

주차	6	학습 주제	객체지향프로그래밍-Blackjack Game 구현하기
----	---	-------	-------------------------------

Blackjack Game - 출처: KAIST

There are 52 cards. Each card has a **face** and a **suit**.

- The suites(무늬): club, spade, heart, diamond
- The faces(숫자): 2,3,4,...10, Jack, Queen, King, Ace

The value of a card is the number for a number card, 11 for an Ace, and 10 for Jack, Queen, and King.

We can represent cards as a tuple (face, suit, value)

if card is a card, then card[0] is the face, card[1] is the suit, card[2] is the value.



1. 튜플로 표현한 카드: 자신이 가지고 있는 카드들의 값을 합해보자.

```
def hand_value(hand):
    total=0
    for card in hand:
        total+= card[2]
    return total
hand=[("Queen", "Spade",10),(3,"Heart",3)]
print("카드 숫자의 합:", hand_value(hand))
```

▶ 카드를 튜플로 표현하면 다음과 같은 실수를 할 수 있다.

- card[2]는 무엇을 의미하나? card[1]과 의미가 헷갈리면?
- 실수로 ("Ace", "Spade",5) 라는 튜플을 만들면?

2. 객체로 표현한 카드

```
class Card:
    pass # 내용 구성할 것
c=Card()
c.face="Ace"
c.suit="Spade"
c.value=11
```

▶ 카드들의 합 계산

```
def hand_value(hand): # hand는 객체가 저장된 배열
    total=0
    for card in hand:
        total+= card.value
    return total
```

## 3. Card 클래스 구현

카드의 값은 숫수에 의해 결정되므로 카드의 값을 의미하는 속성이 따로 있을 필요는 없다. 대신, 값을 계산해 주는 멤버 함수 value()를 만들어 사용하자.

```
class Card:
    def __init__(self, face=None, suit=None):
        self.face=face
        self.suit=suit
    def value(self):
        if type(self.face)==int:
            return self.face
        elif self.face=="Ace":
            return 11
        else:
            return 10
    def __str__(self):
        article="a "
        if self.face in [8, "Ace"]:
            article="an "
        return article+str(self.face)+" of "+self.suit
```

```
card1=Card("Ace", "Spade")
card2=Card(2, "Club")
print(card1)
print(card1.value())
print(card2)
print(card2.value())
```

```
# 출력결과
an Ace of Spade
11
a 2 of Club
2
```

## 4. 카드 객체 생성

```
hand=[Card("Ace", "Spade"), Card(8, "Diamond"),
      Card("Jack","Heart"), Card(10,"Club")]
print(hand[1])
print(hand[1].value())
```

```
# 출력 결과=
an 8 of Diamond
8
```

5. 블랙잭 게임에는 52장의 모든 카드를 섞어서 만든 카드 덱(deck)이 필요하다. 카드 덱을 의미하는 클래스를 만들고, 덱에서 카드를 한 장 뽑는 역할의 멤버 함수(draw)를 추가해보자.

```
FACES=list(range(2,11))+["Jack","Queen","King","Ace"]
```

```
SUITS=["Club","Diamond","Heart","Spade"]
```

```
class Deck():
    def __init__(self):
        self.cards=[]
        for suit in SUITS:
            for face in FACES:
                self.cards.append(Card(face,suit))
        random.shuffle(self.cards)
    def draw(self):
        return self.cards.pop()
```

6. 카드 덱

```
def hand_value(hands):
    total=0
    for i in hands:
        total+=i.value()
    return total
```

```
num_players=3
```

```
num_cards=5
```

```
deck=Deck()
```

```
hands=[] # A list of lists (one for each player)
```

```
for i in range(num_players):
```

```
    hands.append([])
```

```
for i in range(num_cards):
```

```
    for j in range(num_players):
```

```
        card=deck.draw()
```

```
        hands[j].append(card)
```

```
        print("player", j+1, "draws", card)
```

```
for i in range(num_players):
```

```
    print("Player {}'s hand (value {}):".format(i+1, hand_value(hands[i])))
```

```
    for card in hands[i]:
```

```
        print(" ", card)
```

## 7. 블랙잭 게임을 만들어 보자. [풀이생략]

You are dealt a 6 of Heart  
 Dealer is dealt a hidden card  
 You are dealt a 3 of Spade  
 Dealer is dealt a 9 of Heart  
 Your total is 9  
 would you like another card? (y/n) y  
 You are dealt an Ace of Club  
 Your total is 20  
 would you like another card? (y/n) n  
 The dealer's hidden card was a 10 of Spades  
 The dealer's total is 19  
 You win!

## 8. 위의 프로그램에 기능을 확장해 보자. (코드 첫 부분에 사용법을 주석으로 달기)

- Ace 카드를 1 또는 11로 계산?
- 딜러가 17이상이 될 때까지 카드를 더 받기
- 사용자와 딜러가 몇 대 몇인지? 승률 계산
- 계속하시겠습니까?(y/n) 묻기
- 기타 등등...

## 9. 블랙잭 게임 인터페이스 예

