



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
 - o 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:

```
File Edit Selection View Go Run Terminal Help
GameOfWar.js - Week6 Coding Assignment - Visual Studio Code

EXPLORER
  GameOfWar.html
  GameOfWar.js
  GameOfWar.test.js
  tests.html
  WEBSITE CODES
  node_modules
  GameOfWar.html
  GameOfWar.js
  GameOfWar.test.js
  package-lock.json
  package.json
  tests.html

OUTLINE
TIMELINE
NPM SCRIPTS

GameOfWar.js
1 alert('Start!');
2
3 class Card { // class to create cards
4   constructor(name, cardValue, cardSuit) {
5     this.name = name;
6     this.cardValue = cardValue;
7     this.cardSuit = cardSuit;
8   }
9 }
10
11 class Deck { //class to create a deck of 52 cards put into an array with a loop to randomize the elements in that array.
12   constructor() { // and another loop to distribute the first element to one array and the next element to another array.
13     this.cards = [];
14   }
15
16   createDeck() {
17     let names = ['Two', 'Three', 'Four', 'Five', 'Six', 'Seven', 'Eight', 'Nine', 'Ten', 'Jack', 'Queen', 'King', 'Ace'];
18     let cardSuits = ['Spades', 'Diamonds', 'Clubs', 'Hearts'];
19     let cardValues = [2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14];
20
21     for (let i = 0; i < cardSuits.length; i++) {
22       for (let x = 0; x < names.length; x++) {
23         this.cards.push(new Card(names[x], cardValues[x], cardSuits[i]));
24       }
25     }
26   }
27
28   shuffleDeck() {
29     for (let i = this.cards.length - 1; i >= 1; i--) {
30       let x = Math.floor(Math.random() * (i + 1));
31       let temp = this.cards[i];
32       this.cards[i] = this.cards[x];
33       this.cards[x] = temp;
34     }
35   }
36
37   dealDeck() {
38     for (let i = 0; i < 52; i++) {
39       let dealCard1 = this.cards.shift();
40       player1.playtand.push(dealCard1);
41       let dealCard2 = this.cards.shift();
42       player2.playtand.push(dealCard2);
43     }
44   }
45 }
46
47 class Player { // class to hold player names, with an array for player hand, and to start player score at zero
48   constructor(playerName) {
49     this.playerName = playerName;
50     this.playerHand = [];
51     this.playerScore = 0;
52   }
53
54   roundEnd() {
55     this.playerHand = [];
56     this.playerScore = 0;
57   }
58 }
```

```
File Edit Selection View Go Run Terminal Help
GameOfWar.js - Week6 Coding Assignment - Visual Studio Code

EXPLORER
  GameOfWar.html
  GameOfWar.js
  GameOfWar.test.js
  tests.html
  WEBSITE CODES
  node_modules
  GameOfWar.html
  GameOfWar.js
  GameOfWar.test.js
  package-lock.json
  package.json
  tests.html

OUTLINE
TIMELINE
NPM SCRIPTS

GameOfWar.js
59 //
60
61 const player1 = new Player('Player One');
62 const player2 = new Player('Player Two');
63
64 class Play { // class to initiate a new deck with a loop to increment rounds with randomized first element of playerhand array
65   start() { // and a if/else statement to increment player score by round
66     player1.roundEnd();
67     player2.roundEnd();
68     const myDeck = new Deck();
69     myDeck.createDeck();
70     myDeck.shuffleDeck();
71     myDeck.dealDeck();
72     this.rounds();
73   }
74
75   rounds() {
76     for (let round = 0; round < 10; round++) {
77       let playerHand1 = player1.playerHand.shift();
78       let playerHand2 = player2.playerHand.shift();
79       alert('Round ' + round + 1);
80       if (playerHand1.cardValue > playerHand2.cardValue) {
81         player1.playerScore++;
82         alert('The winner of round ' + round + 1 is ' + player1.playerName + '. 1 point awarded.');
83       } else if (playerHand2.cardValue > playerHand1.cardValue) {
84         player2.playerScore++;
85         alert('The winner of round ' + round + 1 is ' + player2.playerName + '. 1 point awarded.');
86       } else {
87         player1.playerScore++;
88         player2.playerScore++;
89         alert('The score remains ' + player1.playerName + ' ' + player1.playerScore + ' - ' + player2.playerName + ' ' + player2.playerScore);
90       }
91     }
92   }
93
94   scoring() { // if/else statement to display final score
95     if (player1.playerScore > player2.playerScore) {
96       alert('The final score for ' + player1.playerName + ' is ' + player1.playerScore);
97       alert('The final score for ' + player2.playerName + ' is ' + player2.playerScore);
98       if (player1.playerScore > player2.playerScore) {
99         alert('The final score for ' + player1.playerName + ' is ' + player1.playerScore);
100         alert('The final score for ' + player2.playerName + ' is ' + player2.playerScore);
101         alert('The final score for ' + player1.playerName + ' is ' + player1.playerScore);
102         alert('The final score for ' + player2.playerName + ' is ' + player2.playerScore);
103         alert('The game is tied!');
104       }
105     }
106   }
107
108   let newPlay = new Play(); // instance to instantiate the start application method
109   newPlay.start();
110 }
```



