



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:

```
<> war-game.html > html
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4    <meta charset="UTF-8">
5    <meta http-equiv="X-UA-Compatible" content="IE=edge">
6    <meta name="viewport" content="width=device-width, initial-scale=1.0">
7    <title>Document</title>
8  </head>
9  <body>
10   <script src="war-game.js"></script>
11 </body>
12 </html>
```

```
1  class Card {
2    constructor(suit, rank, value) {
3      this.suit = suit;
4      this.rank = rank;
5      this.value = value;
6    }
7  }
8  class Deck {
9    constructor() {
10     this.cards = [];
11   }
12   createDeck() {
13     let suits = ['clubs', 'diamonds', 'hearts', 'spades'];
14     let ranks = ['ace', '2', '3', '4', '5', '6', '7', '8', '9', '10', 'jack', 'queen', 'king'];
15     let values = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13];
16     for (let i = 0; i < suits.length; i++) {
17       for (let j = 0; j < ranks.length; j++) {
18         this.cards.push(new Card(suits[i], ranks[j], values[j]));
19       }
20     }
21   }
22 }
```

```
22   shuffleDeck() {
23     for (let i = this.cards.length - 1; i > 0; i--) {
24       let j = Math.floor(Math.random() * (i + 1));
25       let temp = this.cards[i];
26       this.cards[i] = this.cards[j];
27       this.cards[j] = temp;
28     }
29   }
30 }
31 }
32
33 const d = new Deck();
34
35 d.createDeck();
36 d.shuffleDeck();
```



PROMINEO TECH

```
40 class Player {
41   constructor(name) {
42     this.playerName = name;
43     this.playerCards = [];
44     this.playerScore = 0
45   }
46 }
47
48 class Board {
49   constructor() {
50     this.cardsInMiddle = [];
51     this.players = [];
52   }
53   start(player1, player2) {
54     this.players.push(new Player(player1));
55     this.players.push(new Player(player2));
56     (method) Deck.createDeck(): void
57     d.createDeck();
58     d.shuffleDeck();
59     this.players[0].playerCards = d.cards.slice(0, 26);
60     this.players[1].playerCards = d.cards.slice(26, 52);
61   }
62 }
```

```
84 function roundOutput(player1, player2, roundNum) {
85   console.log(`${player1.playerName} plays: ${player1.playerCards[roundNum].value} of ${player1.playerCards[roundNum].suit}
86   `);
87   console.log(`${player2.playerName} plays: ${player2.playerCards[roundNum].value} of ${player2.playerCards[roundNum].suit}
88   `);
89 }
90
91 function playRoundResults(player1, player2) {
92   for (let i = 0; i < player1.playerCards.length; i++) {
93     roundOutput(player1, player2, i);
94     if (player1.playerCards[i].value > player2.playerCards[i].value) {
95       player1.playerScore += 1;
96       console.log(`${player1.playerName} WINS THIS ROUND!!!`);
97     } else if (player1.playerCards[i].value < player2.playerCards[i].value) {
98       player2.playerScore += 1;
99       console.log(`${player2.playerName} WINS THIS ROUND!!!`);
100     } else {
101       console.log("PLAYERS TIED!!! NO POINTS AWARDED.");
102     }
103   }
104 }
```

```
88 function finalTally(player1, player2) {
89   if (player1.playerScore > player2.playerScore) {
90     console.log(`CONGRATULATIONS ${player1.playerName} WINS THE GAME WITH A FINAL SCORE OF: ${player1.playerScore}!!!!`);
91   } else if (player1.playerScore < player2.playerScore) {
92     console.log(`CONGRATULATIONS ${player2.playerName} WINS THE GAME WITH A FINAL SCORE OF: ${player2.playerScore}!!!!`);
93   } else {
94     console.log(`WHAT ARE THE ODDS? ${player1.playerName} AND ${player2.playerName} HAVE A TIE!!!!`);
95   }
96 }
97 }
```



