**Top Trumps Java Code**

**Game Sequence**

* Menu – (1) see stats (2) play game
* Deal all user/cpu cards from main deck between players
* Start round 1
* select next user/cpu cards from their own decks
* begin round (print statement)
* display user card to user
* display # of cards in user’s deck
* if player’s turn to choose, display category choices
* else if cpu’s choice, use a method
* compare current cards for all players
* if winner: display winner name, winning card, & allocate win to stats
* if draw, add all current round cards to common pile
* run an isGameWon method to check if game is over (if only 1 player has cards left)
* if a player has lost (0 cards left) & game is not over yet, display “X has lost” & continue game
* Start round 2 … 3 … 4
* End of Game
* Once isGameWon = true, end game (print statement)
* Display winner’s name
* Display game scores (USER = 7 // AI\_1 = 29 // AI\_2 = 15 …)
* Menu

**Methods (Game Logic)**

­dealCards - start of game, give all players cards from the main deck

printRound - start of each round, print "Round x! Players have drawn their cards."

displayCurrentCard - show stats from current card to user

displayCardsLeft - show remaining cards in user's deck

displayCategoryChoices – for human player, list of 5 categories

cpuChooseCategory - find highest number on cpu card & pick that category

compareAllCards - find which player(s) have highest num in chosen category

printRoundWinner - "player X has won this round" & display winning card

roundDraw - adds all current cards to common pile

isGameWon - true if only 1 player has cards left

displayLost - print statement for a player with 0 cards left

displayGameWinner - end of game, print game winner

**Methods (Statistics)**

* trackCards – check # of cards each player has at end of each round. If any have 0, they have lost. (linked to displayLost method)
* roundsWon – keep a count of each round won by each player for stats
* displayGameStats - end of game, print rounds won by each player

**Methods (Get Resources)**

* getCardInfo – retrieve card stats (Height = 5, Strength = 7 …) and load into card objects
* getAllGameStats – retrieve stats on all previous games

**Classes**

* Card class creates card objects for each card in deck (load card values into an array stored in each card object)
* Game objects hold game methods

**Template Game Classes (shown in example program):**

|  |  |
| --- | --- |
| **categoryList**  **currentAIs**  **currentPlayer**  **drawCards**  **game**  **gameStats**  **getLosers**  **humanPlayer**  **isGameOver** | **playerlost**  **processRound**  **roundSelectedCategory**  **roundStatus**  **roundWinner**  **selectCategory**  **startGame**  **stats**  **trySelectCategory** |