_nology

Java Halfway Challenge

_nology + DB - 25th March 2021

Overview

Your task is to model a card game using the Java language. We've broken this challenge down into small parts, with the expectation that some of you will keep things simple and master the basics, while some of you will push on into the stratosphere!

Stage 1

Using classes, ArrayLists and methods, create a deck of Cards. You should use the following classes:

CardGame

- - Contains an ArrayList<Card> for the deckOfCards that contains all 52 cards. This is created and populated when the game is constructed.
- - Has a name which is also defined in the constructor.
- - Has a getDeck method that lists out the cards in the deck.

Card

- - Has a String suit. Use the unicode characters of heart, club, diamond and spade

- - Has a String symbol (2,3,4,5,6,7,8,9,10,J,Q,K,A)
- - Has an int value (2,3,4,5,6,7,8,9,10,11,12,13,14)
- - Has a toString method that describes the class

Stage 2

Add methods for the CardGame class that allow for the following:

- Card dealCard() - takes the card from the top of the deck and returns it
- ArrayList<Card> sortDeckInNumberOrder() - sorts the deck in number order (e.g. 2222333344445555 etc) and stores the new shuffled deck back into the deckOfCards attribute
- ArrayList<Card> sortDeckIntoSuits() - sorts the deck into suits (2,3,4,5,6,7,8,9,10,J,Q,K,A of hearts, then 2,3,4,5,6,7,8,9,10,J,Q,K,A of clubs etc.) and stores the new shuffled deck back into the deckOfCards attribute
- ArrayList<Card> shuffleDeck() - shuffles the deck into a random order and stores the new shuffled deck back into the deckOfCards attribute

Stage 3

Create class for Snap that extends CardGame. This class should use the methods defined above, as well as some new ones, to enable the user to play the game snap according to the following rules:

By pressing enter in the command line, the user takes their turn. Each turn, a new card is dealt from the deck. The game continues until two cards in a row have the same symbol, at which point the “player” wins and the game ends.

Stage 4a

Create a Player class and enable the snap game to be two player, with the users taking it in turns to go. If the snap occurs on the users turn, they win.

Stage 4b

Add a timer so that when there is a snap opportunity, the player has 2 seconds to submit the word “snap” in order to win. If they don’t type it in time, they lose.

Good Luck!