PROMINEO TECH

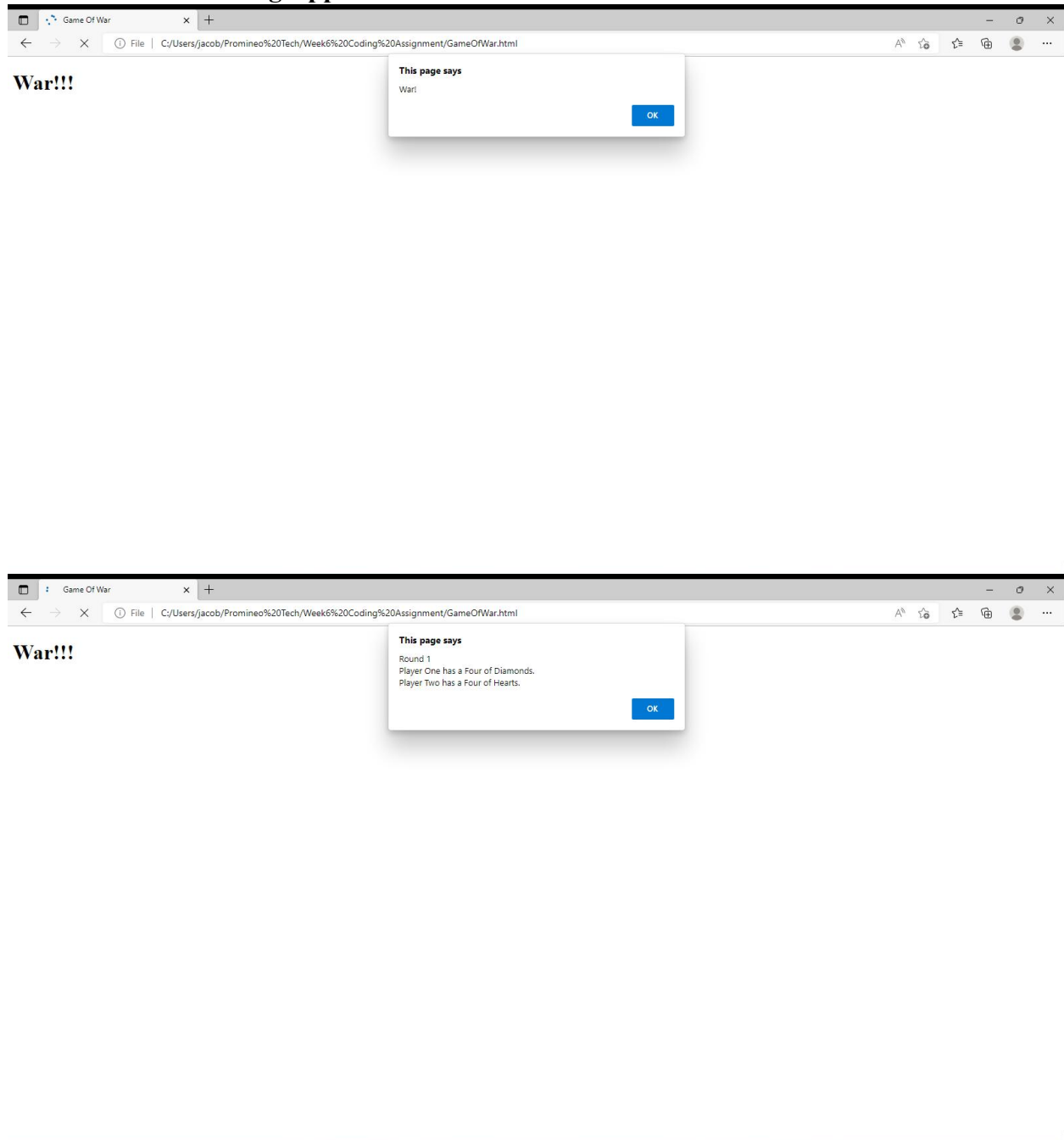
```
1 <!DOCTYPE html>
2 <html>
3
4   <head>
5     <link rel="stylesheet" href="node_modules/mocha/mocha.css">
6   </head>
7   <body>
8     <div id="mocha"><p><a href=".">Index</a></p></div>
9     <div id="messages"></div>
10    <div id="fixtures"></div>
11    <script src="node_modules/mocha/mocha.js"></script>
12    <script src="node_modules/chai/chai.js"></script>
13    <script src="GameOfWar.js"></script>
14    <script>mocha.setup('bdd')</script>
15    <script src="GameOfWar.test.js"></script>
16    <script>mocha.run();</script>
17  </body>
18 </html>
```

```
1 let expect = chai.expect;
2
3 describe('Player', function() {
4   describe('#playerScore', function() {
5     it('should start player score at zero', function() {
6       let x = new Player()
7       expect(x.playerScore).to.equal(0);
8     });
9
10    it('should throw an error if player score starts at any other number than zero', function() {
11      expect(function() {
12        x.playerScore(5);
13      }).to.throw(Error);
14    });
15  });
16 });
17
```



