Testing Log

|  |  |
| --- | --- |
| Card(), (Card class) | The default constructor for card that defines a card with no suit and the number 0. We used this class to create a deck full of cards like this instead of using null to avoid a null pointer exception. |
| Card(aSuit,Number),(Card class) | Another way to create a card class. In this constructor the user can specify which suit and which number that we want. We used this class with the shuffler method to randomize a deck of cards. |
| getNumber(),(Card class) | This method was used to return the number a specific card. We used this when we wanted to get the sum of the user’s hand. |
| toString(),(Card class) | This method correctly formated a string of the type of card (four of spades). We used this when we wanted to print out the user’s hand. |
| Deck(), (Deck class) | The default constructor for the deck class. This constructor initializes a previously defined deck array with each possible card type. |
| shuffle(),(Deck class) | This method creates a random number and replaces the card at each index with a card at the random index. We used this after initializing the deck so that when the cards are dealt they are random. |
| deal(),(Deck class) | This method returns the card one index down from the top of the deck. We used this at the start of the game and when the user or CPU wanted to hit. |
| Player(),(player class) | The default constructor for the player class. We used this to test if a player, named user is created, with an empty hand. |
| Player(name), (player class) | The other constructor for the player class. It creates a player with the specified name and an empty hand. This was used to create the cpu. |
| emptyhand(),(player class) | This method empties the player’s hand by going through each index and setting the card in that spot to an empty card. We used this at the start of each run of the game to empty the player’s hand so that the old cards don’t effect the next round. |
| addCard(card),(player class) | This method adds another card to the player’s hand. We used this method to add another card when the user selects ‘H’. It also check to see if the user will go over 21 when adding the card and will stop the user from adding more cards. |
| gethandsum(), (player class) | This method adds each index of the player’s hand to give a final sum of the total hand. We use this when we check to see who wins the game. |
| printhand(showhand), (player class) | This method prints out the user hand or prints out hidden based on the boolean passed in for showhand. We use this when we want to show the use what cards he/she has so that they can decide if they want to hit or stay. |

Overall Testing Report:

The overall idea of testing our code was to use the constructors and methods to see if the game behaved the way we wanted it to be. This way we can add on to any of the methods if something goes wrong and enhance the game to do things at the specific time.