
pokercore Documentation

Author

Apr 06, 2019

Contents

1	pokercore package	3
1.1	Subpackages	3
1.1.1	pokercore.test package	3
1.1.1.1	Submodules	3
1.1.1.2	pokercore.test.test_card module	3
1.1.1.3	pokercore.test.test_deck module	3
1.1.1.4	pokercore.test.test_hand module	3
1.1.1.5	Module contents	4
1.2	Submodules	4
1.3	pokercore.card module	4
1.4	pokercore.deck module	4
1.5	pokercore.exceptions module	5
1.6	pokercore.hand module	5
1.7	Module contents	6
2	Indices and tables	7
	Python Module Index	9

Contents:

1.1 Subpackages

1.1.1 pokercore.test package

1.1.1.1 Submodules

1.1.1.2 pokercore.test.test_card module

```
class pokercore.test.test_card.TestCard (methodName='runTest')  
    Bases: unittest.case.TestCase  
  
    test_arithmetic()  
    test_compare()  
    test_create()
```

1.1.1.3 pokercore.test.test_deck module

```
class pokercore.test.test_deck.TestDeck (methodName='runTest')  
    Bases: unittest.case.TestCase  
  
    test_create()  
    test_draw()  
    test_shuffle()
```

1.1.1.4 pokercore.test.test_hand module

```
class pokercore.test.test_hand.TestHand (methodName='runTest')  
    Bases: unittest.case.TestCase
```

setUp()

Hook method for setting up the test fixture before exercising it.

test_compare()

test_create()

test_value()

1.1.1.5 Module contents

1.2 Submodules

1.3 pokercore.card module

class pokercore.card.Card(*rank, suit*)

Bases: object

Class representing a playing card.

A playing card consists of two integers, passed to the constructor during instantiation:

- rank - the rank of the card [0-12]
- suit - the suit of the card [0-3]

A Card can be compared to, added to, subtracted to and subtracted by other Card objects and integers (using their ranks, resulting in plain integers). For identity check, the `identical_to` method is provided.

classmethod from_chars(*chars*)

return a new object from a pair of character symbols

identical_to(*other*)

ranks = ('2', '3', '4', '5', '6', '7', '8', '9', 'T', 'J', 'Q', 'K', 'A')

suits = ('c', 'd', 'h', 's')

1.4 pokercore.deck module

class pokercore.deck.Deck

Bases: object

Class representing a deck of cards.

A Deck consists of 52 Card objects, starting shuffled. It has two methods:

- shuffle - Restore all cards to the deck, then shuffle it.
- **draw** - Draw *n* cards from the top of the deck, returning them as a list of Card objects. If *n* is not provided, draw and return one Card object.

draw(*n=None*)

remove and return a card from the deck, or a list of *n* cards, if *n* is given

shuffle()

restore all cards to deck, then shuffle it

1.5 pokercore.exceptions module

exception `pokercore.exceptions.CardArithmeticError`

Bases: `pokercore.exceptions.CardError`

Invalid arithmetic operation on a Card object

exception `pokercore.exceptions.CardComparisonError`

Bases: `pokercore.exceptions.CardError`

Invalid comparison of a Card object

exception `pokercore.exceptions.CardCreationError`

Bases: `pokercore.exceptions.CardError`

Attempt to create a Card object using invalid arguments

exception `pokercore.exceptions.CardError`

Bases: `pokercore.exceptions.PokerError`

Card-related error

exception `pokercore.exceptions.DeckError`

Bases: `pokercore.exceptions.PokerError`

Deck-related error

exception `pokercore.exceptions.DeckNotIntegerError`

Bases: `pokercore.exceptions.DeckError`

Given draw count is not an integer

exception `pokercore.exceptions.DeckTooManyError`

Bases: `pokercore.exceptions.DeckError`

Given draw count is greater than the count of remaining cards

exception `pokercore.exceptions.HandComparisonError`

Bases: `pokercore.exceptions.HandError`

Invalid comparison of a Hand object

exception `pokercore.exceptions.HandCreationError`

Bases: `pokercore.exceptions.HandError`

Attempt to create a Hand object using invalid arguments

exception `pokercore.exceptions.HandError`

Bases: `pokercore.exceptions.PokerError`

Hand-related error

exception `pokercore.exceptions.PokerError`

Bases: `exceptions.Exception`

Generic poker error

1.6 pokercore.hand module

class `pokercore.hand.Hand(cards)`

Bases: `object`

Class representing a poker hand.

