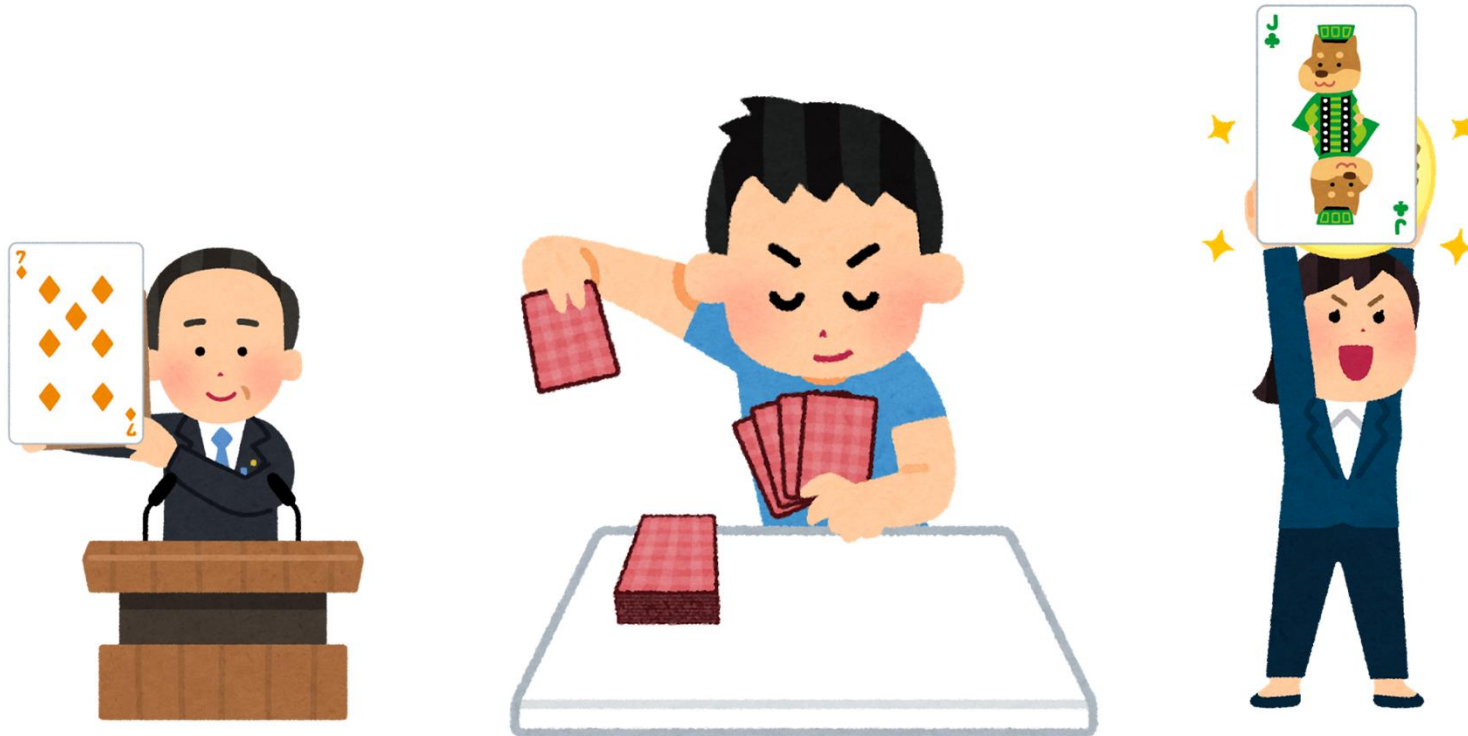


演習課題：
インディアンパワーカー

はじめに

- Java基本文法のまとめとしてプログラミング演習を行います。
- 今回は『インディアンポーカー』のプログラムを作成します。



インディアンポーカーのルール

ルールは以下のとおりです。

(ローカルルールが色々ありますが今回は下記ルールとしてください)

- 参加者は、トランプの山札から1枚を裏向きのまま引きます。
- 引いたトランプ(手札)を自分からは見えないように、他の参加者に公開します（おでこ前に掲げる）。
- 参加者は他人のトランプの数字を確認し、望むなら自分のトランプを捨てて、山札から1枚引き直せます（2回まで）。
- 全員の引き直しが終わったら、全員のトランプを公開します。
- トランプの数字が最も高い人の勝ちです。
- トランプの強さの順序は以下の通り
 - $2 < 3 < 4 < 5 < 6 < 7 < 8 < 9 < 10 < J < Q < K < A$
 - Jokerは無し
 - スート（トランプのマーク）は強さに関係しません
- 捨てたトランプは捨札置き場においておきます（山札に戻さない）。