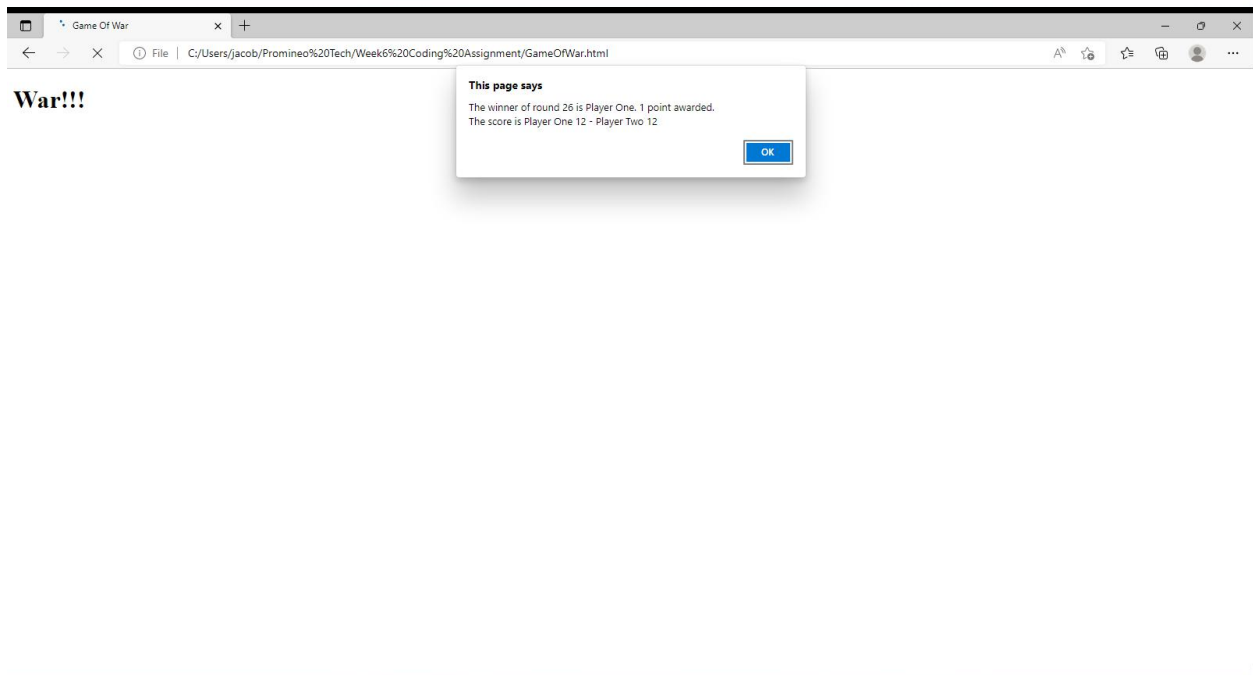
