

>>>>>>>>>>> 제어 구조의 설계 원리를 중심으로 배우는 >>>>>>>>>>>

# 프로그래밍의 정석

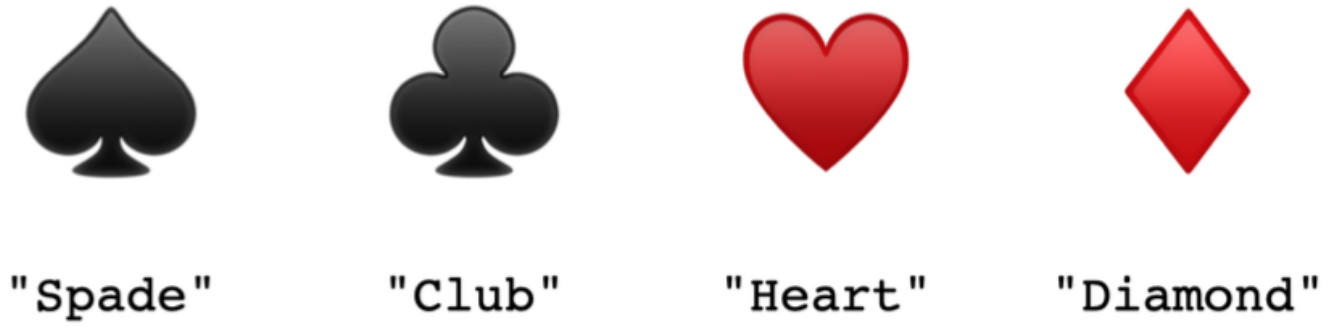
# 과이썬

도경구 지음



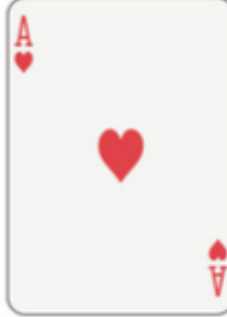

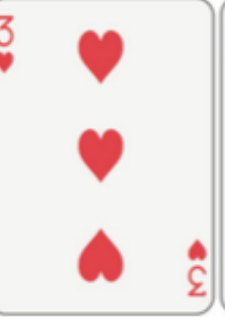
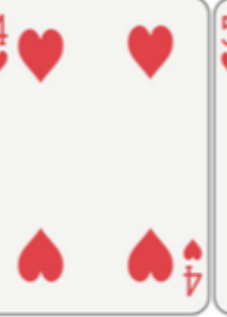