どんなプログラムを作成するか

今回作成するプログラムの仕様は以下のとおりです。

- プレイヤー（あなた）とコンピュータ(相手)の二人勝負とします。
- 3回勝負とし、引き分けも1回としてカウントします。
- 相手は引き直しを行いません。
- あなたは2回まで引き直しができます。
- トランプの出力は「スート 数字」の形式とします。絵札の場合はそのアルファベットを表示してください。
 - (例)ハート 10
 - (例)クラブ J
- 各勝負のあとに、捨札の一覧と現在までの勝利点を表示します。
- 捨札一覧はスート毎に、数字の小さい順に表示してください。
- 使用するトランプはジョーカーを除いた52枚です。
- 両者の引くトランプはランダムとなるように処理してください。

実行結果

【ゲームスタート】

★1回戦

山札から1枚引きます

相手の表示札：ハート Q

あなたの手札を交換しますか？(残2回)

1:交換

2:勝負

>1

あなたの手札：ハート 8でした

山札から1枚引きます

あなたの手札を交換しますか？(残1回)

1:交換

2:勝負

>1

あなたの手札：クラブ 2でした

山札から1枚引きます

【勝負！】

相手の表示札：ハート Q

あなたの手札：ダイヤ A

あなたの勝ちです！

捨札

スペード		
ハート		8 Q
ダイヤ		A
クラブ		2

勝利点 あなた1：相手0

★2回戦

山札から1枚引きます

相手の表示札：ハート 9

あなたの手札を交換しますか？(残2回)

1:交換

2:勝負

>2

【勝負！】

相手の表示札：ハート 9

あなたの手札：クラブ 7

あなたの負けです！

捨札

スペード		
ハート		8 9 Q
ダイヤ		A
クラブ		2 7

勝利点 あなた1：相手1

★3回戦

山札から1枚引きます

相手の表示札：スペード 2

あなたの手札を交換しますか？(残2回)

1:交換

2:勝負

>2

【勝負！】

相手の表示札：スペード 2

あなたの手札：ハート 6

あなたの勝ちです！

捨札

スペード		2
ハート		6 8 9 Q
ダイヤ		A
クラブ		2 7

勝利点 あなた2：相手1

【最終結果】

あなたの勝ちです！

【ゲーム終了】

プロジェクトについて

- 下記のルールに従ってプロジェクトを作成してください
 - プロジェクト名は「java_exercise_indianpoker_名前」（名前はローマ字）
 - パッケージ名は「jp.co.exercise.ip」
 - クラス名は「IndianPoker」

プログラミングの流れ

- いきなりコードを書き始めるのではなく、プログラムに必要な処理や部品を洗い出すことから始めると良いでしょう
 - 例
 - トランプ52枚は決まった数字の組み合わせだから定数にできそう
 - メッセージも定数にできそう
 - 配列として扱えるデータはなにか
 - 繰り返す必要のある処理はなにか
- プログラム中の処理はセクション分けをして整理すると考えやすくなります
 - 例
 - 山札からトランプを引く
 - トランプを引き直す
 - 勝ち負け判定

定数について

- javaにおける『定数の定義方法』は下記のようになります。

```
public class FizzBuzz {  
    >> public static final int NUM_15 = 15;  
    >> public static final int NUM_3 = 3;  
    >> public static final int NUM_5 = 5;  
  
    >> public static void main(String[] args) throws InterruptedException {  
    >>     int endNum = 100;  
  
    >>     for (int i = 1; i <= endNum; i++) {  
    >>         >> if (i % NUM_15 == 0) {  
    >>             >> System.out.println("FizzBuzz");  
    >>         } else if (i % NUM_3 == 0) {  
    >>             >> System.out.println("Fizz");  
    >>         } else if (i % NUM_5 == 0) {  
    >>             >> System.out.println("Buzz");  
    >>         } else {  
    >>             >> System.out.println(i);  
    >>         }  
    >>         TimeUnit.MILLISECONDS.sleep(200);  
    >>     }  
    >> }  
}
```

public static final 型名 変数名 = 値
これを「mainメソッドの外」に記述します。