**URL to GitHub Repository:**

**URL to Your Coding Assignment Video:**

**Instructions:**

* In Visual Studio Code, write the code that accomplishes the objectives listed below and ensures that the code compiles and runs as directed.
* Create a new repository on GitHub for this week’s assignments and push this document, with your project code, to the repository.
* Include the URLs for this week’s repository and video where instructed.
* Submit this document as a .PDF file in the LMS.

**Coding Steps:**

* For the final project you will be creating an automated version of the classic card game *WAR!* There are many versions of the game *WAR.* In this version there are only 2 players.
  + You do not need to do anything special when there is a tie in a round.
* Think about how you would build this project and write your plan down. Consider classes such as: **Card**, **Deck**, **Player**, as well as what **properties** and **methods** they may include.
  + You do not need to accept any user input, when you run your code, the entire game should play out instantly without any user input inside of your browser’s console.

**The completed project should, when executed, do the following:**

* Deal 26 Cards to each Player from a Deck of 52 cards.
* 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 both Players
* After all cards have been played, display the score and declare the winner.
* Write a Unit Test using Mocha and Chai for at least one of the functions you write.

**Video Steps:**

* Create a video, up to five minutes max, showing and explaining how your project works with an emphasis on the portions you contributed.
* This video should be done using screen share and voice over.
* This can easily be done using Zoom, although you don't have to use Zoom, it's just what we recommend.
  + You can create a new meeting, start screen sharing, and start recording.
  + This will create a video recording on your computer.
* This should then be uploaded to a publicly accessible site, such as YouTube.
  + Ensure the link you share is **PUBLIC** or **UNLISTED**!
  + If it is not accessible by your grader, your project will be graded based on what they can access.

class card {

    constructor(face, value, suit){

        this.face = face;

        this.value = value;

        this.suit = suit;

    }

}

//console.log(deckOfCards);

//const firstCard = new card ('A', 14, '♣️');

//console.log(firstCard);

class deck {

    constructor() {

     this.wholeDeck = this.makeDeck();

    }

    makeDeck(){

        let values = [2,3,4,5,6,7,8,9,10,'J','Q','k','A'];

        let suits = ['♣️','♦️','♠️','❤️'];

        let deckOfCards = [];

        for (let x = 0; x < values.length; x++){

            for (let s = 0; s < suits.length; s++){

                let War = new card(suits[s],values[x],x+2);

                deckOfCards.push(War);

            }

        }

        deckOfCards = deckOfCards.sort((a, b) => 0.5 - Math.random());

        return deckOfCards;

    }

}

const Najmeh = new deck();

playGame(Najmeh.wholeDeck);

function playGame(deck){

    round = 1;

    let player1 = {

     hand: deck.splice(0,26),

     //26 cards

    }

    let player2 = {

        hand: deck,

    }

   console.log({player1});

   console.log({player2});

//show your top card

alert(`player1 plays a ${player1.hand[0].face}; player2 plays a${player2.hand[0].face}`);

}