }
116    }
117
118    deleteContinent() {
119      let index = prompt('Enter the index of the continent you wish to delete:');
120      if (index > -1 && index < this.continents.length) {
121          this.continents.splice(index, 1);
122      }
123    }
124
125    createCountry() {
126      let name = prompt('Enter name for new country:');
127      let capital = prompt('Enter capital for new country');
128      this.selectedContinent.countries.push(new Country(name, capital));
129    }
```

```
130
131     deleteCountry() {
132       let index = prompt('Enter the index of the country you with to delete:');
133       if(index > -1 && index < this.selectedContinent.countries.length) {
134           this.selectedContinent.countries.splice(index, 1);
135       }
136     }
137   }
138
139   let menu = new Menu();
140   menu.start();
141
```

Screenshots of Running Application:

```
PROBLEMS    OUTPUT    TERMINAL    DEBUG CONSOLE    JUPYTER

PS C:\Users\austi\Desktop> node week5.js

    0) exit
    1) create new continent
    2) view continent
    3) delete continent
    4) display all continents
▯
```

URL to GitHub Repository:

https://github.com/alexlaurencarey/week-5-coding-assignment.git