Project: Poker

Overview

In this project, you are tasked with implementing a Python program named `poker.py` that simulates the evaluation of poker hands to determine the winning hand among multiple players. Poker, a popular card game, involves strategy, skill, and a bit of luck. Hands in poker are ranked based on the cards' values and suits, with various combinations leading to different hand rankings.

Objectives

Your primary objective is to create a function `poker` that takes as input a list of poker hands, evaluates them according to the standard poker hand rankings, and returns the winning hand.

Specifications

In the card game poker, a hand consists of five cards and are ranked, from lowest to highest, in the following way:

- High Card: Highest value card.
- One Pair: Two cards of the same value.
- Two Pairs: Two different pairs.
- Three of a Kind: Three cards of the same value.
- Straight: All cards are consecutive values.
- Flush: All cards of the same suit.
- Full House: Three of a kind and a pair.
- Four of a Kind: Four cards of the same value.
- Straight Flush: All cards are consecutive values of the same suit.
- Royal Flush: Ten, Jack, Queen, King, Ace, in the same suit.

Card Values: The cards are valued in ascending order as follows:

- Numeric cards: 2 through 10
- Face cards: Jack, Queen, King, Ace (with Ace being the highest)
- Suit: The suit of the cards (Diamonds, Clubs, Spades, Hearts) plays a role in flush-based hands.
- Tie-breaking: If two hands are of the same rank, the winner is determined by the highest-value card, followed by the next highest-value card, and so on.

Hand	Player 1	Player 2	Winner
1	5H 5C 6S 7S KD Pair of Fives	2C 3S 8S 8D TD Pair of Eights	Player 2
2	5D 8C 9S JS AC Highest card Ace	2C 5C 7D 8S QH Highest card Queen	Player 1
3	2D 9C AS AH AC Three Aces	3D 6D 7D TD QD Flush with Diamonds	Player 2
4	4D 6S 9H QH QC Pair of Queens Highest card Nine	3D 6D 7H QD QS Pair of Queens Highest card Seven	Player 1
5	2H 2D 4C 4D 4S Full House With Three Fours	3C 3D 3S 9S 9D Full House with Three Threes	Player 1

Requirements

- Implement the `poker` function which:
  - Accepts a list of 2 poker hands.
  - Each hand is represented as a string separated by spaces, where each separation represents a card. For example, "2D" represents the 2 of Diamonds, and "AH" represents the Ace of Hearts.
  - Returns the winning hand based on poker hand rankings.

Test Cases

Your implementation should pass the following test cases provided in `test_poker.py`. Each test case simulates a round of poker with multiple hands, expecting your function to return the winning hand correctly.

Submission Guidelines

- Your final submission should include the `poker.py` file with the implemented logic and the `test_poker.py` file with all test cases.
- Before submitting, verify that all test cases in `test_poker.py` pass successfully.