PROMINEO TECH

```
100 let gameBoard = new Board();
101 gameBoard.start('BONNIE', 'CLYDE');
102 playRoundResults(gameBoard.players[0], gameBoard.players[1]);
103 finalTally(gameBoard.players[0], gameBoard.players[1])
104 console.log(gameBoard.players);
```

```
<> index-test.html > html
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8   <link rel="stylesheet" href="node_modules/mocha/mocha.css">
9 </head>
10 <body>
11   <div id="mocha"><p><a href=".">Index</a></p></div>
12   <div id="messages"></div>
13   <div id="fixtures"></div>
14   <script src="node_modules/mocha/mocha.js"></script>
15   <script src="node_modules/chai/chai.js"></script>
16   <script>
17     mocha.setup('bdd');
18   </script>
19
20
21
22   <script src="war-game.js"></script>
23   <script src="index-test.js"></script>
24
25   <script>
26     mocha.run();
27   </script>
28 </body>
29 </html>
```



Screenshots of Running Application:

```

(52) [Card, Card, Card, Card, Card, Card, Card, Card, Card, Card, Card, Card, Card, Card, Card, Card, Card, Card, Card, Card, Card, Card, Card]
    war-game.js:25
    ▶ 0: Card {suit: 'clubs', rank: 'ace', value: 1}
    ▶ 1: Card {suit: 'clubs', rank: '2', value: 2}
    ▶ 2: Card {suit: 'clubs', rank: '3', value: 3}
    ▶ 3: Card {suit: 'clubs', rank: '4', value: 4}
    ▶ 4: Card {suit: 'clubs', rank: '5', value: 5}
    ▶ 5: Card {suit: 'clubs', rank: '6', value: 6}
    ▶ 6: Card {suit: 'clubs', rank: '7', value: 7}
    ▶ 7: Card {suit: 'clubs', rank: '8', value: 8}
    ▶ 8: Card {suit: 'clubs', rank: '9', value: 9}
    ▶ 9: Card {suit: 'clubs', rank: '10', value: 10}
    ▶ 10: Card {suit: 'clubs', rank: 'jack', value: 11}
    ▶ 11: Card {suit: 'clubs', rank: 'queen', value: 12}
    ▶ 12: Card {suit: 'clubs', rank: 'king', value: 13}
    ▶ 13: Card {suit: 'diamonds', rank: 'ace', value: 1}
    ▶ 14: Card {suit: 'diamonds', rank: '2', value: 2}
    ▶ 15: Card {suit: 'diamonds', rank: '3', value: 3}
    ▶ 16: Card {suit: 'diamonds', rank: '4', value: 4}
    ▶ 17: Card {suit: 'diamonds', rank: '5', value: 5}
    ▶ 18: Card {suit: 'diamonds', rank: '6', value: 6}
    ▶ 19: Card {suit: 'diamonds', rank: '7', value: 7}
    ▶ 20: Card {suit: 'diamonds', rank: '8', value: 8}
    ▶ 21: Card {suit: 'diamonds', rank: '9', value: 9}
    ▶ 22: Card {suit: 'diamonds', rank: '10', value: 10}
    ▶ 23: Card {suit: 'diamonds', rank: 'jack', value: 11}
    ▶ 24: Card {suit: 'diamonds', rank: 'queen', value: 12}
    ▶ 25: Card {suit: 'diamonds', rank: 'king', value: 13}
    ▶ 26: Card {suit: 'hearts', rank: 'ace', value: 1}
    ▶ 27: Card {suit: 'hearts', rank: '2', value: 2}
    ▶ 28: Card {suit: 'hearts', rank: '3', value: 3}
    ▶ 29: Card {suit: 'hearts', rank: '4', value: 4}
    ▶ 30: Card {suit: 'hearts', rank: '5', value: 5}
    ▶ 31: Card {suit: 'hearts', rank: '6', value: 6}
    ▶ 32: Card {suit: 'hearts', rank: '7', value: 7}
    ▶ 33: Card {suit: 'hearts', rank: '8', value: 8}
    ▶ 34: Card {suit: 'hearts', rank: '9', value: 9}
    ▶ 35: Card {suit: 'hearts', rank: '10', value: 10}
    ▶ 36: Card {suit: 'hearts', rank: 'jack', value: 11}
    ▶ 37: Card {suit: 'hearts', rank: 'queen', value: 12}
    ▶ 38: Card {suit: 'hearts', rank: 'king', value: 13}
    ▶ 39: Card {suit: 'spades', rank: 'ace', value: 1}
    ▶ 40: Card {suit: 'spades', rank: '2', value: 2}
    ▶ 41: Card {suit: 'spades', rank: '3', value: 3}
    ▶ 42: Card {suit: 'spades', rank: '4', value: 4}
    ▶ 43: Card {suit: 'spades', rank: '5', value: 5}
    ▶ 44: Card {suit: 'spades', rank: '6', value: 6}
    ▶ 45: Card {suit: 'spades', rank: '7', value: 7}
    ▶ 46: Card {suit: 'spades', rank: '8', value: 8}
    ▶ 47: Card {suit: 'spades', rank: '9', value: 9}
    ▶ 48: Card {suit: 'spades', rank: '10', value: 10}
    ▶ 49: Card {suit: 'spades', rank: 'jack', value: 11}
    ▶ 50: Card {suit: 'spades', rank: 'queen', value: 12}
    ▶ 51: Card {suit: 'spades', rank: 'king', value: 13}
length: 52