CHAPTER 9

프로젝트 기반 학습 II  
카드게임 블랙잭



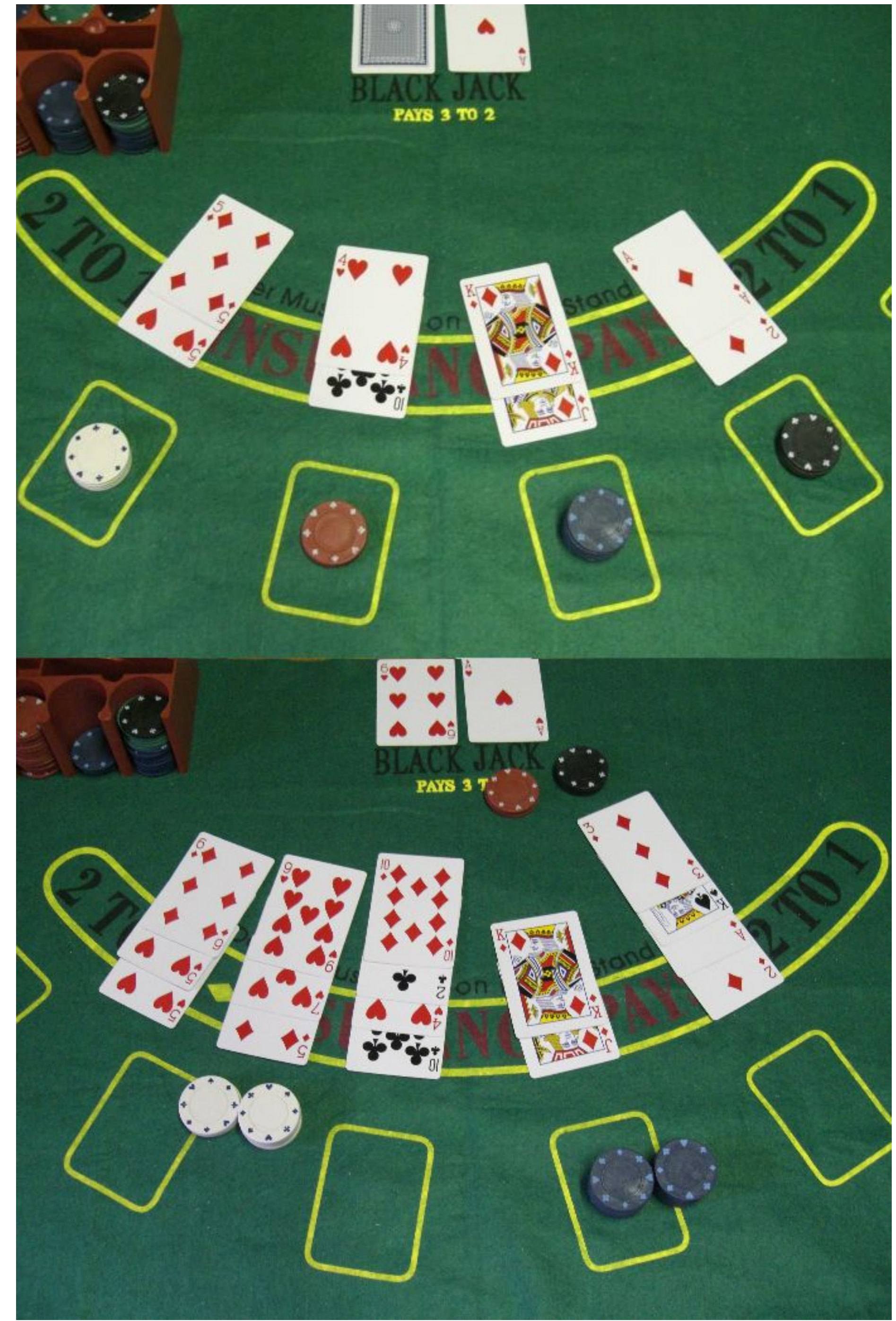
# 블랙잭

## Blackjack

												
"A"	2	3	4	5	6	7	8	9	10	"J"	"Q"	"K"
1										10	10	10
11												



21



# 블랙잭 게임 규칙

Dealer



Player

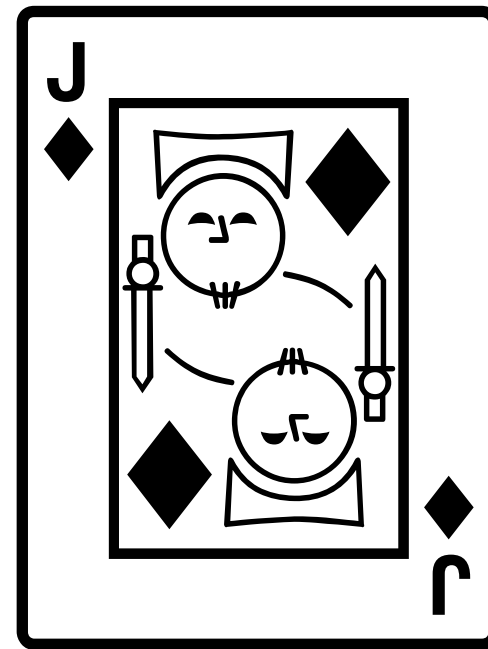


# 블랙잭 게임 규칙

Dealer

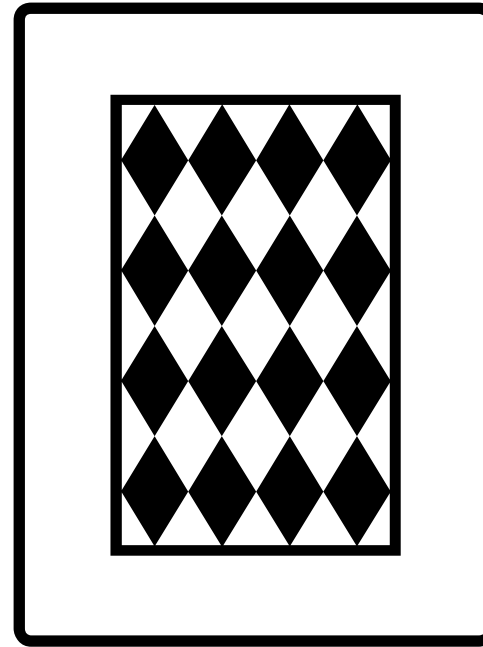


Player

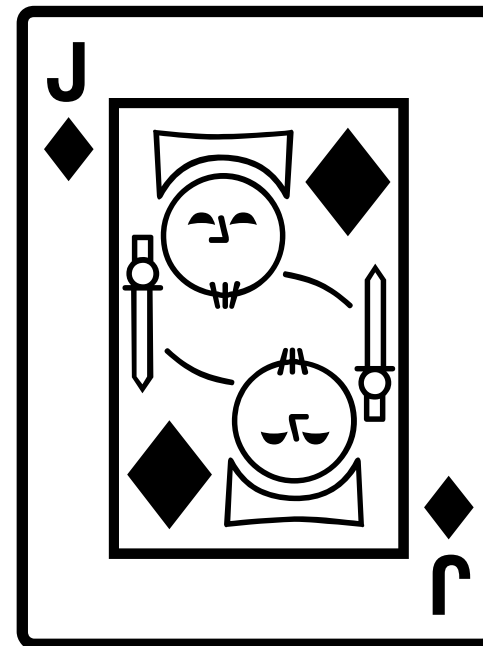


# 블랙잭 게임 규칙

Dealer

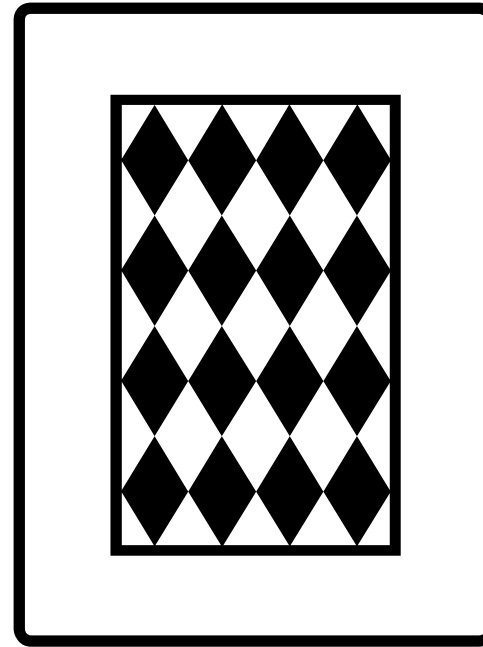


Player

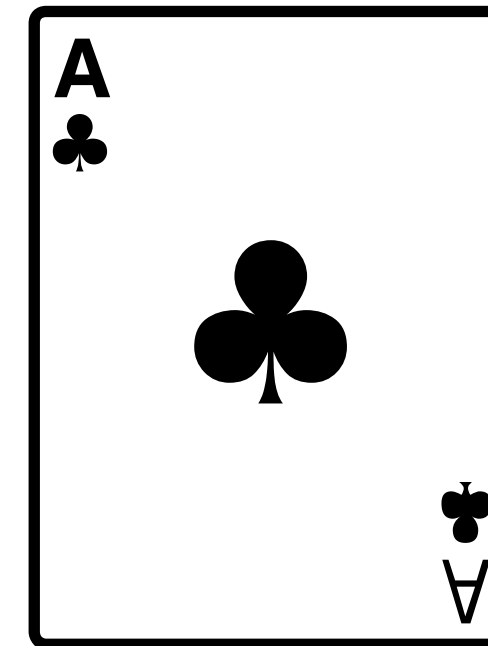
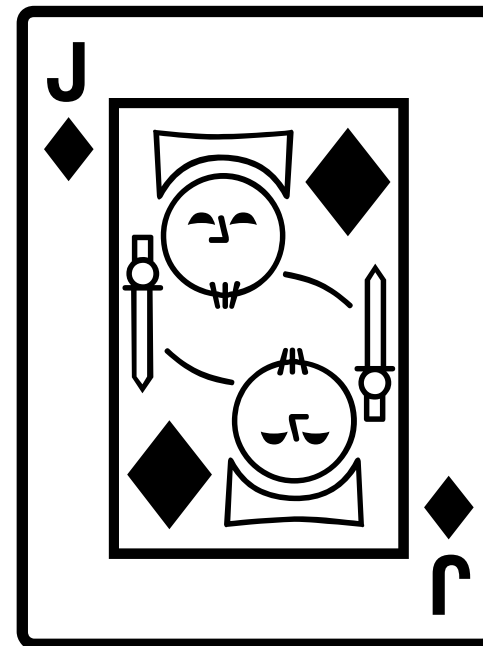


# 블랙잭 게임 규칙

Dealer

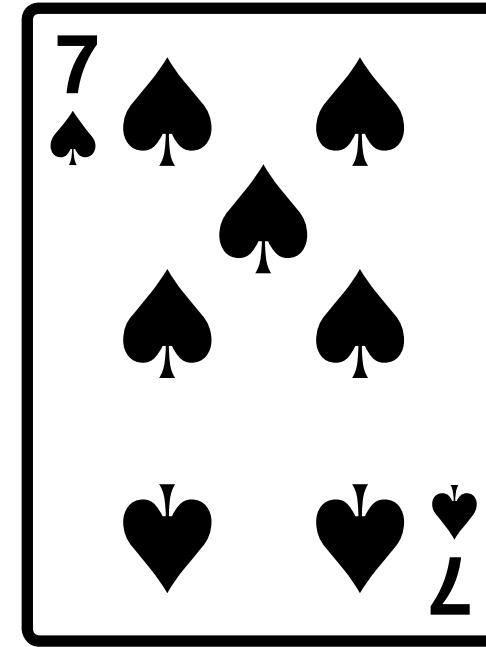
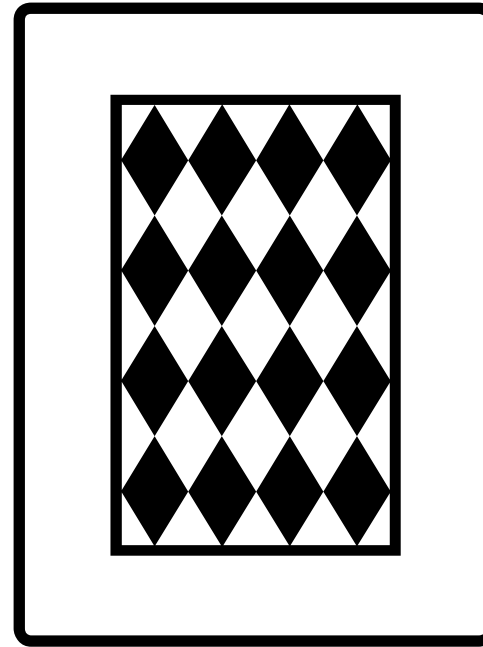


Player

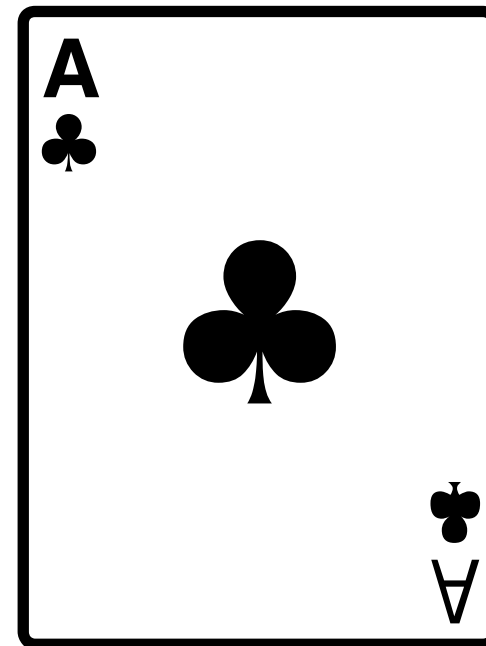
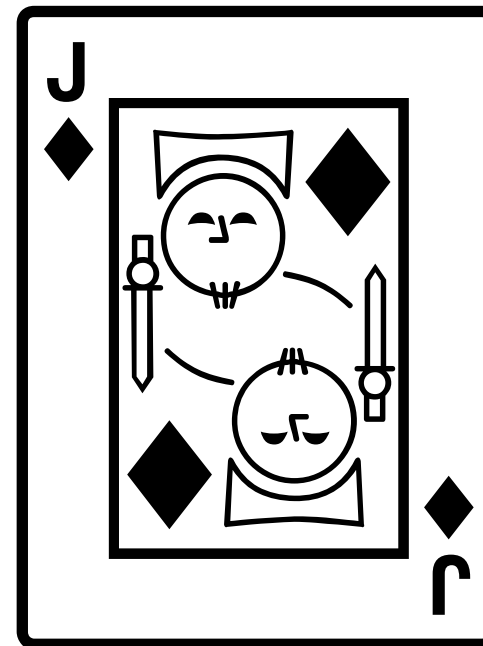


# 블랙잭 게임 규칙

Dealer

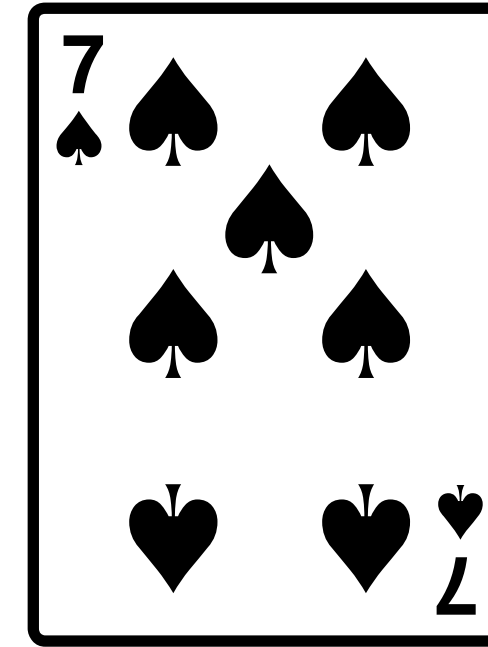
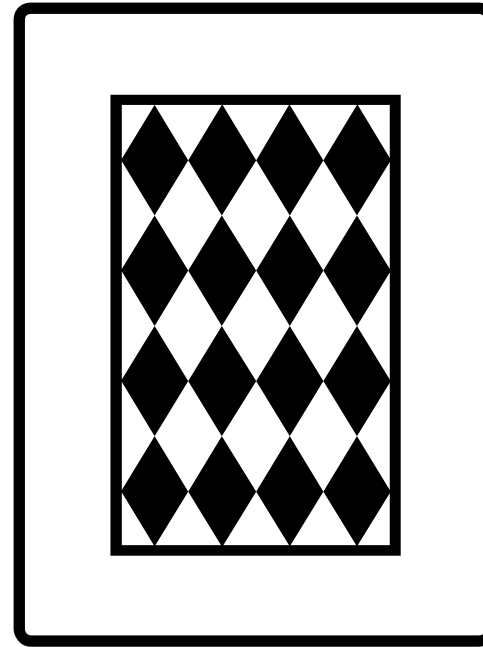