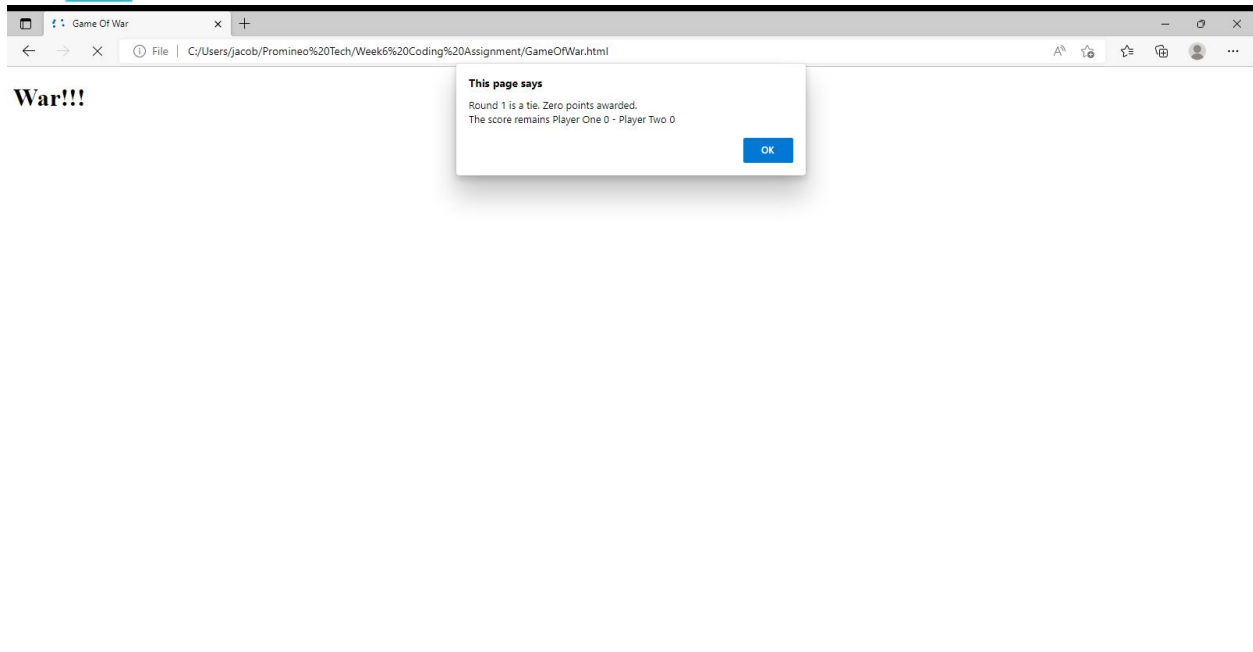
PROMINEO TECH

