Intro to JavaScript Week 6 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 Visual Studio 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:

For the final project you will be creating an automated version of the classic card game *WAR*.

Think about how you would build this project and write your plan down. Consider classes such as Card, Deck, and Player and what fields and methods they might each have. You can implement the game however you'd like (i.e. printing to the console, using alert, or some other way). The completed project should, when ran, do the following:

- Deal 26 Cards to two Players from a Deck.
- Iterate through the turns where each Player plays a Card
- The Player who played the higher card is awarded a point
    - Ties result in zero points for either Player
- After all cards have been played, display the score.

Write a Unit Test using Mocha and Chai for at least one of the functions you write.

PROMINEO TECH

Screenshots of Code

```
8   const suits = ["diamond", "spade", "heart", "club"];
9   const values = ["2", "3", "4", "5", "6", "7", "8", "9", "10", "J", "Q", "K", "A"];
10  const rank = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13];
11
12  class Card {
13      constructor(suit, value, rank){
14          this.suit = suit;
15          this.value = value;
16          this.rank = rank;
17      }
18  }
19
20
21  class Deck{
22      constructor(suit, value, rank){
23          this.suit = suit;
24          this.value = value;
25          this.rank = rank;
26          this.deck = [];
27      }
28
29      createDeck(){
30          for (let i = 0; i < this.suit.length; i++) {
31              for (let j = 0; j < this.value.length; j++) {
32                  let card = { suit: this.suit[i], value: this.value[j], rank: this.rank[j] };
33                  this.deck.push(card);
34              }
35          }
36          return this.deck;
37      }
38
39      shuffle(){
40          for (let i = this.deck.length - 1; i > 0; i--) {
41              const newIndex = Math.floor(Math.random() * (i + 1));
42              const oldValue = this.deck[newIndex];
43              this.deck[newIndex] = this.deck[i];
44              this.deck[i] = oldValue;
45          }
```

PROMINEO TECH

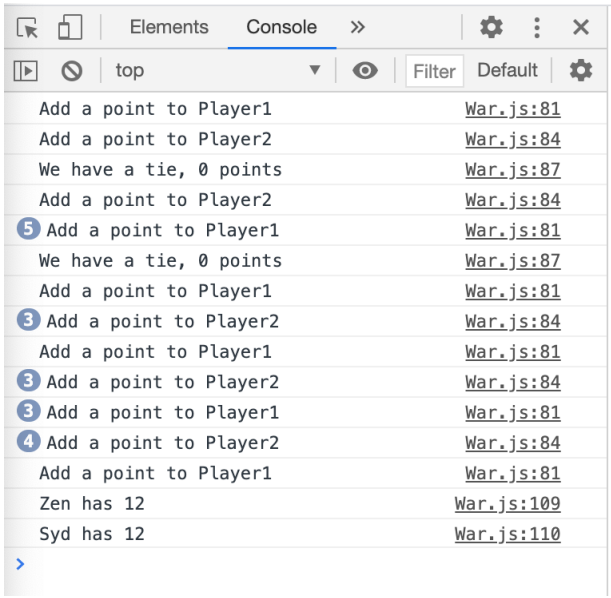
```
47
48      deal(){
49          const halfDeck = (this.deck.length / 2);
50          player1Hand = (this.deck.slice(0, halfDeck));
51          player2Hand = (this.deck.slice(halfDeck, this.deck.length));
52      }
53
54      describe(){
55          console.log("Number of cards in deck: " + this.deck.length);
56      }
57  }
58
59
60  class Player {
61      constructor(name){
62          this.name = name;
63          this.hand = [];
64          this.score = 0;
65      }
66
67      newHand(cards){
68          this.hand = cards;
69      }
70
71      describe(){
72          console.log(this.name + " has " + this.hand.length + " cards in her hand.");
73      }
74
75  }
76
```

