blackJack -PUSH:int -PLAYERWINS: int -DEALERWINS: int -BLACKJACK: int -playerIndex: int -dealerIndex: int -balance: int -wager: int -hiddenCard: boolean -stay: boolean -myDeck: cardDeck -dealerHand: String [] -playerHand: String [] -reader: Scanner +blackJack() +getWager(): int +setWager(amount:int): void +getBalance(): int +setBalance(amount:int): void +printDeck():void +printPlayerCards(numCards:int): void +printDealerCards(numCards:int): void +playerTotal():int +dealerTotal():int +checkHands():boolean +getValue(card: String): int +shuffle: void +getCardsLeft():int +dealerPlay(): void +hit():void +hitDD():void +initializeHands():void +initialHand(): void +getBet():void +userOption(): void +userLimitedOption(): void +playBJ(): void +testValueFunction(): void

runBJ

+main(args: String []): void

cardDeck

-DECKSIZE:int

-cards: String[]

-nextCard: int

-deck: int[]

+cardDeck()

+getDeckSize(): int

+setNextCard(value: int): void

+shuffleDeck():void

+getCard(): String

+getCardValue(checkCard: String): int

+getCardsLeftInDeck(): int +getCard(index: int): String

+printDeck(): void