**Test Cases for Deck class**

**Test Case 1: createDeck() : void**

**Rules:**

|  |  |  |
| --- | --- | --- |
| **Rules/Constraints** | **Valid**  **Equivalence Classes** | **Invalid**  **Equivalence**  **Classes** |
| * The method must set the unshuffledDeck variable to a list of 52 cards | 1. 52 | 1. Everything else |
| * The "cards" in the unshuffledDeck queue are instances of the Card class | 1. True | 1. False |

**Mapping:**

|  |  |
| --- | --- |
| **Test equivalence # mapping** | **Test value** |
| 1 | unshuffledDeck.size() = 52 |
| 2 | unshuffledDeck.size() = 51 |
| 3 | unshuffledDeck.peek() instanceof Card = true |
| 4 | unshuffledDeck.peek() instanceof Card = false |

**Test Case 2: shuffle() : void**

**Rules:**

|  |  |  |
| --- | --- | --- |
| **Rules/Constraints** | **Valid**  **Equivalence Classes** | **Invalid**  **Equivalence**  **Classes** |
| * The shuffled deck (result) must be different from the unshuffled one | 1. They are different | 1. They are the same |
| * Both the shuffled and unshuffled decks must have a size of 52 | 1. 52 | 1. Numbers other than 52 |

**Mapping:**

|  |  |
| --- | --- |
| **Test equivalence # mapping** | **Test value** |
| 1 | Shuffled deck: [AC, KC, QC, JC, 10C …]  Unshuffled deck: [5D, 4C, KS, 9H, 9D …] |
| 2 | Shuffled deck: [AC, KC, QC, JC, 10C …]  Unshuffled deck: [AC, KC, QC, JC, 10C …] |
| 3 | Shuffled deck with a size 52  Unshuffled deck with a size of 52 |
| 4 | Unshuffled deck with a size of 52  Shuffled deck with a size of 51 |
| 4 | Unshuffled deck with a size of 52  Shuffled deck with a size of 53 |

**Test Case 3: deal(): Card**

**Rules:**

|  |  |  |
| --- | --- | --- |
| **Rules/Constraints** | **Valid**  **Equivalence Classes** | **Invalid**  **Equivalence**  **Classes** |
| * The method must remove the first card from a deck | 1. Removes the first card from a deck | 1. Removes a card other than the first one |

**Mapping:**

|  |  |
| --- | --- |
| **Test equivalence # mapping** | **Test value** |
| 1 | deck: AD, 4C, 3H  deck.deal() : AD |
| 2 | deck: AD, 4C, 3H  deck.deal() : 4C |

**Test Case 4: size(): int**

**Rules:**

|  |  |  |
| --- | --- | --- |
| **Rules/Constraints** | **Valid**  **Equivalence Classes** | **Invalid**  **Equivalence**  **Classes** |
| * Must Return the size of the number of cards that have not been dealt yet | 1. Returns the correct size of shuffledDeck | 1. Returns the wrong size of shuffledDeck |
| * Must Return a value between 0 and 52 (inclusive) | 1. 0 <= size <= 52 | 1. size < 0 OR size > 52 |

**Mapping:**

|  |  |
| --- | --- |
| **Test equivalence # mapping** | **Test value** |
| 1,3 | shuffledDeck: 52 cards  deck.size() = 52 |
| 1,3 | shuffledDeck: 0 cards  deck.size() = 0 |
| 1,3 | shuffledDeck: 51 cards  deck.size() = 51 |
| 1,3 | shuffledDeck: 1 card  deck.size() = 1 |
| 2,4 | shuffledDeck: 1 card  deck.size() = -1 |
| 2,4 | shuffledDeck: 52 cards  deck.size() = 53 |
| 2,4 | shuffledDeck: 52 cards  deck.size() = 54 |

**Test Cases for Card class**

**Test Case 1: toString(): String**

**Rules:**

|  |  |  |
| --- | --- | --- |
| **Rules/Constraints** | **Valid**  **Equivalence Classes** | **Invalid**  **Equivalence**  **Classes** |
| * Returns a string of the rank and the suit of the current Card object | 1. Returns the correct string | 1. Returns the wrong string |

**Mapping:**

|  |  |
| --- | --- |
| **Test equivalence # mapping** | **Test value** |
| 1 | Card: {rank: 10, suit: Spades}  "10 of Spades" |
| 2 | Card: {rank: 2, suit: Hearts}  "4 of Spades" |

**Test cases for Player Class**

**Test Case #: method(): void**

**Rules:**

|  |  |  |
| --- | --- | --- |
| **Rules/Constraints** | **Valid**  **Equivalence Classes** | **Invalid**  **Equivalence**  **Classes** |
|  |  |  |

**Mapping:**

|  |  |
| --- | --- |
| **Test equivalence # mapping** | **Test value** |
| 1 |  |
| 2 |  |