**Poker Java BestHand class**

package poker;

import java.util.Arrays;

import java.util.Set;

import java.util.List;

import java.util.Map;

import java.util.\*;

/\*

\* All functions in BestHand class are static, because I need to access BestHand.bestHand() from Poker\_test.java

\* - create Poker object for each hand

\* - then compare each to find the winner

\*/

**class BestHand {**

// constructor, I don't need getter and setter,

BestHand(String[] hands) {

}

// compare two List by compare keys of the List

// return a1 < a2 => -1, a1==a2 => 0; a1>a2 => 1

public static **int compareList(List** a1, List a2) {

// create Integer a1\_keys[]

Integer[] a1\_keys = new Integer[5];

Map.Entry<Integer, Integer> entry1;

for (int i=0; i< a1.size(); i++) {

entry1 = (Map.Entry<Integer, Integer>) a1.get(i);

a1\_keys[i] = entry1.getKey();

}

// create Integer a2\_keys[]

Integer[] a2\_keys = new Integer[5];

Map.Entry<Integer, Integer> entry2;

for (int i=0; i< a1.size(); i++) {

entry2 = (Map.Entry<Integer, Integer>) a2.get(i);

a2\_keys[i] = entry2.getKey();

}

// compare Integer[] a1\_keys and Integer[] a2\_keys

int flag = 0;

for (int i=0; i< a1\_keys.length; i++) {

if (a1\_keys[i] < a2\_keys[i]) { flag = -1; break;}

if (a1\_keys[i] > a2\_keys[i]) { flag = 1; break; }

}

return flag;

}

public static **int comparePokerResult**(PokerResult pokerR1, PokerResult pokerR2) {

int score1, score2;

List ranks1, ranks2;

int flag=0;

score1 = pokerR1.getScore();

score2 = pokerR2.getScore();

ranks1 = pokerR1.valuesRanks();

ranks2 = pokerR2.valuesRanks();

if (score1 > score2) {

flag = 1;

} else if (score1 < score2) {

flag = -1;

} else { // score1 == score2

// need to compare List ranks

flag = compareList(ranks1, ranks2);

}

return flag;

}

public static **String[]** **bestHand(**String[] handsAry) {

//String[] hands = { "4S 2H 6S 2D JH", "2S 4H 6S 4D JH"};

String[] hands = handsAry;

PokerResult bestScore;

PokerResult currentScore;

Poker pk1 = new Poker(hands[0]);

bestScore = pk1.scoreHand();

String bestHand = hands[0];

String currentHand;

for (int i=1; i< hands.length; i++) {

currentHand = hands[i];

Poker pk = new Poker(hands[i]);

currentScore = pk.scoreHand();

if ( **comparePokerResult(**currentScore, bestScore) > 0 ){

bestScore = currentScore;

//bestHand = hands[i];

bestHand = currentHand;

} else if ( comparePokerResult(currentScore, bestScore) == 0 ){

String[] bestHand2 = new String[2];

bestHand2[0] = bestHand;

bestHand2[1] = currentHand;

return bestHand2;

}

}

String[] bestHand1 = new String[1];

bestHand1[0] = bestHand;

return bestHand1;

}

}