

Deck of Cards Problem Statement

Your assignment is to code in Java a set of classes that represent a deck of poker-style playing cards. (Fifty-two playing cards in four suits: hearts, spades, clubs, diamonds, with face values of Ace, 2-10, Jack, Queen, and King.)

Within one of your classes, you must provide **two** operations:

1. `shuffle()` Shuffle returns no value, but results in the cards in the deck being randomly permuted. Please do not use library-provided “shuffle” operations to implement this function. You may use library provided random number generators in your solution if needed.
2. `dealOneCard()` This function should return one card from the deck to the caller. Specifically, a call to `shuffle` followed by 52 calls to `dealOneCard()` should result in the caller being provided all 52 cards of the deck in a random order. If the caller then makes a 53rd call `dealOneCard()`, no card is dealt.

Many details of this assignment have been left intentionally vague.

Follow the principle of least surprise in making reasonable decisions regarding the implementation. While this is a trivial assignment, pretend that this code will become a **foundational part of a new Appian product**. Take whatever measures you feel are required for your code to meet this bar. We are not concerned with how quickly you complete this assignment.

Take your time and do it right!

Sincerely,
Appian Engineering