Player



# 블랙잭 게임 규칙

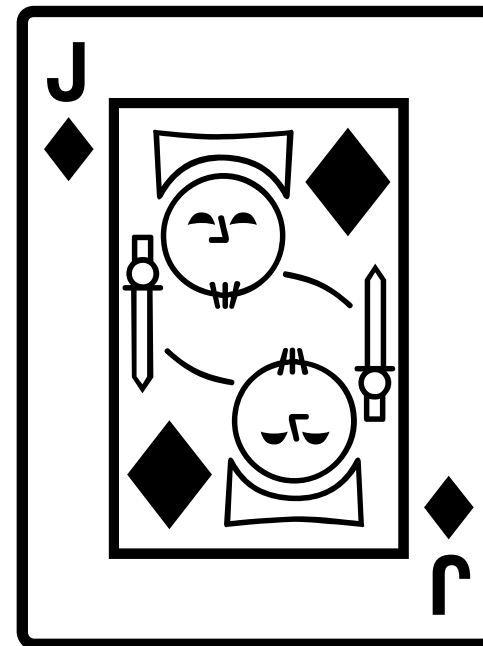
Dealer



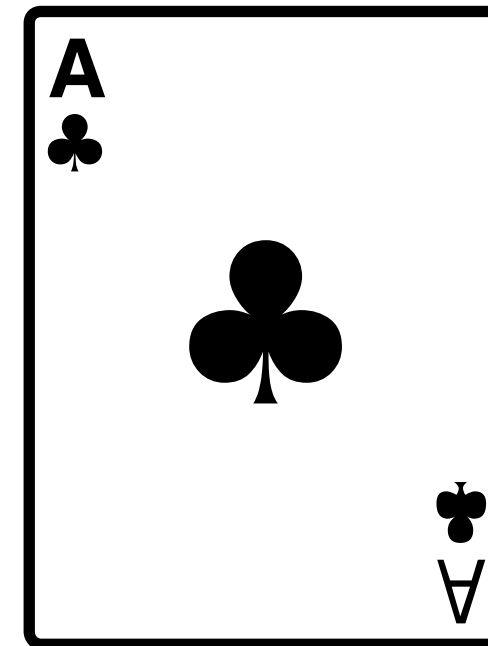
Player



21



10



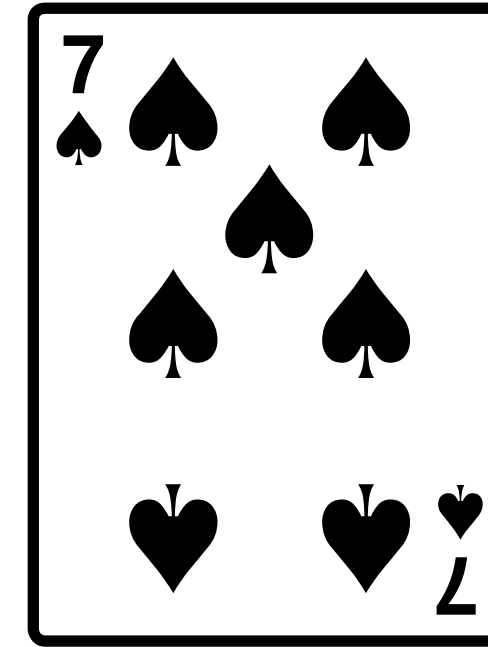
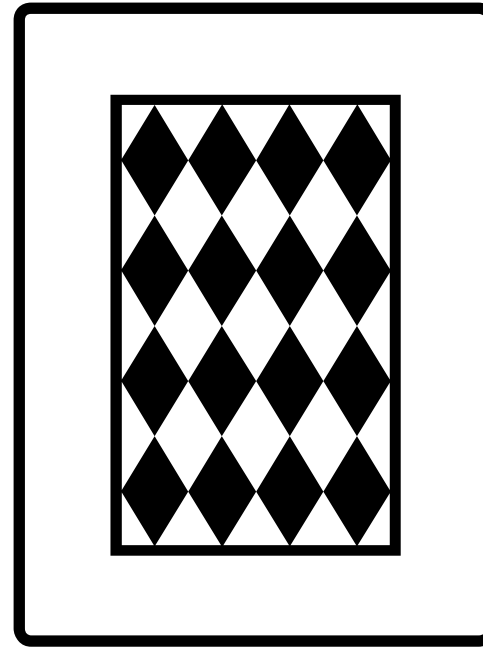
11

Blackjack!



# 블랙잭 게임 규칙

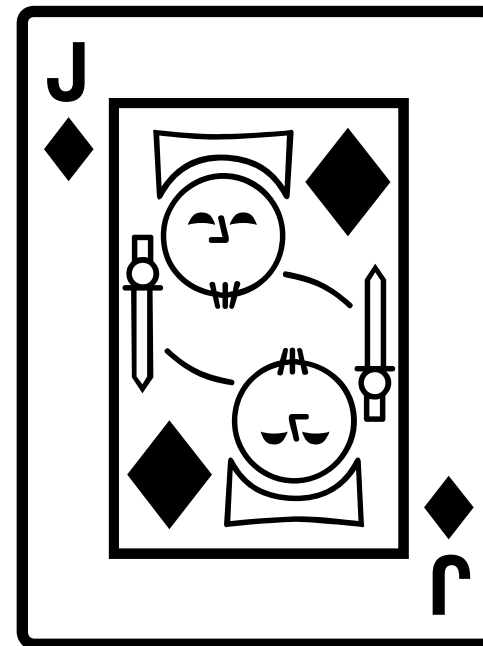
Dealer



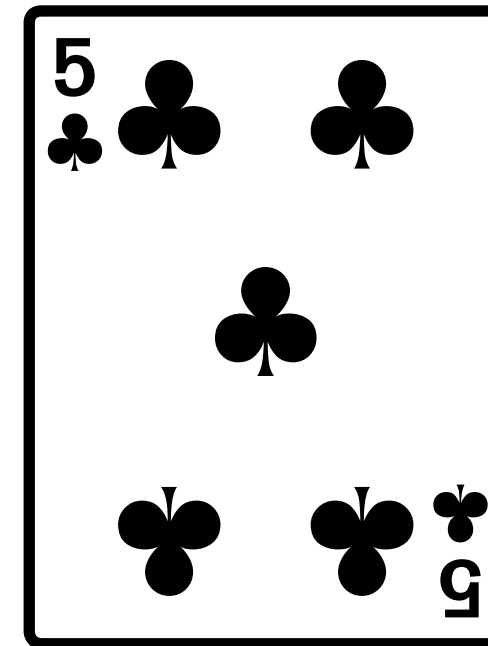
Player



15



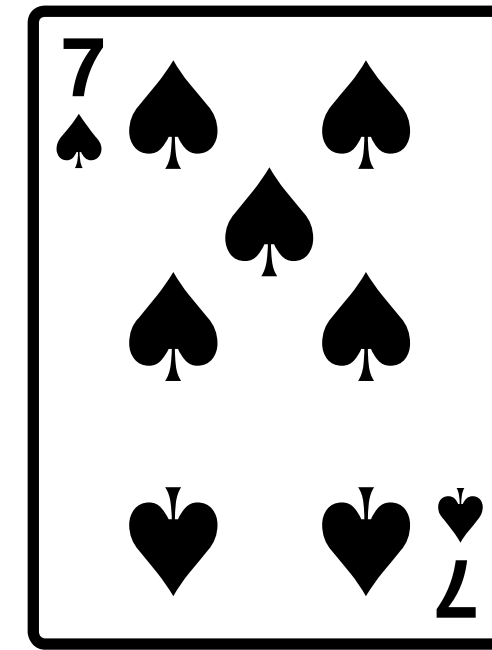
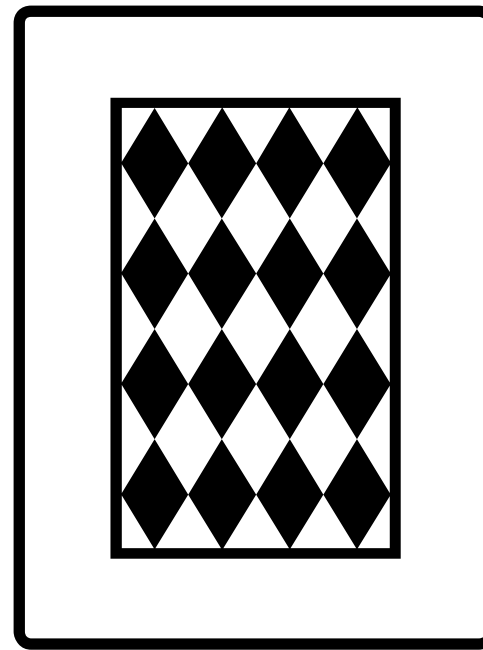
10



