JAVA PROGRAMMING LANGUAGE
Assignment 7

1. Write an A*ddressBook* class that manages a collection of Person objects. An *AddressBook* will allow a person to add, delete, or search for a Person object in the address book.
    - *AddressBook()* constructor to create an initialize the ad
    - The *add(Person p)* method should add a person object to the address book. Make sure that the add method does not add duplicate person objects to the address book.
    - The *delete(Person p)* method should remove the specified person object from the address book.
    - The *delete(int id)* method should remove person object with the specified id from the address book.
    - The *search(String name)* method that searches the address book for a specified person and returns the list of persons matching name.
    - The *search(int id)* method that searches the address book for the person with the given id and returns the person object if found and none if the there is no person object with that it in the address bbok.
    - The *print* method to print all the person objects in the address book.

    Write an AddressBookTester class to test your class.

2. It is said that if a deck of cards is given perfect shuffles enough times, it will return to its original order. A perfect shuffle is done by splitting the deck exactly in half and interleaving the cards from the two halves; that is, the first card is from the first half, the second from the second half, the third from the first half, and so on. For example,

    Initial State:

    AH 2H 3H 4H 5H 6H 7H 8H 9H 10H JH QH KH
    AC 2C 3C 4C 5C 6C 7C 8C 9C 10C JC QC KC
    AD 2D 3D 4D 5D 6D 7D 8D 9D 10D JD QD KD
    AS 2S 3S 4S 5S 6S 7S 8S 9S 10S JS QS KS

    After first shuffle:

    AH AD 2H 2D 3H 3D 4H 4D 5H 5D 6H 6D 7H
    7D 8H 8D 9H 9D 10H 10D JH JD QH QD KH KD
    AC AS 2C 2S 3C 3S 4C 4S 5C 5S 6C 6S 7C
    7S 8C 8S 9C 9S 10C 10S JC JS QC QS KC KS

    Create a Deck class with the following methods

    - *Deck()* constructor that creates an unshuffled deck.
    - A *shuffle()* method that does a perfect shuffle.
    - A *toString()* method that print the deck as shown above.
    - An *equals(Deck aDeck)* method that compares itself with the given deck and returns true if all the cards in both decks are in the same order and false otherwise