```



```

0: Card {suit: 'hearts', rank: '8', value: 8}
► 1: Card {suit: 'spades', rank: '3', value: 3}
► 2: Card {suit: 'diamonds', rank: '10', value: 10}
► 3: Card {suit: 'spades', rank: '7', value: 7}
► 4: Card {suit: 'hearts', rank: '9', value: 9}
► 5: Card {suit: 'spades', rank: '5', value: 5}
► 6: Card {suit: 'hearts', rank: '2', value: 2}
► 7: Card {suit: 'diamonds', rank: '8', value: 8}
► 8: Card {suit: 'diamonds', rank: '9', value: 9}
► 9: Card {suit: 'diamonds', rank: '7', value: 7}
► 10: Card {suit: 'diamonds', rank: '3', value: 3}
► 11: Card {suit: 'hearts', rank: '10', value: 10}
► 12: Card {suit: 'spades', rank: 'queen', value: 12}
► 13: Card {suit: 'clubs', rank: '6', value: 6}
► 14: Card {suit: 'clubs', rank: '5', value: 5}
► 15: Card {suit: 'spades', rank: 'ace', value: 1}
► 16: Card {suit: 'spades', rank: 'king', value: 13}
► 17: Card {suit: 'clubs', rank: '3', value: 3}
► 18: Card {suit: 'diamonds', rank: '2', value: 2}
► 19: Card {suit: 'clubs', rank: '7', value: 7}
► 20: Card {suit: 'hearts', rank: 'king', value: 13}
► 21: Card {suit: 'hearts', rank: '6', value: 6}
► 22: Card {suit: 'hearts', rank: '7', value: 7}