5

# 블랙잭 게임 규칙

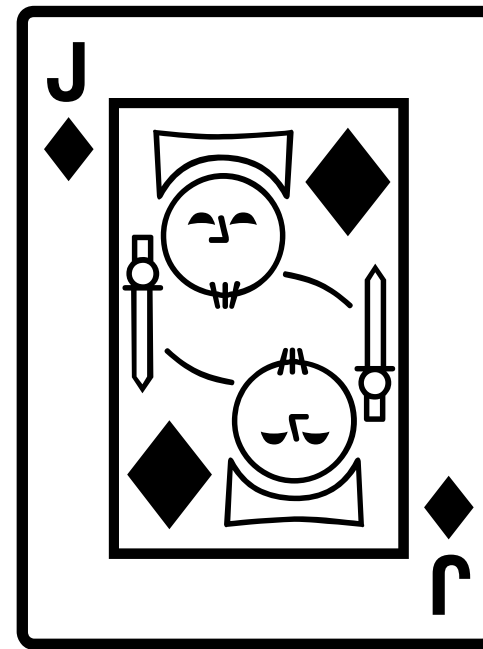
Dealer



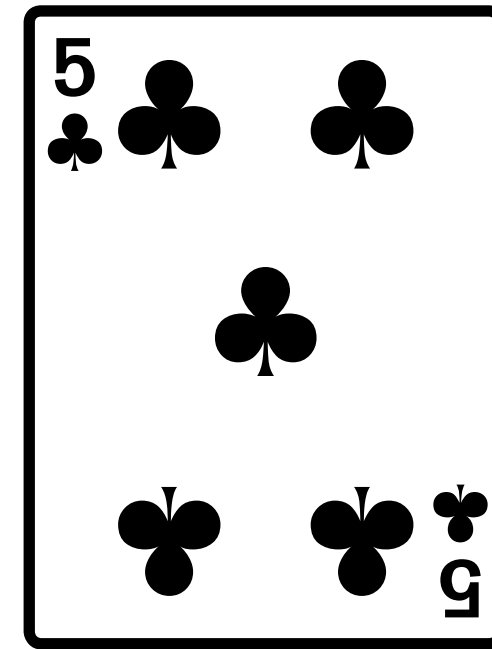
Player



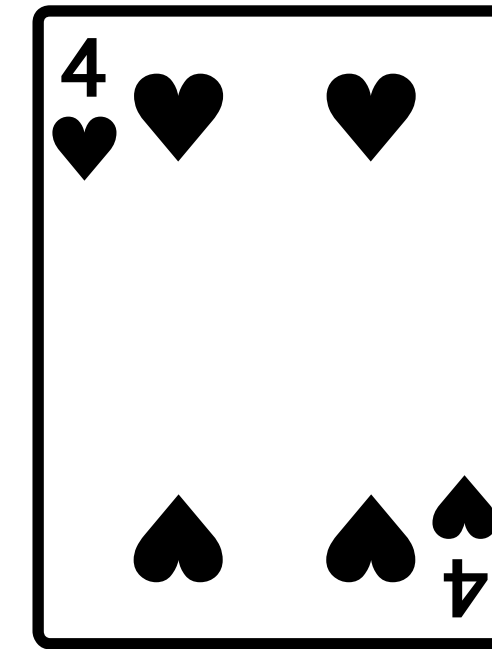
19



10



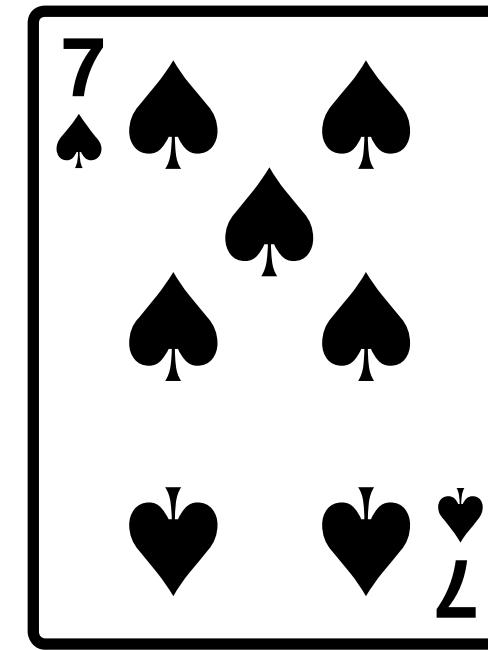
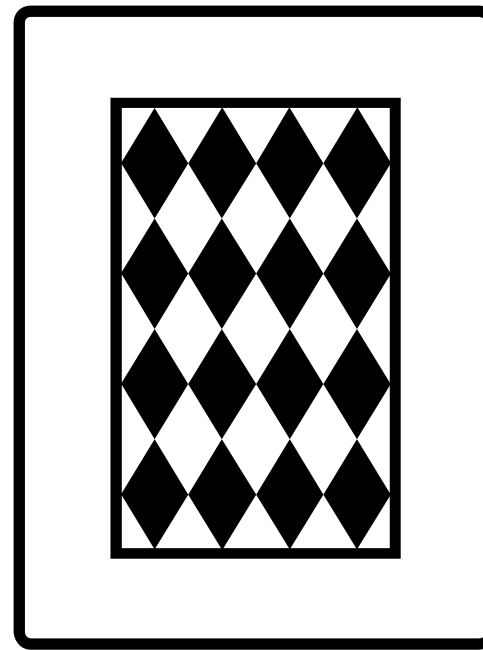
5



4

# 블랙잭 게임 규칙

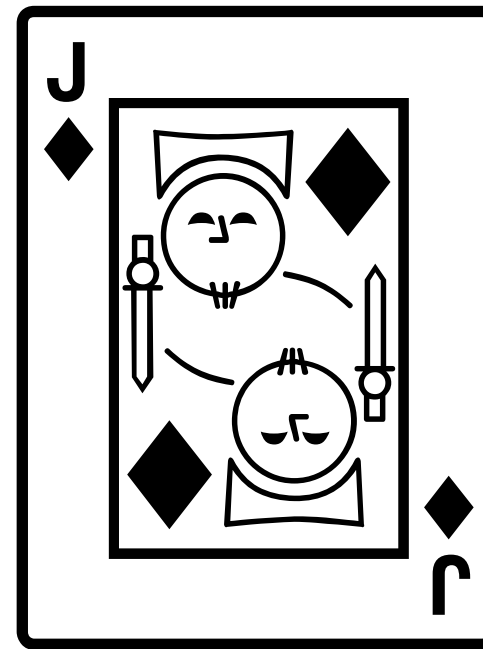
Dealer



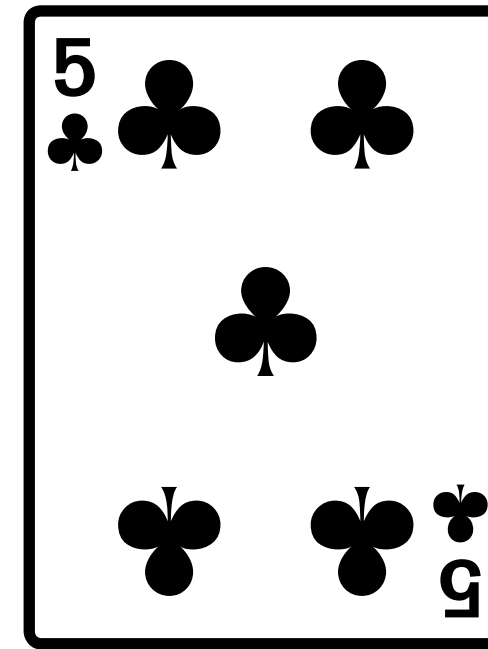
Player



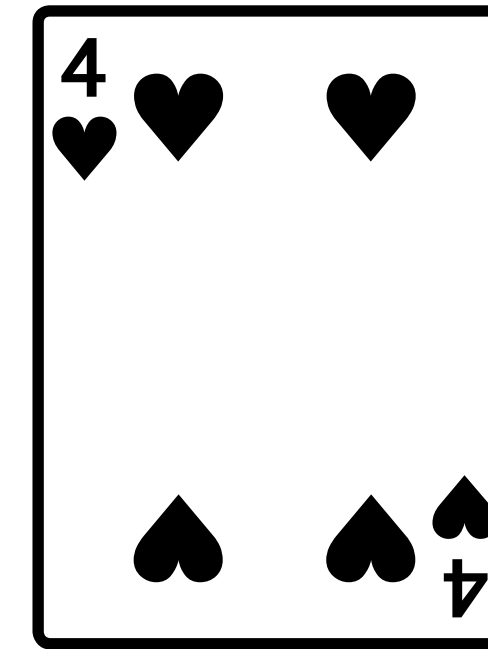
22



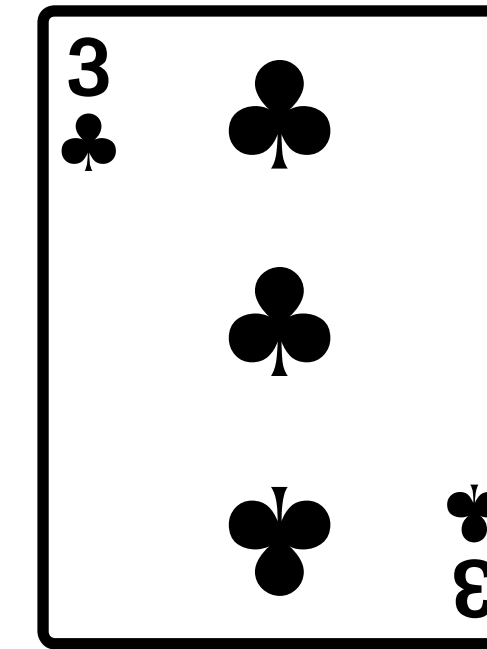
10



5



4

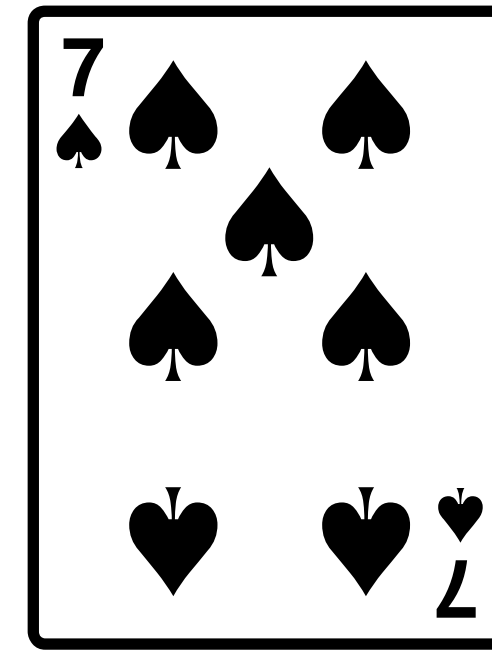
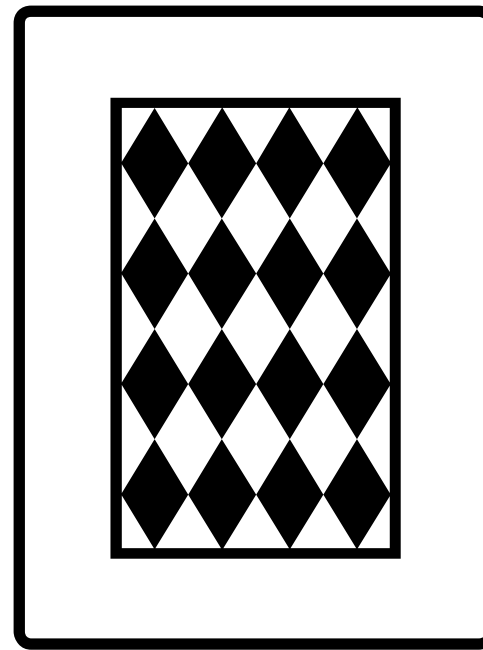


3

Bust!

