

Activity 7: Elevens Board Class Design

Introduction:

Now that the `Card` and `Deck` classes are completed, the next class to design is `ElevensBoard`. This class will contain the state (instance variables) and behavior (methods) necessary to play the game of Elevens.

Questions:

1. What items would be necessary if you were playing a game of Elevens at your desk (not on the computer)? List the private instance variables needed for the `ElevensBoard` class.
2. Write an algorithm that describes the actions necessary to play the Elevens game.
3. Now examine the partially implemented `ElevensBoard.java` file found in the **Activity7 Starter Code** directory. Does the `ElevensBoard` class contain all the state and behavior necessary to play the game?

4. `ElevenBoard.java` contains three helper methods. These helper methods are `private` because they are only called from the `ElevenBoard` class.

a. Where is the `dealMyCards` method called in `ElevenBoard`?

b. Which `public` methods should call the `containsPairSum11` and `containsJQK` methods?

c. It's important to understand how the `cardIndexes` method works, and how the list that it returns is used. Suppose that `cards` contains the elements shown below. Trace the execution of the `cardIndexes` method to determine what list will be returned. Complete the diagram below by filling in the elements of the returned list, and by showing how those values index `cards`. Note that the returned list may have less than 9 elements.

	0	1	2	3	4	5	6	7	8
cards ->	J♥	6♣	null	2♠	null	null	A♠	4♥	null

	0	1	2	3	4	5	6	7	8
returned -> list									

- d. Complete the following `printCards` method to print all of the elements of `cards` that are indexed by `cIndexes`.

```
public static printCards(ElevensBoard board) {
    List<Integer> cIndexes = board.cardIndexes();

    /* Your code goes here. */

}
```

- e. Which one of the methods that you identified in question 4b above needs to call the `cardIndexes` method before calling the `containsPairSum11` and `containsJQK` methods? Why?

