

Card Game Documentation

1. Author: Brian(Kangjian) Ma (kangjianma@gmail.com)
2. Date: September 8, 2020
3. Programming Language: Python 3
4. Project Interpreter: Python 3.6
5. Class Abstraction: Card Game class with 3 operations including randomly shuffle, deal the top card, and sort in assigned order, which supports two players to determine a winner.
6. Design Assumptions:
 - 1) User can assign the number of decks to play, which should be at least 1.
 - 2) User can assign suit types in a deck while the assigned suit type should be unique, e.g. suits = ['red', 'yellow', 'green'], suits = ['spade', 'club', 'heart', 'diamond'] are valid, while suits = ['red', red, 'green'] will induce an Error.
 - 3) When initiated, each suit has user-assigned number of cards, which can be same or different to other suits. The range of the number on a card of one suit is [0, total number of cards of the suit - 1]
 - 4) User can assign suit-point matching pattern to calculate points in the play, e.g. points = {'red': 3, 'yellow': 2, 'green': 1}
 - 5) There is one type of deck of cards in each play, and different decks of cards can not mix in a play.
7. Operations:

The designed Card Game supports 3 operations and two players can play the game.

 - 1) Shuffle cards in the deck: randomly mix the cards in the card deck, and return a whole deck of cards with a mixed order.
 - 2) Get a card from the top of the deck: get one card from top of the card deck, return a card, and if there is no card left in the deck return error or exception.
 - 3) Sort cards: take a list of color as parameter and sort the card in that color order. Numbers should be in ascending order.
i.e. If the deck has a card contains with following order
(red, 1), (green, 5), (red, 0), (yellow, 3), (green, 2)
Sort cards(['yellow', 'green', 'red']) will return the cards with following order
(yellow, 3), (green, 0), (green, 5), (red, 0), (red, 1)
8. Game Play Rule:

2 players play the game. They will draw 3 cards by taking turns.
Whoever has the high score wins the game. (color point calculation, e.g. red = 3, yellow = 2, green = 1) the point is calculated by color point * number in the card.
9. Functions with Examples:

Class name: CardGame
Funtions:

 - 1) `__init__(self, card_number, suits, points, deck_number)`: Initiate a Card Game

Parameters:

card_number: A list including the number of cards for each suit in one deck
suits: A list including unique suits in one deck
points: A dictionary including suit-point pair to determine play winner
deck_number: an integer no less than 1 which indicate the number of decks to play

Example (Test Case 1):**Input:**

```
from CardGame import *

cardgame = CardGame(card_number=[2, 4, 6], suits=['red', 'yellow', 'green'],
                    points={'red': 3, 'yellow': 2, 'green': 1}, deck_number=1)

print("cards=", cardgame.cards, '\n')

print("suits=", cardgame.suits, '\n')

print("points=", cardgame.points, '\n')

print("deck number=", cardgame.deck_number)
```

Output:

```
cards= [('red', 0), ('red', 1), ('yellow', 0), ('yellow', 1), ('yellow', 2), ('yellow', 3),
        ('green', 0), ('green', 1), ('green', 2), ('green', 3), ('green', 4), ('green', 5)]

suits= ['red', 'yellow', 'green']

points= {'red': 3, 'yellow': 2, 'green': 1}

deck number= 1
```

Explanation:

Initiate a Card Game object named “cardgame”.
In each deck of cards in the game, there are 3 suits, including 2 “red” cards, 4 “yellow” cards, and 6 “green” cards.
The suit-point calculation in the play follow the rule: “red” = 3, “yellow” = 2, “green” = 1.
In each play of the game, there is only deck of cards to play.

- 2) shuffleCards(self, seed=None): Shuffle cards in the deck

Parameters:

seed: A value to reproduce the random shuffle result.
Default value is None, under which case, the shuffle result is not reproducible and unpredictable.

By re-using a seed value, the same sequence should be reproducible from run to run as long as multiple threads are not running.

Example (Test Case 6):

Input:

```
from CardGame import *

cardgame = CardGame(card_number=[2, 4, 6], suits=['red', 'yellow', 'green'],
                    points={'red': 3, 'yellow': 2, 'green': 1}, deck_number=1)

print("Original Cards=", cardgame.cards, '\n')

cardgame.shuffleCards(seed=1)

print("Shuffled Cards=", cardgame.cards, '\n')
```

Output:

```
Original Cards= [('red', 0), ('red', 1), ('yellow', 0), ('yellow', 1), ('yellow', 2), ('yellow',
3), ('green', 0), ('green', 1), ('green', 2), ('green', 3), ('green', 4), ('green', 5)]

Shuffled Cards= [ ('green', 4), ('red', 1), ('yellow', 0), ('yellow', 1), ('yellow', 2),
('yellow', 3), ('green', 0), ('green', 1), ('green', 2), ('red', 0), ('green', 3),
('green', 5)]
```

Explanation:

Initiate a Card Game object named “cardgame” with *Original Cards*.
After reproducibly randomly shuffling with seed=1, the cards are *Shuffled Cards*.

