Over the last three weeks of the semester, we will be developing a card game using a regular deck of 52 cards. This week you are to create a class that represents a single card, `Card`. A standard deck of playing cards consists of:

- Four **suits**, `int` : **H**earts (1), **C**lubs (2), **D**iamonds (3), and **S**pades (4)
- Thirteen **ranks**, `int` : 1 (Ace), 2-10, 11 (Jack), 12 (Queen), and 13 (King)
- We will omit the jokers

This week: Develop an *immutable* (no setters) class for a single playing card:
- Data members should include *integer* representations for `suit` and `rank`
- Member methods should include:
    - at least one constructor – two parameter, the suit and rank
    - inspectors for `suit`, and `rank`
    - no mutators, since the class is *immutable*.
    - three boolean methods: `suitsMatch()`, `ranksMatch()`, and `equals()` to compare two `Card` objects — the `Card` object receiving the message and a `Card` object parameter
    - a `toString()` method to nicely print the value of the card, such as: "Queen of Spades" or "3 of Diamonds," and
    - a `quickString()` method to print the value of a card as two characters, for example: "QS" or "3D."

```
public class Card{
```