# 블랙잭 게임 규칙

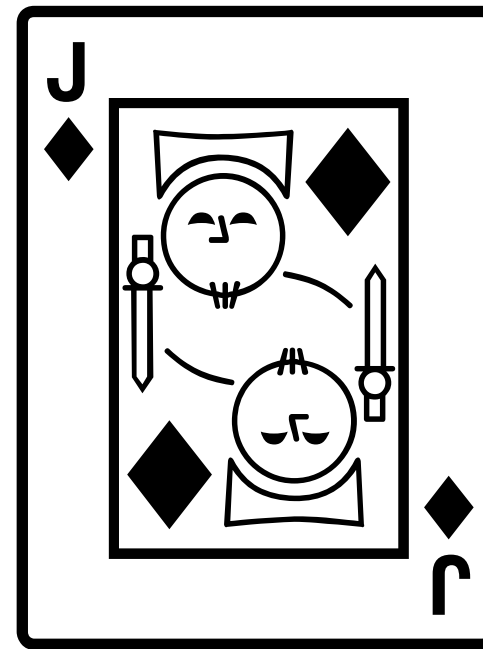
Dealer



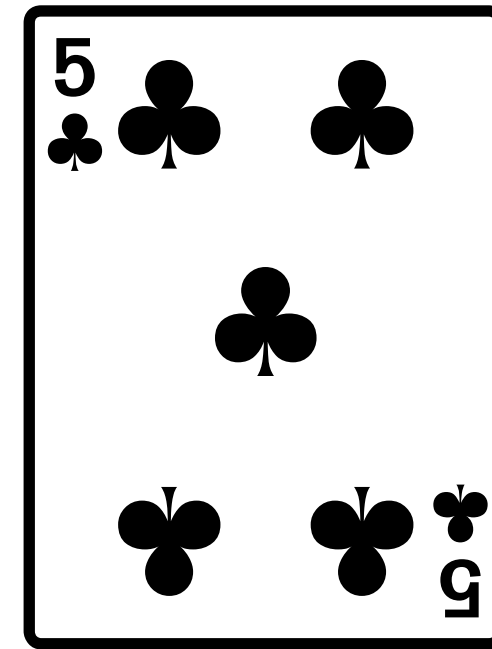
Player



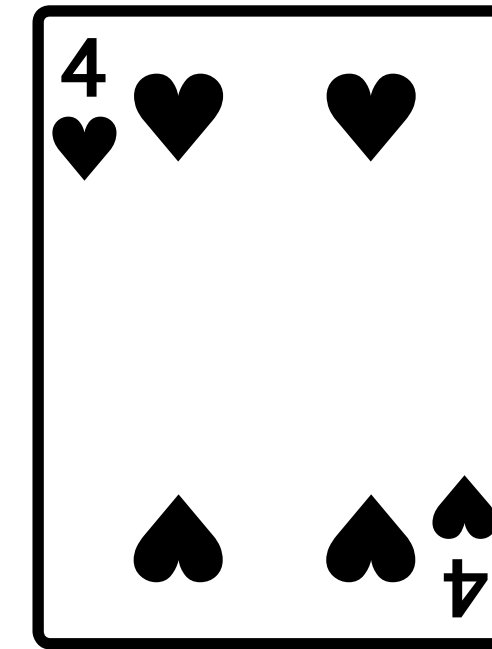
19



10



5



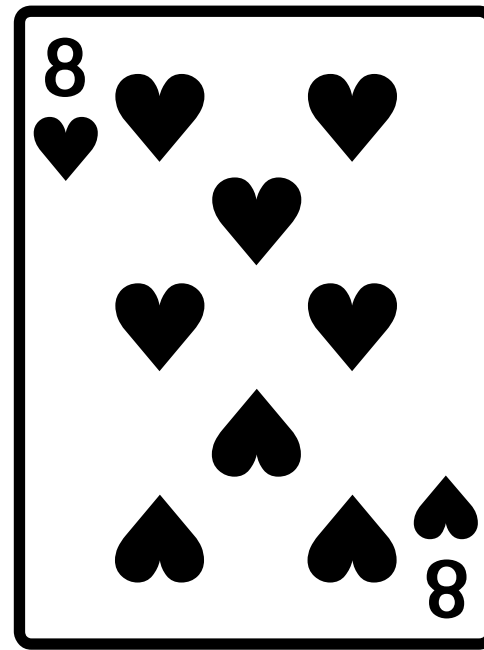
4

# 블랙잭 게임 규칙

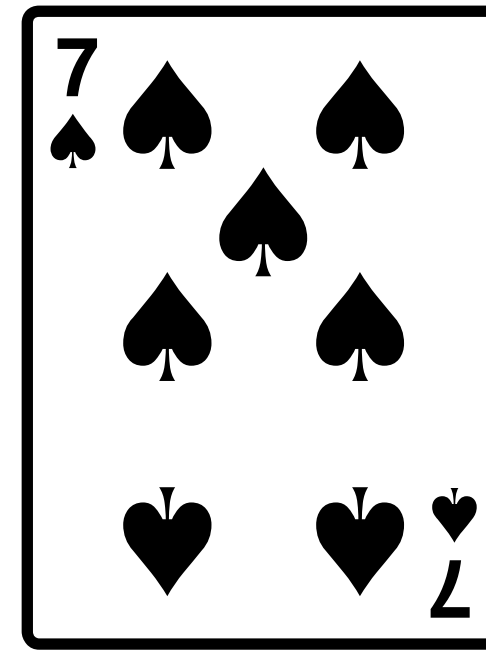
Dealer



15



8

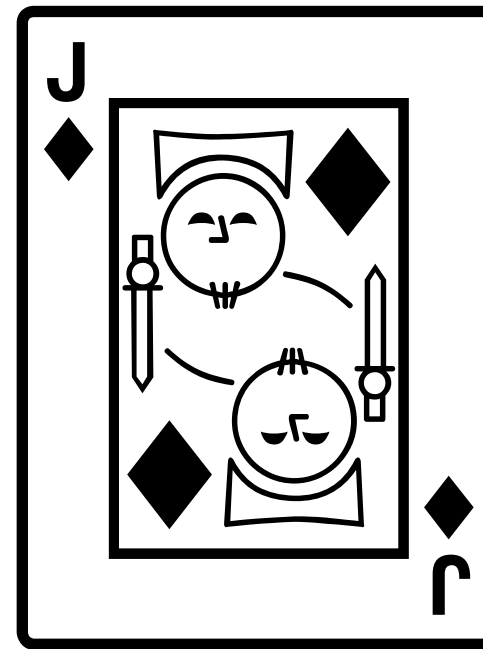


7

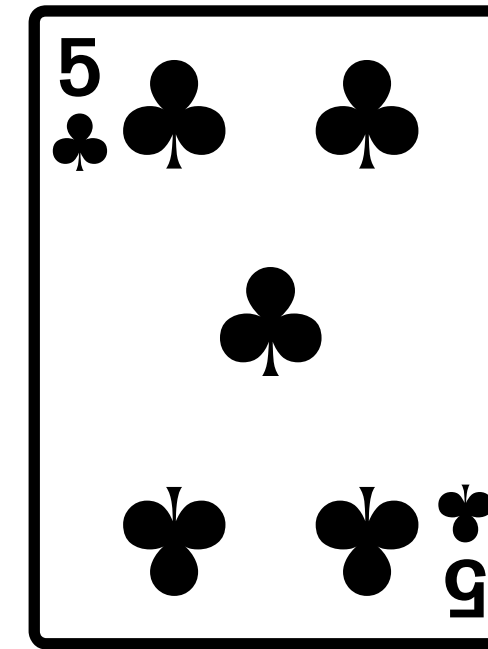
Player



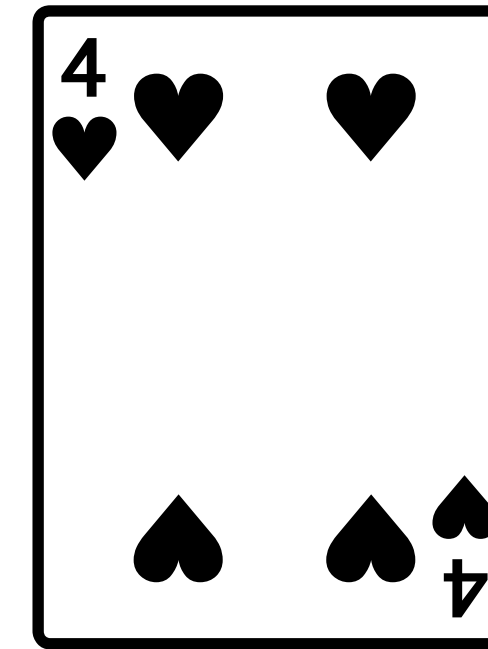
19



10



5



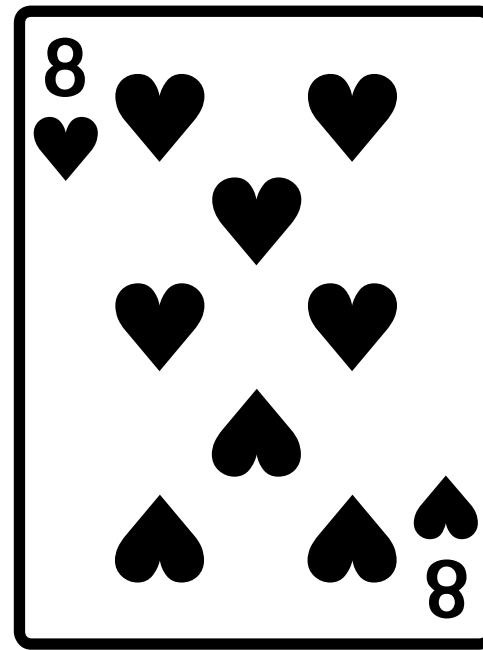
4

# 블랙잭 게임 규칙

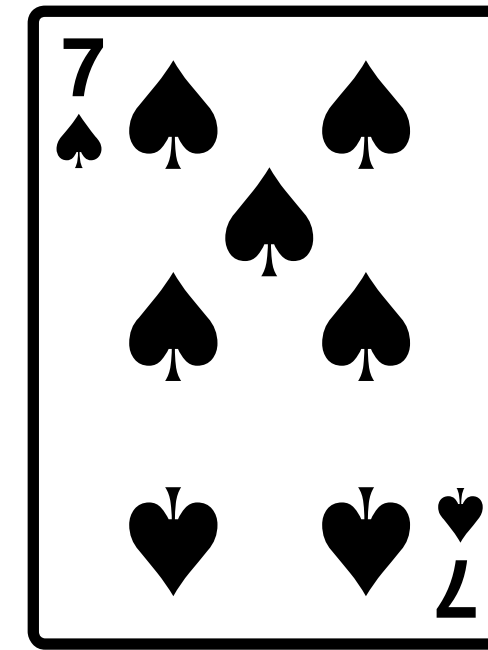
Dealer



