Forritun
Háskólinn í Reykjavík Haustönn 2018

# Skilaverkefni 9 / Project 9

In this project, a main program and some functions are given, but you need to implement three classes used by the main program: *Card()*, *Deck()*, and *PlayingHand()*.

Each *card* has two primary attributes: rank and suit. In a *deck*, there are four suits -- hearts, spades, diamonds, and clubs --, that we represent by their first letters: *H(h)*, *S(s)*, *D(d)*, *C(c)*. Each suit has thirteen cards -- Ace, 2-10, Jack, Queen, and King – with ranks 1-13 in that order. Thus, a *deck* has 52 cards. A *playing hand* has thirteen *cards*.

The following are the requirements for the three classes:

- *Card():*
  - A *constructor* with the parameters *rank* (either a character or an integer) and *suit* (a character). Default values are 0 and '' (empty string). Note that you can use the *type()* function to check the type of a parameter. The internal representation for the rank is an integer in the range -13. The internal representation for the suit is a character: H, S, D or C.
  - Method *__str__()* for returning a string representation of a card. The representation of a card is printed in a right justified field of 3 characters: the ranks followed by the suit. If a card has default values, then 'blk' (blank) is printed. The letters A,J,Q,K are printed for the ranks 1,11,12,13, respectively.
  - Method *is_blank()* that returns True if a card is blank, otherwise False.
- *Deck():*
  - A *constructor* without any parameters. The constructor creates a deck of 52 cards.
  - Method *__str__()* for returning a string representation of a deck, consisting of 4 lines containing 13 cards each.
  - Method *shuffle()*. Shuffles the cards in the deck.
  - Method *deal()*. Deal a single card by returning the card that is removed off the top of the deck.

- *PlayingHand():*
    - A *constructor* without any parameters. The constructor creates a hand of 13 blank cards.
    - Method *__str__()* for returning a string representation of a playing hand, consisting of a single line containing a string representation of each card.
    - Method *add_card()* with the parameter denoting a card. The methods adds the given card to the playing hand at the first blank position.
    - A constant, NUMBER_CARDS, with value 13.

<u>Main program and functions given:</u>

```
def test_cards():
    card1 = Card()
    print(card1)
    card2 = Card(5,'s')
    print(card2)
    card3 = Card('Q','D')
    print(card3)
    card4 = Card('x', 7)
    print(card4)

def print_4_hands(hand1, hand2, hand3, hand4):
    ''' Prints the 4 hands '''
    print(hand1)
    print(hand2)
    print(hand3)
    print(hand4)


def deal_4_hands(deck, hand1, hand2, hand3, hand4):
    ''' Deals cards for 4 hands '''
    for i in range(PlayingHand.NUMBER_CARDS):
        hand1.add_card(deck.deal())
        hand2.add_card(deck.deal())
        hand3.add_card(deck.deal())
        hand4.add_card(deck.deal())

def test_hands(deck):
    hand1 = PlayingHand()
    hand2 = PlayingHand()
    hand3 = PlayingHand()
    hand4 = PlayingHand()
    print("The 4 hands:")
    print_4_hands(hand1, hand2, hand3, hand4)

    deal_4_hands(deck, hand1, hand2, hand3, hand4)
    print("The 4 hands after dealing:")
    print_4_hands(hand1, hand2, hand3, hand4)

# The main program starts here
random.seed(10)
test_cards()

deck = Deck()
deck.shuffle()
print("The deck:")
print(deck)

test_hands(deck)
print("The deck after dealing:")
print(deck)
```

<u>Output from the above program:</u>

```
blk
 5S
 QD
blk
The deck:
 6D 10H  JS  QC  JD  5S  AH  9S  AC  5D  7H  QD  2D
 8S  9D  2C 10D  QS  KS  7S  2H  4D  3S  6H  3H  QH
 4S  3C  5C  9C  KH  7D 10C  6C  4C  2S  6S  JC  9H
 KD  3D  JH  5H  8C  8H  4H  AS  KC  8D  7C  AD 10S
The 4 hands:
blk blk blk blk blk blk blk blk blk blk blk blk blk
blk blk blk blk blk blk blk blk blk blk blk blk blk
blk blk blk blk blk blk blk blk blk blk blk blk blk
blk blk blk blk blk blk blk blk blk blk blk blk blk
The 4 hands after dealing:
 6D  JD  AC  2D 10D  2H  3H  5C 10C  6S  3D  8H  8D
10H  5S  5D  8S  QS  4D  QH  9C  6C  JC  JH  4H  7C
 JS  AH  7H  9D  KS  3S  4S  KH  4C  9H  5H  AS  AD
 QC  9S  QD  2C  7S  6H  3C  7D  2S  KD  8C  KC 10S
The deck after dealing:
```