PROMINEO TECH

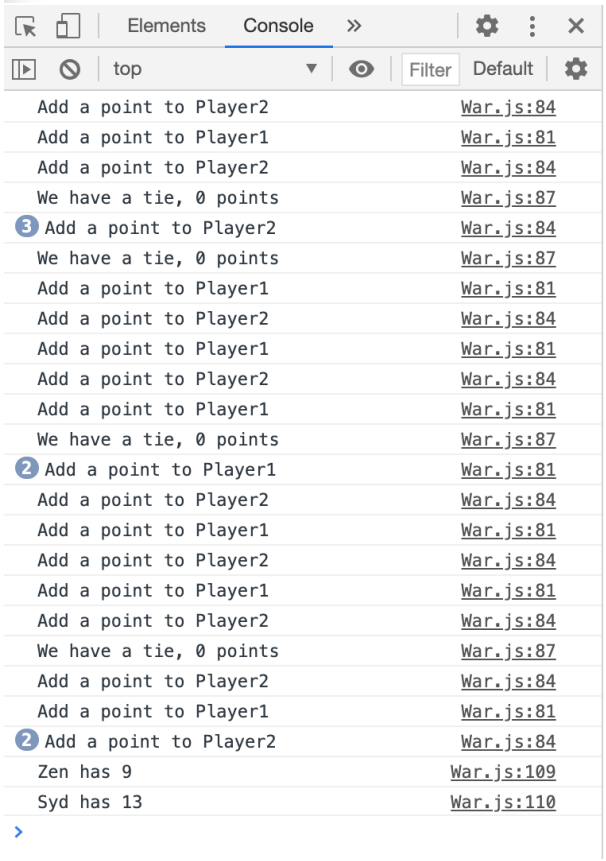
```
78  function compareCards(player1, player2){
79      for(let i = 0; i < player1Hand.length; i++){
80          if(player1.hand[i].rank > player2.hand[i].rank){
81              console.log(`Add a point to Player1`);
82              player1.score++;
83          } else if(player1.hand[i].rank < player2.hand[i].rank){
84              console.log(`Add a point to Player2`);
85              player2.score++;
86          } else {
87              console.log(`We have a tie, 0 points`);
88          }
89      }
90  }
91
92
93
94  let player1Hand = [];
95  let player2Hand = [];
96
97  let myWarDeck = new Deck(suits, values, rank);
98  myWarDeck.createDeck();
99  myWarDeck.shuffle();
100 myWarDeck.deal();
101
102 let player1 = new Player("Zen");
103 let player2 = new Player("Syd");
104 player1.newHand(player1Hand);
105 player2.newHand(player2Hand);
106
107
108 compareCards(player1, player2);
109 console.log("Zen has " + player1.score);
110 console.log("Syd has " + player2.score);
111
112
```

PROMINEO TECH

Screenshots of Running Application:



War Game - Assignment 6



War Game - Assignment 6

PROMINEO TECH

Assignment 6 > JS War Unit Testing.js > ...

```
var expect = chai.expect;       // imports

describe("MyFunctions", function() {


    describe('players', function () {
        it('assigns a name', function () {
            let player1 = new Player("Zen");
            expect(player1.name).to.equal('Zen');
        });
        it('assigns a hand', function () {
            let zen = new Player("Zen");
            const cards = 'test';
            expect(Array.isArray(player1.hand)).to.be.true;
            expect(player1.hand).to.have.length(26);
            // expect(zen.hand).to.deep.equal([]);
            player1.newHand(cards);
            expect(player1.hand).equals('test');
        });
    });
});
```

PROMINEO TECH

Index

MyFunctions

players

✓ assigns a name

```
let player1 = new Player("Zen");
expect(player1.name).to.equal('Zen');
```

✓ assigns a hand

```
let zen = new Player("Zen");
const cards = 'test';
expect(Array.isArray(player1.hand)).to.be.true;
expect(player1.hand).to.have.length(26);
// expect(zen.hand).to.deep.equal([]);
player1.newHand(cards);
expect(player1.hand).equals('test');
```

URL to GitHub Repository:

https://github.com/zenidev/Coding-Assignment-Week-6.git