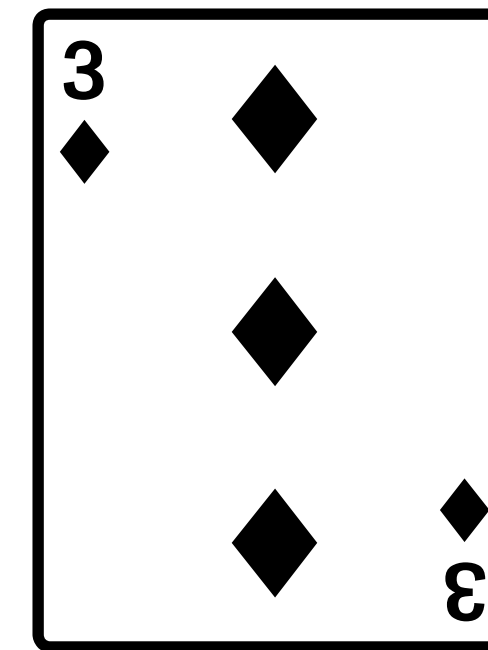
18



8



7

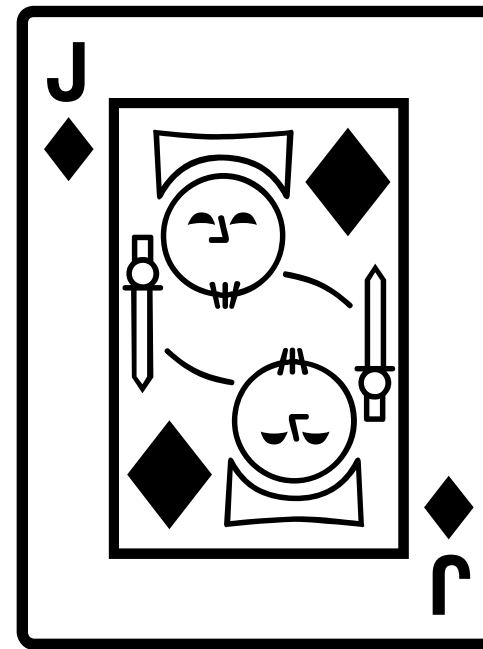


3

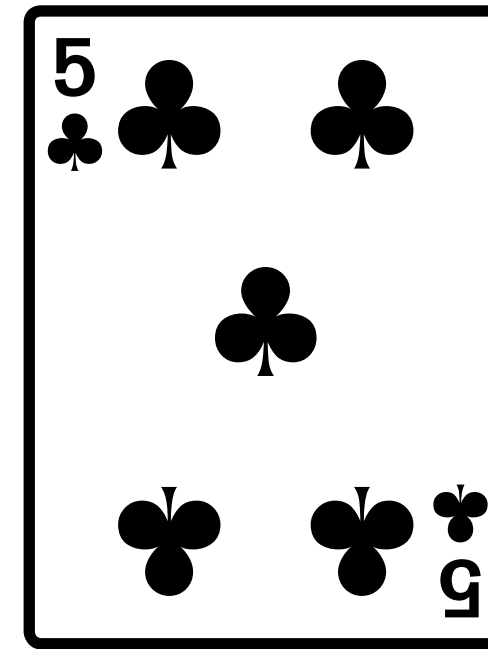
Player



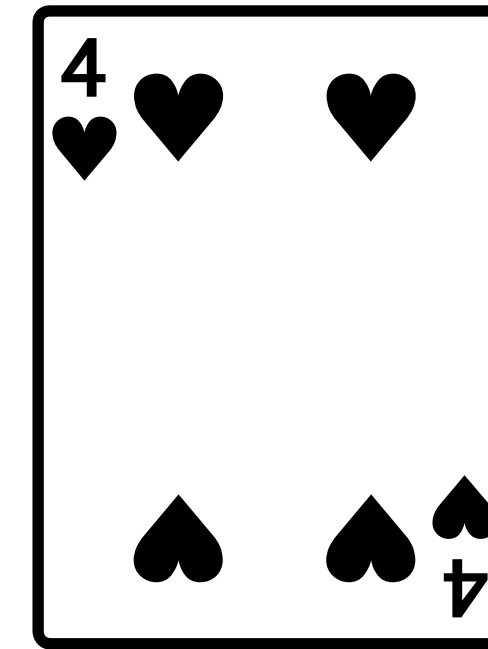
19



10



5



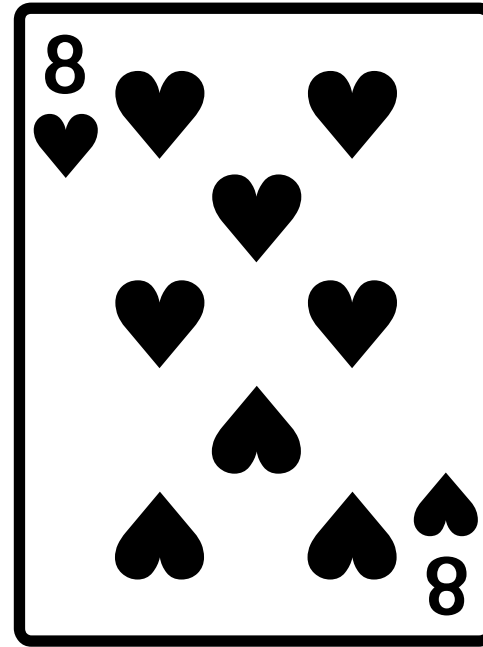
4

# 블랙잭 게임 규칙

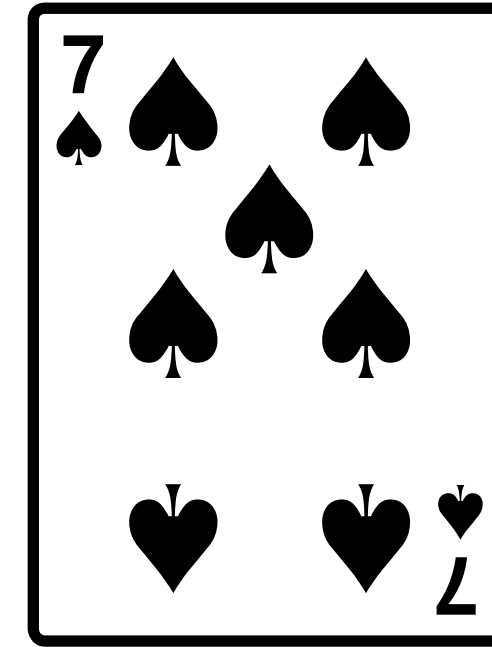
Dealer



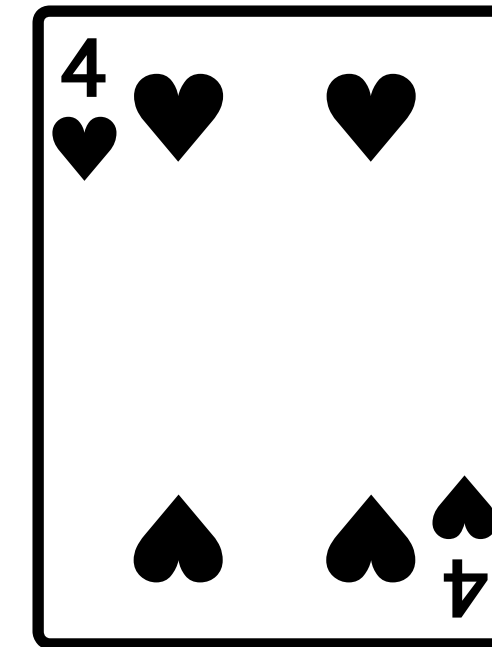
19



8



7

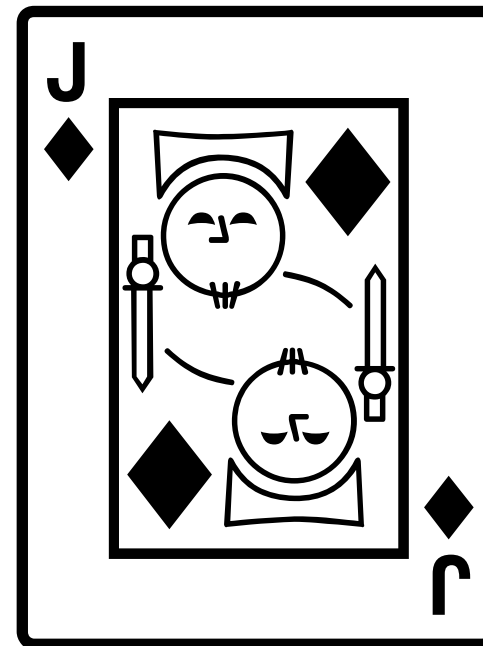


4

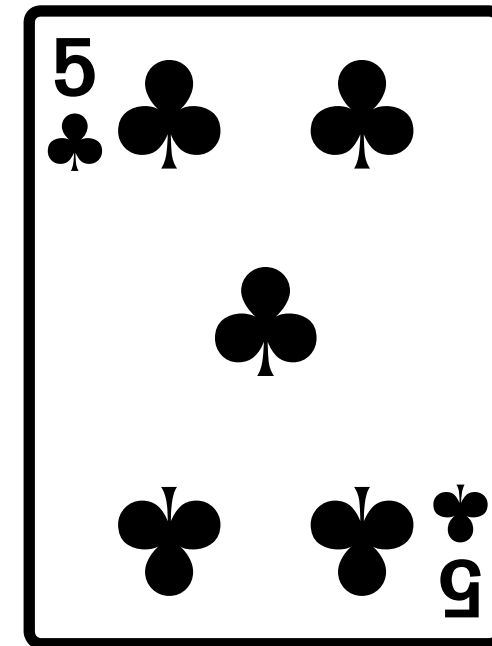
Player



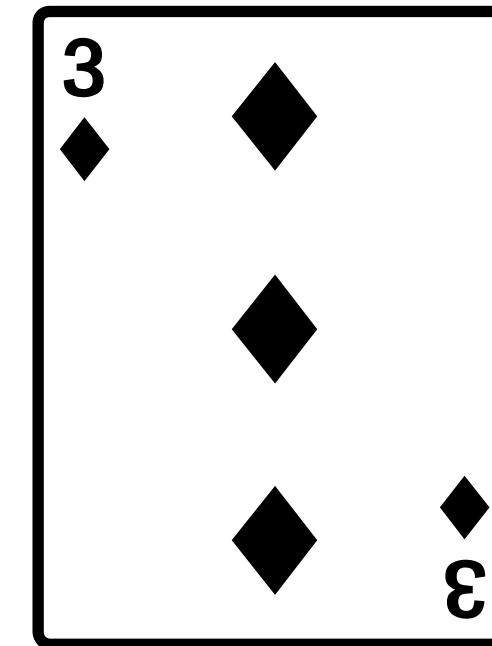
18



10



5