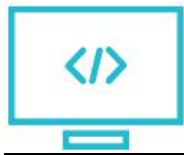
Screenshots of Running Application:



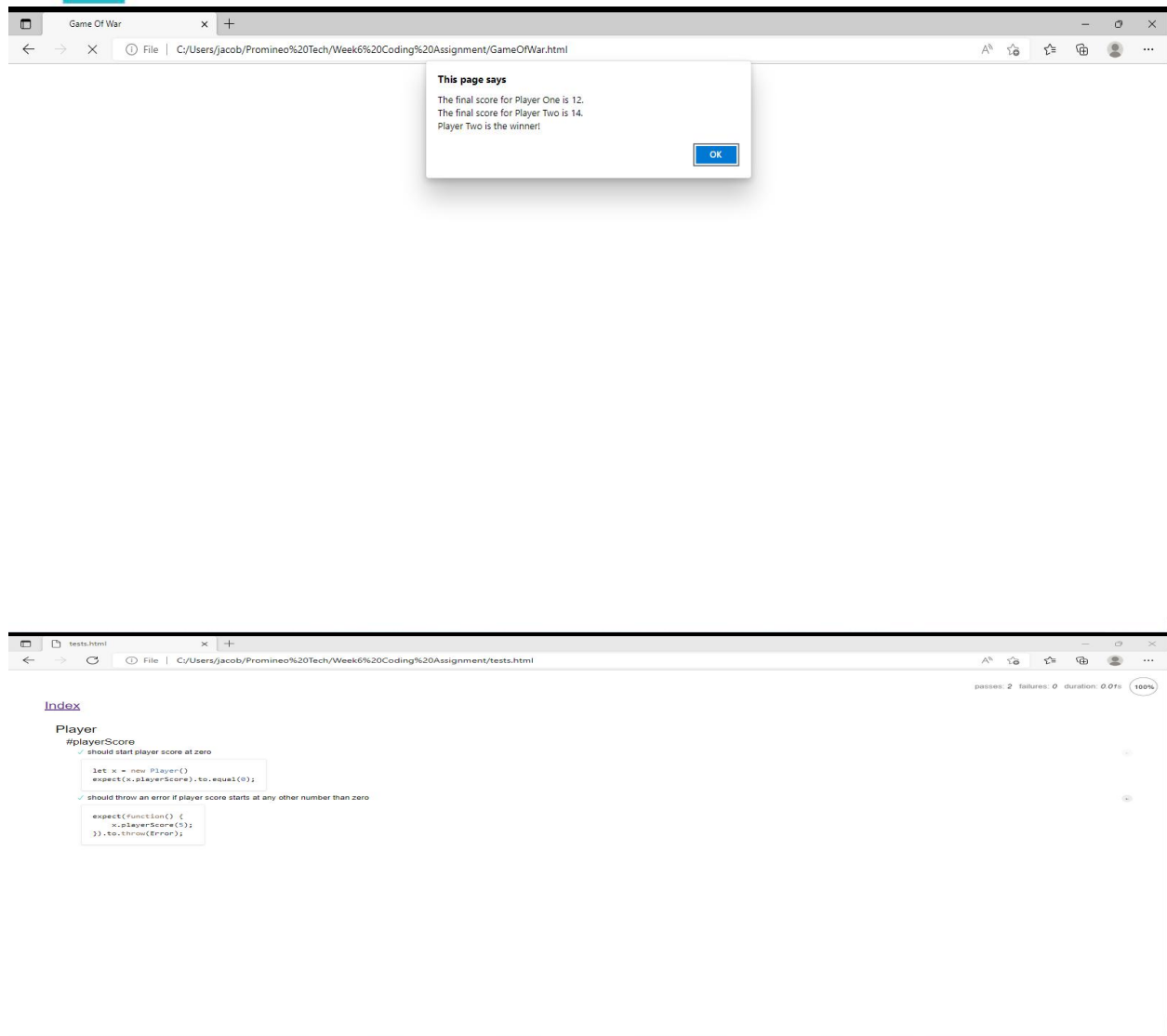


PROMINEO TECH





PROMINEO TECH



URL to GitHub Repository: <https://github.com/JacobStuder/Promineo-Tech-Week-6-Coding-Assignment.git>