

## TypeScript Project: Blackjack Card Game

This document contains the instructions for a take-home project involving implementing a blackjack card game. This will be a simplified version of the game that involves no splitting, doubling down or surrendering. The only available options for the player will be to “hit” or “stand”.

### **Project Description**

Code a simplified version of Blackjack in TypeScript, focusing on fundamental programming concepts and TypeScript's features. This project is designed to familiarize you with TypeScript syntax, object-oriented programming, and basic game development concepts within a Node.js environment. This project should be completed as a command line application that uses Node.js.

### **Template/Starting Code**

In this google drive folder you will find a folder named: “Starting Code”. This contains a template .ts file that you can use to get started with. It contains **some** of the functions you will need to write to complete this project. Using this starting code is optional and I recommend first attempting the project on your own before using it.

### **Solution**

The official solution and completed code for this project is also available for reference in the google drive folder. I recommend attempting the project on your own before looking at it.

Note: There are many ways to complete this project, the provided solution is simply one of many correct solutions.

### **Collection User Input**

To collect user-input in node.js you can use the npm package: prompt-sync. You can install it using the below commands:

```
npm install prompt-sync
npm i --save-dev @types/prompt-sync
```

To collect input you'll need to use the following code snippet:

```
import promptSync from 'prompt-sync';
const prompt = promptSync();

// Ask for the user's name
const name = prompt('What is your name? ');
```

## Shuffling The Cards

To shuffle an array of elements (the cards) you can use this code snippet.

```
function shuffleArray(array) {
  for (let i = array.length - 1; i > 0; i--) {
    // Generate a random index from 0 to i
    const j = Math.floor(Math.random() * (i + 1));
    // Swap elements at indices i and j
    [array[i], array[j]] = [array[j], array[i]];
  }
  return array;
}

// Example usage:
const myArray = [1, 2, 3, 4, 5];
console.log("Original array:", myArray);

const shuffledArray = shuffleArray(myArray);
console.log("Shuffled array:", shuffledArray);
```

## Explanation of the Game

Blackjack, often known as 21, is a card game where the player competes against the dealer with the goal of having the total value of their cards come as close to 21 as possible without exceeding it. Here's a detailed breakdown of how the game works and the rules you'll implement:

- **Game Setup:**
  - The game uses a standard deck of 52 playing cards.
  - Each card has a value:
    - Number cards (2 through 10) are worth their face value.
    - Face cards (Jack, Queen, King) are each worth 10.

- Aces can be worth either 1 or 11, depending on which value benefits the player's hand the most.
- The player starts with a bankroll of \$100.
- Before each round, the player places a bet. If the player runs out of money, the game ends.
- **Game Flow:**
  - Placing Bets: The player decides how much to bet from their available funds.
  - Dealing Cards: Initially, both the player and the dealer are dealt two cards. The player's cards are both face up, while the dealer has one card face up and one face down (hidden).
  - Player's Turn:
    - If the player's initial two cards total 21 (an Ace and a 10-value card), this is called a "Blackjack." The player wins 3:2 on their bet immediately, unless the dealer also has a Blackjack, in which case the game ends.
    - If not a Blackjack, the player has the option to "Hit" (request additional cards) one at a time to try to get closer to 21. The player can hit as many times as they like but will "Bust" (automatically lose) if their total exceeds 21.
    - The player can also "Stand" (not take any more cards) if they are satisfied with their hand's total value.
  - Dealer's Turn: After the player stands, the dealer reveals their hidden card. The dealer must hit if their total is less than 17 and stand once it reaches 17 or more.
  - Determining the Winner:
    - If the dealer busts or the player's total is closer to 21 than the dealer's without exceeding 21, the player wins and doubles their bet.
    - If both the player and the dealer have the same total, the game is a "Push" (tie), and the player's bet is returned.
    - If the dealer has a higher total than the player without busting, or if the player busts, the player loses their bet.

## Examples of Gameplay

### *Example 1: Player Wins by Getting Closer to 21*

Player's funds: \$100

Enter your bet: \$20

Your hand: 7♠, 5♥ (Total: 12)

Dealer's hand: 10♣, [hidden]

Your action (hit/stand): hit

Your hand: 7♠, 5♥, 9♦ (Total: 21)  
Your action (hit/stand): stand  
Dealer's hand: 10♣, 6♠ (Total: 16)  
Dealer hits: 10♣, 6♠, 4♠ (Total: 20)  
You win \$20!  
Player's funds: \$120

*Example 2: Player Gets Blackjack*

Player's funds: \$100  
Enter your bet: \$10  
Your hand: A♥, K♦ (Blackjack!)  
Dealer's hand: 9♠, [hidden]  
You win \$15! (3:2 payout for Blackjack)  
Player's funds: \$115

*Example 3: Player Busts*

Player's funds: \$100  
Enter your bet: \$20  
Your hand: 8♦, 6♣ (Total: 14)  
Dealer's hand: K♠, [hidden]  
Your action (hit/stand): hit  
Your hand: 8♦, 6♣, 9♥ (Total: 23 - Bust!)  
Dealer's hand: K♠, 3♦ (Total: 13)  
You bust and lose \$20.  
Player's funds: \$80

*Example 4: Dealer Busts, Player Wins*

Player's funds: \$80  
Enter your bet: \$10  
Your hand: 10♣, 2♠ (Total: 12)  
Dealer's hand: 6♥, [hidden]  
Your action (hit/stand): hit  
Your hand: 10♣, 2♠, 7♦ (Total: 19)  
Your action (hit/stand): stand  
Dealer's hand: 6♥, 10♠ (Total: 16)  
Dealer hits: 6♥, 10♠, 6♦ (Total: 22 - Dealer Busts!)  
You win \$10.  
Player's funds: \$90

*Example 5: Push (Tie)*

Player's funds: \$90  
Enter your bet: \$15  
Your hand: Q♦, 7♠ (Total: 17)  
Dealer's hand: 9♦, [hidden]  
Your action (hit/stand): stand  
Dealer's hand: 9♦, 8♣ (Total: 17)  
It's a push! Your bet is returned.  
Player's funds: \$90

*Example 6: Dealer Has Blackjack, Player Loses*

Player's funds: \$90  
Enter your bet: \$20  
Your hand: 9♠, 8♣ (Total: 17)  
Dealer's hand: A♠, [hidden]  
Dealer reveals: A♠, K♣ (Blackjack!)  
Dealer has Blackjack. You lose \$20.  
Player's funds: \$70

*Example 7: Dealer Wins By Getting Closest to 21*

Player's funds: \$70  
Enter your bet: \$25  
Your hand: 4♥, 5♦ (Total: 9)  
Dealer's hand: 7♠, [hidden]  
Your action (hit/stand): hit  
Your hand: 4♥, 5♦, 8♠ (Total: 17)  
Your action (hit/stand): stand  
Dealer's hand: 7♠, 9♥ (Total: 16)  
Dealer hits: 7♠, 9♥, 5♣ (Total: 21)  
Dealer wins. You lose \$25.  
Player's funds: \$45