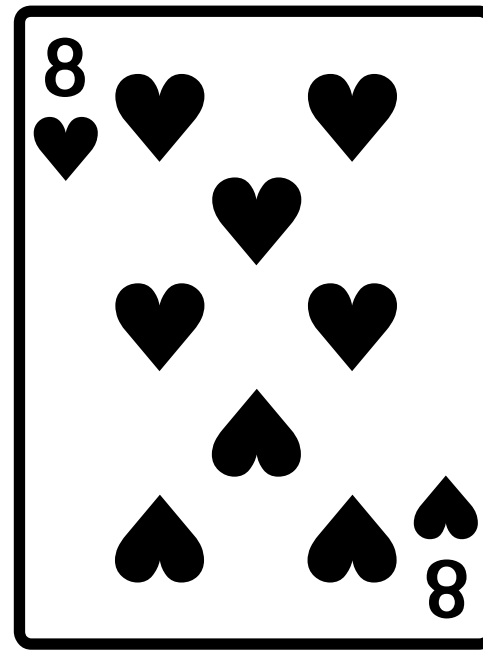
3

# 블랙잭 게임 규칙

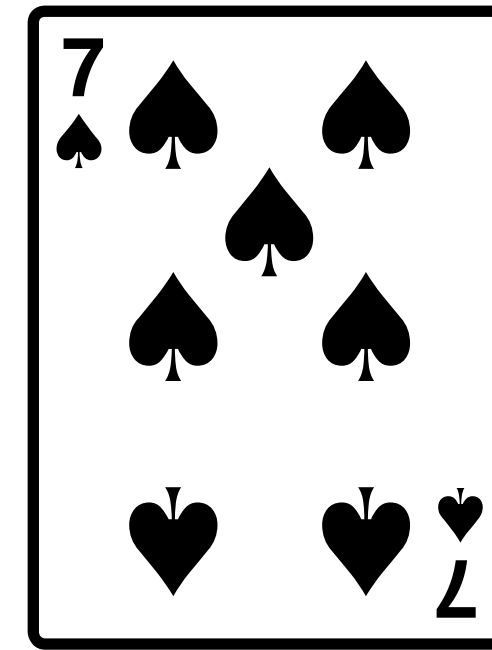
Dealer



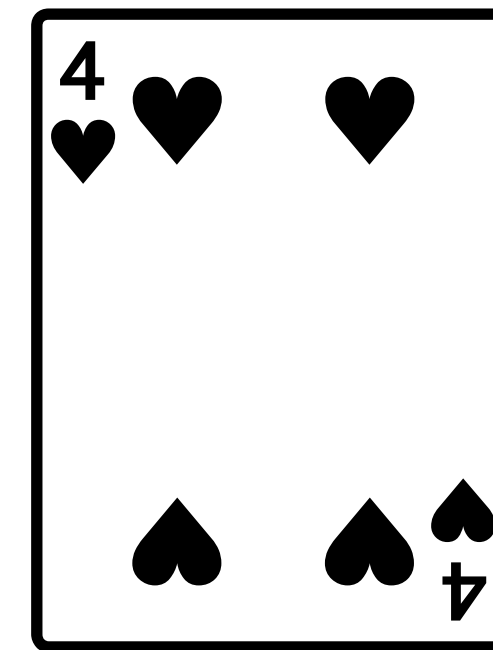
19



8



7

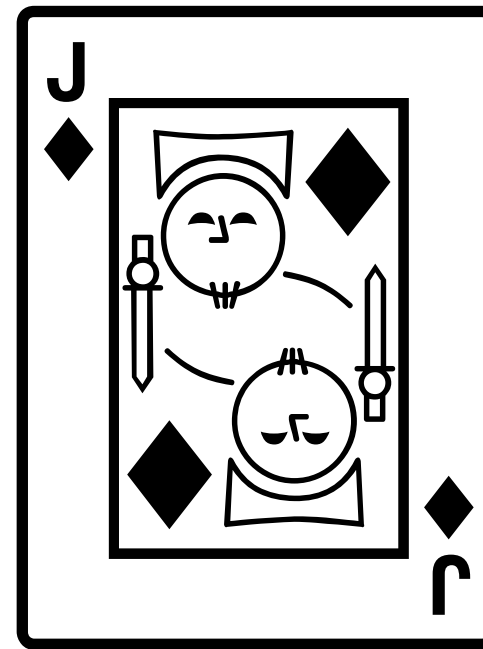


4

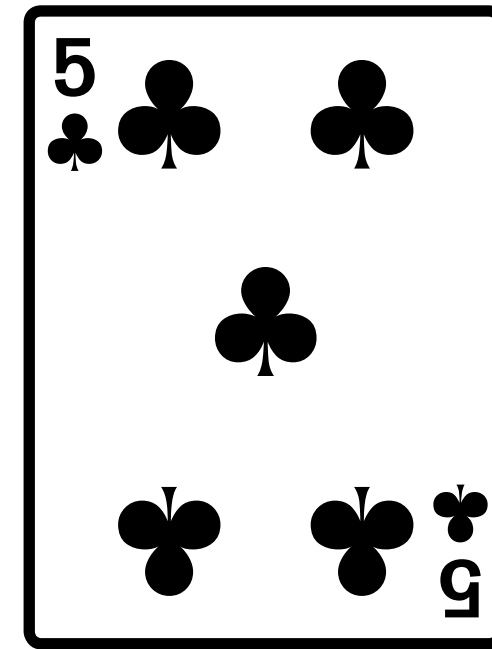
Player



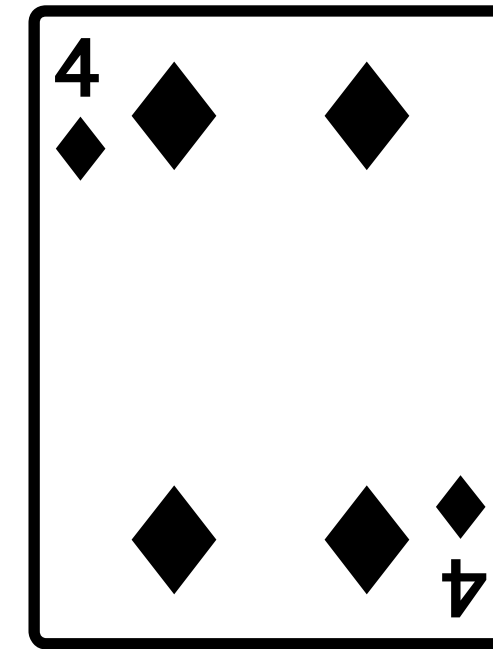
19



10



5



4



# 블랙잭 카드게임 프로젝트

1단계

컴퓨터 딜러를 상대로  
표준 입출력창에서 인터랙티브하게 진행

2단계

멤버십 기능 추가  
게임 기록 영구 보존

프로그래밍의 정석  
파이썬

# 9

## 프로젝트 기반 학습 II 카드게임 블랙잭

9.1 집합 · 9.2 카드게임 API 라이브러리 모듈 · 9.3 프로그래밍 프로젝트 1단계 : 블랙잭  
9.4 디셔너리 · 9.5 프로그래밍 프로젝트 2단계 : 블랙잭 (기능 확장)