A poker hand consists of one or more Card objects, passed to the constructor contained in some iterable. Its main attributes are two:

- **value** - an integer between 0 and 8 representing the category of the poker hand
- **best_cards** - the best (at most 5) cards that consist the actual hand

A Hand can be compared to other Hand objects, judging by the value, and then the best cards, lexicographically.

classmethod `from_chars(*args)`

return a new object from pairs of character symbols

works with either multiple arguments, or a single iterable

names = ('high card', 'one pair', 'two pair', 'three of a kind', 'straight', 'flush',

1.7 Module contents

pokercore

A poker engine core, in Python

pokercore provides 3 classes to be used in a poker engine. Card (a playing card), Hand (a poker hand consisting of Cards, with evaluation capabilities) and Deck (a deck of Cards).

It is a simple starter, mainly written for exploring purposes, but can be extended and/or used to build something bigger. It is released under the MIT license.

Example

```
>>> from pokercore import Deck, Hand
>>> deck = Deck()
>>> first = Hand(deck.draw(5))
>>> first
Hand(one pair: Card(9c), Card(9h), Card(Ac), Card(Jh), Card(2h))
>>> second = Hand(deck.draw(5))
>>> second
Hand(high card: Card(As), Card(Kc), Card(9s), Card(7h), Card(4h))
>>> first > second
True
```

CHAPTER 2

Indices and tables

- `genindex`
- `modindex`
- `search`

p

- `pokercore`, 6
- `pokercore.card`, 4
- `pokercore.deck`, 4
- `pokercore.exceptions`, 5
- `pokercore.hand`, 5
- `pokercore.test`, 4
- `pokercore.test.test_card`, 3
- `pokercore.test.test_deck`, 3
- `pokercore.test.test_hand`, 3

C

Card (*class in pokercore.card*), 4
 CardArithmeticError, 5
 CardComparisonError, 5
 CardCreationError, 5
 CardError, 5

D

Deck (*class in pokercore.deck*), 4
 DeckError, 5
 DeckNotIntegerError, 5
 DeckTooManyError, 5
 draw() (*pokercore.deck.Deck method*), 4

F

from_chars() (*pokercore.card.Card class method*), 4
 from_chars() (*pokercore.hand.Hand class method*),
 6

H

Hand (*class in pokercore.hand*), 5
 HandComparisonError, 5
 HandCreationError, 5
 HandError, 5

I

identical_to() (*pokercore.card.Card method*), 4

N

names (*pokercore.hand.Hand attribute*), 6

P

pokercore (*module*), 6
 pokercore.card (*module*), 4
 pokercore.deck (*module*), 4
 pokercore.exceptions (*module*), 5
 pokercore.hand (*module*), 5
 pokercore.test (*module*), 4
 pokercore.test.test_card (*module*), 3

pokercore.test.test_deck (*module*), 3
 pokercore.test.test_hand (*module*), 3
 PokerError, 5

R

ranks (*pokercore.card.Card attribute*), 4

S

setUp() (*pokercore.test.test_hand.TestHand method*), 3
 shuffle() (*pokercore.deck.Deck method*), 4
 suits (*pokercore.card.Card attribute*), 4

T

test_arithmetic() (*pokercore.test.test_card.TestCard method*), 3
 test_compare() (*pokercore.test.test_card.TestCard method*), 3
 test_compare() (*pokercore.test.test_hand.TestHand method*), 4
 test_create() (*pokercore.test.test_card.TestCard method*), 3
 test_create() (*pokercore.test.test_deck.TestDeck method*), 3
 test_create() (*pokercore.test.test_hand.TestHand method*), 4
 test_draw() (*pokercore.test.test_deck.TestDeck method*), 3
 test_shuffle() (*pokercore.test.test_deck.TestDeck method*), 3
 test_value() (*pokercore.test.test_hand.TestHand method*), 4
 TestCard (*class in pokercore.test.test_card*), 3
 TestDeck (*class in pokercore.test.test_deck*), 3
 TestHand (*class in pokercore.test.test_hand*), 3