Note: if the “seed” parameter is not implicitly set or set as “None” when calling the shuffleCard(), the shuffling result is random and not reproducible.

- 3) dealCard(self): Get a card from the top of the deck

Parameters: None

Example (Test Case 3):

Input:

```
from CardGame import *

cardgame = CardGame(card_number=[2, 4, 6], suits=['red', 'yellow', 'green'],
                    points={'red': 3, 'yellow': 2, 'green': 1}, deck_number=1)

print("Original Cards=", cardgame.cards)

print("Card Number=", len(cardgame.cards), '\n')
```

```

deal_card = cardgame.dealCard()

print("Deal Card=", deal_card)

print("Cards=", cardgame.cards)

print("Card Number=", len(cardgame.cards), '\n')

```

Output:

```

Original Cards= [('red', 0), ('red', 1), ('yellow', 0), ('yellow', 1), ('yellow', 2), ('yellow',
3), ('green', 0), ('green', 1), ('green', 2), ('green', 3), ('green', 4), ('green', 5)]
Card Number= 12

Deal Card= ('green', 5)
Cards= [('red', 0), ('red', 1), ('yellow', 0), ('yellow', 1), ('yellow', 2), ('yellow', 3),
('green', 0), ('green', 1), ('green', 2), ('green', 3), ('green', 4)]
Card Number= 11

```

Explanation:

A Card Game object is created with one deck of cards and the top card is ('green', 5)].
After calling cardgame.dealCard(), the top card is dealt.

4) sortCards(self, suits_order): Sort cards

Parameters:

suits_order: a list of suits, according to which the cards left in the deck are sorted first and the cards of the same suit are sorted in ascending order afterwards.

Example(Case 4):

Input:

```

cardgame = CardGame(card_number=[2, 4, 6], suits=['red', 'yellow', 'green'],
points={'red': 3, 'yellow': 2, 'green': 1}, deck_number=1)

print("Original Cards=", cardgame.cards, '\n')

cardgame.shuffleCards(seed=1)

print("Shuffled Cards=", cardgame.cards, '\n')

cardgame.sortCards(suits_order=["yellow", "green", "red"])

print("Sorted Cards=", cardgame.cards)

```

Output:

Original Cards= [('red', 0), ('red', 1), ('yellow', 0), ('yellow', 1), ('yellow', 2), ('yellow', 3), ('green', 0), ('green', 1), ('green', 2), ('green', 3), ('green', 4), ('green', 5)]

Shuffled Cards= [('green', 1), ('green', 5), ('red', 0), ('green', 2), ('yellow', 3), ('green', 0), ('yellow', 1), ('green', 4), ('yellow', 2), ('red', 1), ('green', 3), ('yellow', 0)]

Sorted Cards= [('yellow', 0), ('yellow', 1), ('yellow', 2), ('yellow', 3), ('green', 0), ('green', 1), ('green', 2), ('green', 3), ('green', 4), ('green', 5), ('red', 0), ('red', 1)]

Explanation:

A Card Game is initiated with one deck of cards with 3 suits: 2 “red” cards, 4 “yellow” cards, 6 “green” cards.

Shuffle cards with seed=1, so that the shuffling result is reproducible and predictable.

Sort the shuffled cards in the order that “yellow” cards before “green” cards before “red” cards, and cards of the same suit are sorted in ascending order of their numbers.

- 5) play(self): Determine winner between two players

Parameters: None

Example(Test Case 5):**Input:**

```
from CardGame import *
```

```
cardgame = CardGame(card_number=[2, 4, 6], suits=['red', 'yellow', 'green'],
points={'red': 3, 'yellow': 2, 'green': 1}, deck_number=1)
```

```
print("Original Cards=", cardgame.cards, '\n')
```

```
cardgame.play()
```

Output:

Original Cards= [('red', 0), ('red', 1), ('yellow', 0), ('yellow', 1), ('yellow', 2), ('yellow', 3), ('green', 0), ('green', 1), ('green', 2), ('green', 3), ('green', 4), ('green', 5)]

Card Suit - Point: {'red': 3, 'yellow': 2, 'green': 1}

Player 1's cards: [('green', 5), ('green', 3), ('green', 1)]

Player 2's cards: [('green', 4), ('green', 2), ('green', 0)]

Player 1 Wins!(Points: 9)

Explanation:

A Card Game is initiated with a deck of cards: 2 “red” cards, 4 “yellow” cards, 6 “green” cards.

Two players play a game by alternatively take the top card from the deck with player 1 taking first, and the cards taken by each player is as follows:

Player 1's cards: [('green', 5), ('green', 3), ('green', 1)]

Player 2's cards: [('green', 4), ('green', 2), ('green', 0)]

The suit-point matching is assigned as “red”=3, “yellow”=2, “green”=1.

The score for player 1 is $1 * 5 + 1 * 3 + 1 * 1 = 9$ points.

The score for player 2 is $1 * 4 + 1 * 2 + 1 * 0 = 6$ points.

Player 1's score is greater than Player 2's score.

Therefore, Player 1 wins!