```

```

0: Player
  ▼ playerCards: Array(26)
    ▶ 0: Card {suit: 'hearts', rank: 'ace', value: 1}
    ▶ 1: Card {suit: 'spades', rank: '4', value: 4}
    ▶ 2: Card {suit: 'spades', rank: '10', value: 10}
    ▶ 3: Card {suit: 'clubs', rank: 'king', value: 13}
    ▶ 4: Card {suit: 'hearts', rank: '9', value: 9}
    ▶ 5: Card {suit: 'spades', rank: '6', value: 6}
    ▶ 6: Card {suit: 'diamonds', rank: '10', value: 10}
    ▶ 7: Card {suit: 'spades', rank: '5', value: 5}
    ▶ 8: Card {suit: 'diamonds', rank: 'ace', value: 1}
    ▶ 9: Card {suit: 'spades', rank: '2', value: 2}
    ▶ 10: Card {suit: 'spades', rank: '9', value: 9}
    ▶ 11: Card {suit: 'clubs', rank: '5', value: 5}
    ▶ 12: Card {suit: 'clubs', rank: 'ace', value: 1}
    ▶ 13: Card {suit: 'spades', rank: 'ace', value: 1}
    ▶ 14: Card {suit: 'clubs', rank: '6', value: 6}
    ▶ 15: Card {suit: 'diamonds', rank: '7', value: 7}
    ▶ 16: Card {suit: 'hearts', rank: '7', value: 7}
    ▶ 17: Card {suit: 'clubs', rank: 'jack', value: 11}
    ▶ 18: Card {suit: 'diamonds', rank: 'queen', value: 12}
    ▶ 19: Card {suit: 'hearts', rank: '3', value: 3}
    ▶ 20: Card {suit: 'diamonds', rank: 'king', value: 13}
    ▶ 21: Card {suit: 'hearts', rank: '5', value: 5}
    ▶ 22: Card {suit: 'spades', rank: 'king', value: 13}
    ▶ 23: Card {suit: 'clubs', rank: '7', value: 7}
    ▶ 24: Card {suit: 'diamonds', rank: '5', value: 5}
    ▶ 25: Card {suit: 'clubs', rank: '8', value: 8}
    length: 26
    ▶ [[Prototype]]: Array(0)
  playerName: "Bonnie"

```



PLAYERS TIED!!! NO POINTS AWARDED
BONNIE plays: 3 of spades
CLYDE plays: 3 of hearts
PLAYERS TIED!!! NO POINTS AWARDED
BONNIE plays: 11 of spades
CLYDE plays: 8 of diamonds
BONNIE WINS THIS ROUND!
BONNIE plays: 12 of hearts
CLYDE plays: 6 of hearts
BONNIE WINS THIS ROUND!
BONNIE plays: 9 of hearts
CLYDE plays: 12 of spades
CLYDE WINS THIS ROUND!
BONNIE plays: 6 of diamonds
CLYDE plays: 2 of clubs
BONNIE WINS THIS ROUND!
BONNIE plays: 2 of spades
CLYDE plays: 11 of diamonds
CLYDE WINS THIS ROUND!
BONNIE plays: 14 of hearts
CLYDE plays: 10 of clubs
BONNIE WINS THIS ROUND!
BONNIE plays: 7 of clubs
CLYDE plays: 10 of hearts
CLYDE WINS THIS ROUND!

```
wac-game, js:8.8
wac-game, js:6.6
wac-game, js:6.6
wac-game, js:8.8
wac-game, js:6.6
wac-game, js:6.6
wac-game, js:7.7
wac-game, js:6.6
wac-game, js:6.6
wac-game, js:8.8
wac-game, js:6.6
wac-game, js:6.6
wac-game, js:6.6
wac-game, js:6.6
wac-game, js:6.6
wac-game, js:8.8
wac-game, js:6.6
wac-game, js:6.6
wac-game, js:7.7
wac-game, js:6.6
wac-game, js:8.8
```



