# Overview

The purpose of this lab is to give you some more practice working with developing a class and working with interfaces that are part of the Java library. You will implement the following class

# UML Diagram

| Card |
| --- |
| * value : String * suit : String |
| * Card(String suit, String value) * equals(Object obj) : boolean * toString(): String * compareTo(Object o) : int * getValue(): String * getSuit(): String |

# Resources

<https://www.geeksforgeeks.org/overriding-equals-method-in-java/>

<https://docs.oracle.com/javase/8/docs/api/java/lang/Comparable.html>

# Additional Details

* Valid String values for suit are Spades, Diamonds, Clubs, Hearts
* Valid String values for **value** are 2, 3, 4, 5, 6, 7, 8, 9, 10, Jack, Queen, King, Ace (NOTE: these are the same ordering for comparison
* For equality, two cards are considered the same if they have the same suit and value
* For comparing a card with the specified Card (parameter)for order. Returns a -1, 0, or +1 as this object is less than, equal to, or greater than the specified object. based on the value ordering only (meaning suit doesn't matter).
  + Ace of Hearts compared with Ace of Diamonds returns 0
  + Ace of Hearts compared with King of Clubs returns 1
  + Six of Clubs compared to Eight of Diamonds returns -1
* Your class signature should be **public class Card implements Comparable<Card>**
* toString method should be formatted as <value> of <suit>. E.g. King of Hearts. Note: numbers should be printed as their word form. Specifically Two, Three, Four, Five, Six, Seven, Eight, Nine, Ten
* You may assume the constructor is always called with a valid value and suit

Hint: Look at the test cases, if you fail a Gradescope test case, bring that code into a main method in your Card class and debug.