Task 13:- finding the winning strategy in a card game

Date: 15/10/25

Aim:- To implement a python program that simulates a card game btw two players and determine the winning strategy using of drawn cards.

Algorithm:-

1. Start the program
2. create a deck of cards.
3. Shuffle the deck randomly.
4. Each player draw a fixed number of cards.
5. Define the strategy
6. Compare the chosen cards of both players.
   - If player1 is card > player 2's card -> player1 scores.
   - else if player 2 is card > player1 is (and) player 2 scores.
   - else -> Draw.
7. Repeat for all rounds.
8. The player with the highest score won the game.
9. Display input output, and final result.
10. End the program

Program:

```
import random
suits = ["Hearts", "Diamonds", "Clubs", "Spades"]
values = list(range(1,14))
deck = [(value, suit) for suit in suits for
    value in values]
random.shuffle(deck)
player1.hand = deck[:5]
player2.hand = deck[5:10]
print("player 1 Hand:", player1.hand)
print("player 2 Hand:", player2.hand)
def play_highest_card(hand):
    highest = max(hand, key=lambda x: x[0])
    hand.remove(highest)
    return highest.
p1_score, p2_score = 0,0
print("\n--- Game Rounds ---")
for i in range(5):
    p1_card = play_highest_card(player1.hand)
    p2_card = play_highest_card(player2.hand)
    print(f"Rand {i+1}=Player 1 -> {p1_card},
        Player 2 -> {p2_card}")
    if p1_card[0] > p2_card[0]:
        print("winner = Player1")
        p1_score += 1
```

Sample input:-

Player 1 Hand :- [('10', 'Hearts'), ('2', 'Clubs'), ('10', 'Spades'), ('3', 'Diamonds'), ('7', 'Clubs')]

Player 2 Hand :- [('4', 'Hearts'), ('12', 'Diamonds'), ('9', 'Clubs'), ('11', 'Spades'), ('6', 'Hearts')]

Sample output:-

Round 1 :- player 1 -> ('10', 'Hearts'), player 2 -> ('12', 'Diamonds')
winner :- player 1

Round 2 :- player 2 -> ('10', 'Spades'), player 2 -> ('11', 'Spades')
winner :- player 2

Round 3 :- player 1 -> ('7', 'Clubs'), player 2 -> ('9', 'Hearts')
winner : player 2

Round 4 : player 1 -> ('5', 'Diamonds'), player 2 -> ('6', 'Hearts')
winner :- player 2

Round 5 :- player 1 -> ('2', 'Clubs'), player 2 -> ('3', 'Clubs')
winner :- player 2

```
elif p2 - card[0] > p1 - card[0]:
    print("winner : player 2")
    p2 - score = 1
else
    print("Result : Draw")
print("---- first Result ....")
print("player 1 score:", p1 - score)
print("player 2 score:", p2 - score)

if p1 - score -> p2 - score
    print("player 1 wone the game with winning
          strategy!")
elif p2 - score > p1 - score
    print("Player 2 wone the game with winning
          strategy!")

else:
    print("the game is a Draw!")
```

VEL TECH	
Ex No.	13
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD	
TOTAL (20)	15
SIGN WITH DATE	

Result: Thus the finding winning strategy for a card game is executed successfully 16/10/25