PROMINEO TECH

▼ 0: Player

▼ playerCards: Array(26)

```
▶ 0: Card {suit: 'spades', rank: '10', value: 10}
▶ 1: Card {suit: 'diamonds', rank: '5', value: 5}
▶ 2: Card {suit: 'spades', rank: '4', value: 4}
▶ 3: Card {suit: 'spades', rank: 'ace', value: 14}
▶ 4: Card {suit: 'spades', rank: '9', value: 9}
▶ 5: Card {suit: 'diamonds', rank: '10', value: 10}
▶ 6: Card {suit: 'spades', rank: '5', value: 5}
▶ 7: Card {suit: 'diamonds', rank: '7', value: 7}
▶ 8: Card {suit: 'clubs', rank: 'queen', value: 12}
▶ 9: Card {suit: 'spades', rank: '6', value: 6}
▶ 10: Card {suit: 'hearts', rank: 'jack', value: 11}
▶ 11: Card {suit: 'diamonds', rank: 'ace', value: 14}
▶ 12: Card {suit: 'clubs', rank: '5', value: 5}
▶ 13: Card {suit: 'clubs', rank: '3', value: 3}
▶ 14: Card {suit: 'clubs', rank: 'ace', value: 14}
▶ 15: Card {suit: 'diamonds', rank: 'king', value: 13}
▶ 16: Card {suit: 'hearts', rank: '4', value: 4}
▶ 17: Card {suit: 'spades', rank: 'king', value: 13}
▶ 18: Card {suit: 'spades', rank: '3', value: 3}
▶ 19: Card {suit: 'spades', rank: 'jack', value: 11}
▶ 20: Card {suit: 'hearts', rank: 'queen', value: 12}
▶ 21: Card {suit: 'hearts', rank: '9', value: 9}
▶ 22: Card {suit: 'diamonds', rank: '6', value: 6}
▶ 23: Card {suit: 'spades', rank: '2', value: 2}
▶ 24: Card {suit: 'hearts', rank: 'ace', value: 14}
▶ 25: Card {suit: 'clubs', rank: '7', value: 7}
length: 26
```

▶ [[Prototype]]: Array(0)

playerName: "BONNIE"

playerScore: 16

BONNIE WINS THIS ROUND!!

▼ 1: Player

▼ playerCards: Array(26)

```
▶ 0: Card {suit: 'hearts', rank: '8', value: 8}
▶ 1: Card {suit: 'diamonds', rank: '2', value: 2}
▶ 2: Card {suit: 'hearts', rank: '2', value: 2}
▶ 3: Card {suit: 'spades', rank: '8', value: 8}
▶ 4: Card {suit: 'hearts', rank: '5', value: 5}
▶ 5: Card {suit: 'diamonds', rank: 'queen', value: 12}
▶ 6: Card {suit: 'diamonds', rank: '3', value: 3}
▶ 7: Card {suit: 'hearts', rank: '7', value: 7}
▶ 8: Card {suit: 'spades', rank: '7', value: 7}
▶ 9: Card {suit: 'diamonds', rank: '4', value: 4}
▶ 10: Card {suit: 'clubs', rank: '8', value: 8}
▶ 11: Card {suit: 'clubs', rank: 'jack', value: 11}
▶ 12: Card {suit: 'clubs', rank: '4', value: 4}
▶ 13: Card {suit: 'clubs', rank: '9', value: 9}
▶ 14: Card {suit: 'clubs', rank: '6', value: 6}
▶ 15: Card {suit: 'hearts', rank: 'king', value: 13}
▶ 16: Card {suit: 'diamonds', rank: '9', value: 9}
▶ 17: Card {suit: 'clubs', rank: 'king', value: 13}
▶ 18: Card {suit: 'hearts', rank: '3', value: 3}
▶ 19: Card {suit: 'diamonds', rank: '8', value: 8}
▶ 20: Card {suit: 'hearts', rank: '6', value: 6}
▶ 21: Card {suit: 'spades', rank: 'queen', value: 12}
▶ 22: Card {suit: 'clubs', rank: '2', value: 2}
▶ 23: Card {suit: 'diamonds', rank: 'jack', value: 11}
▶ 24: Card {suit: 'clubs', rank: '10', value: 10}
▶ 25: Card {suit: 'hearts', rank: '10', value: 10}
length: 26
```

▶ [[Prototype]]: Array(0)

playerName: "CLYDE"

playerScore: 6

CONGRATULATIONS BONNIE WINS THE GAME WITH A FINAL SCORE OF: 14!!!

passes: 2 failures: 0 duration: 0.07s (100%)

[Index](#)

Deck

- ✓ should create a new deck
- ✓ should shuffle deck

URL to GitHub Repository:

<https://github.com/tspears9/war.git>