Deck

- DECK_SIZE : int
- SHUFFLE_COUNT : int
- deck : Card
- rng : Random
- dealIndex : int
+Deck()
+ shuffle()
+ getIndex(i : int) : Card
+ getDeckSize: int
+ deal() : Card
+ resetDeck()

Card

+ value : int
+ suit : int

+ Card(suit: int, value:int)
+ getCard() : String
+ getSuitString(): String
+ compareTo(other : Card) : int

Poker main

+ checkhand(hand:Hand)

Hand

- HAND_SIZE : int
+Hand(c1:Card, c2:card)
+ getCard(): Card

Board

+printBoard()
+getBoard() : Card

CompareHands

This was our Use Case developed early on in the project. It did not end up being our final outcome, but helped us to align the program we developed to match our User Stores.