### CHAPTER 9

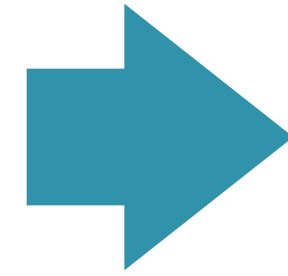
## 프로젝트 기반 학습 II 카드게임 블랙잭

- ✓ 9.1 집합
- 9.2 카드게임 API 라이브러리 모듈
- 9.3 프로그래밍 프로젝트 1단계 : 블랙잭
- 9.4 디셔너리
- 9.5 프로그래밍 프로젝트 2단계 : 블랙잭 (기능 확장)

# 집합

set

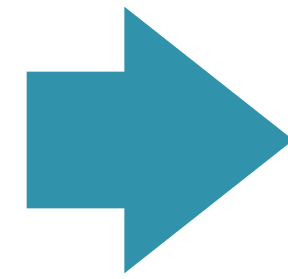
순서와 중복 없이 데이터 값을 모아놓을 수 있는 컬렉션<sup>collection</sup> 데이터 구조



# 집합

## set

순서와 중복 없이 데이터 값을 모아놓을 수 있는 컬렉션<sup>collection</sup> 데이터 구조



집합 메소드	의미
<code>s.add(n)</code>	집합 <code>s</code> 에 원소 <code>n</code> 을 추가한다.
<code>s.remove(n)</code>	집합 <code>s</code> 에서 원소 <code>n</code> 을 제거한다. <code>s</code> 에 <code>n</code> 이 없으면 <code>KeyError</code> 오류가 발생한다.

프로그래밍의 정석  
파이썬

# 9

## 프로젝트 기반 학습 II 카드게임 블랙잭

9.1 집합 · 9.2 카드게임 API 라이브러리 모듈 · 9.3 프로그래밍 프로젝트 1단계 : 블랙잭  
9.4 디셔너리 · 9.5 프로그래밍 프로젝트 2단계 : 블랙잭 (기능 확장)

### CHAPTER 9

## 프로젝트 기반 학습 II 카드게임 블랙잭

- 9.1 집합
- ✓ 9.2 카드게임 API 라이브러리 모듈
- 9.3 프로그래밍 프로젝트 1단계 : 블랙잭
- 9.4 디셔너리
- 9.5 프로그래밍 프로젝트 2단계 : 블랙잭 (기능 확장)

# **API**

**Application Programming Interface**

# 카드 1벌 만들어 무작위로 섞기



"Spade"



"Club"



"Heart"



"Diamond"



"A"

2

3

4

5

6

7

8

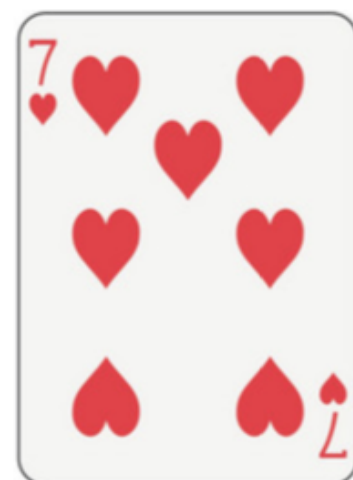
9

10

"J"

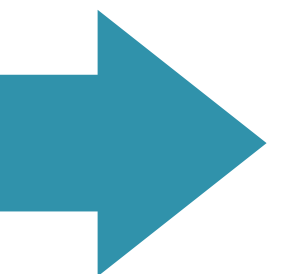
"Q"

"K"



("Heart", 7)

fresh\_deck()





## 실습 9.1 fresh\_deck 함수 만들기

다음 뼈대코드의 비어있는 부분(라인 6~10)에 코드를 채워서 `fresh_deck()` 함수를 완성하자.

code : 9-1.py

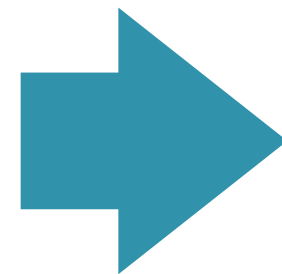
```
1 import random
2 def fresh_deck():
3     suits = {"Spade", "Heart", "Diamond", "Club"}
4     ranks = {2, 3, 4, 5, 6, 7, 8, 9, 10, "J", "Q", "K", "A"}
5     deck = []
6     # Write your nested for-loop creating 52 cards in deck.
7
8
9
10    # shuffle deck using random.shuffle function
11    return deck
```



# 카드 덱에서 카드 한 장 뽑아주기

code : 9-2.py

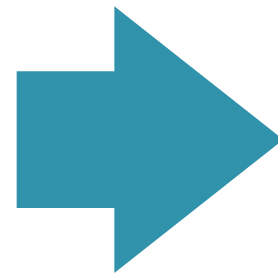
```
1 def hit(deck):  
2     if deck == []:  
3         deck = fresh_deck()  
4     return (deck[0], deck[1:])
```



# 카드 프린트해서 보여주기

code : 9-4.py

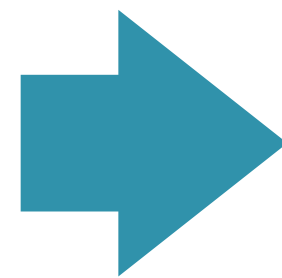
```
1 def show_cards(cards, message):  
2     print(message)  
3     for card in cards:  
4         print(' ', card[0], card[1])
```



# 카드를 더 받을지 물어보기

code : 9-5.py

```
1 def more(message):  
2     answer = input(message)  
3     while not (answer == 'y' or answer == 'n'):  
4         answer = input(message)  
5     return answer == 'y'
```



# 카드 점수 계산하기

pp.426~427



## 실습 9.2 count\_score 함수 만들기

카드 리스트 `cards`를 인수로 받아 카드 점수의 합을 내주는 함수를 만들자.

각 카드의 점수는 다음과 같이 매긴다.

- 2부터 9까지는 액면대로 매긴다.
- "J", "Q", "K"는 10점으로 매긴다.
- "A"는 11점과 1점 중에서 유리한 수를 선택한다. 즉, 11점으로 매겨서 계산하되, 합이 21점이 넘으면 1점으로 매겨 다시 계산한다.



## 실습 9.2 count\_score 함수 만들기

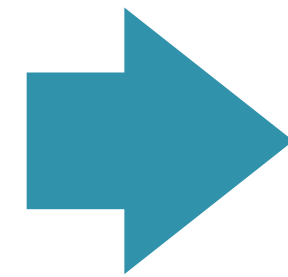
아래 뼈대코드의 빈 공간을 채워서 함수를 완성하자.

code : 9-3.py

```
1 def count_score(cards):
2     score = 0
3     number_of_ace = 0
4     for card in cards:
5         rank = card[1]
6         # accumulate score (counts A as 11)
7
8
9
10
11
12
13     # adjust score if score is over 21 and there is A
14     # there may be two or more A's
15
16     return score
```

# 모듈

함수를 모아둔 파일.



프로그래밍의 정석  
파이썬

# 9

## 프로젝트 기반 학습 II 카드게임 블랙잭

9.1 집합 · 9.2 카드게임 API 라이브러리 모듈 · 9.3 프로그래밍 프로젝트 1단계 : 블랙잭  
9.4 디셔너리 · 9.5 프로그래밍 프로젝트 2단계 : 블랙잭 (기능 확장)

### CHAPTER 9

## 프로젝트 기반 학습 II 카드게임 블랙잭

### 9.1 집합

### 9.2 카드게임 API 라이브러리 모듈

### ✓ 9.3 프로그래밍 프로젝트 1단계 : 블랙잭

### 9.4 디셔너리

### 9.5 프로그래밍 프로젝트 2단계 : 블랙잭 (기능 확장)





# 블랙잭 알고리즘

1. 환영인사를 프린트한다.

```
print("Welcome to Softopia Casino")
```

2. 잘 섞은 카드 1벌을 준비한다.

```
deck = fresh_deck()
```

3. 칩의 개수를 0으로 초기화한다.

```
chips = 0
```

4. 손님이 원하는 한, 단계 5~14을 반복한다.

5. 카드를 1장씩 손님, 딜러, 손님, 딜러 순으로 배분한다.

```
dealer = [ ]
```

```
player = [ ]
```

```
card, deck = hit(deck) # 1장 뽑아서
```

```
player.append(card) # 손님에게 주고
```

```
card, deck = hit(deck) # 1장 뽑아서
```

```
dealer.append(card) # 딜러에게 주고
```

```
card, deck = hit(deck) # 1장 뽑아서
```

```
player.append(card) # 손님에게 주고
```

```
card, deck = hit(deck) # 1장 뽑아서
```

```
dealer.append(card) # 딜러에게 준다.
```

6. 딜러의 첫 카드를 제외하고 모두 보여준다.

```
print("My cards are:")
```

```
print(" ", "*****", "***")
```

```
print(" ", dealer[1][0], dealer[1][1])
```

## 블랙잭 알고리즘 (계속)

7. 손님 카드 보여준다.

```
show_cards(player, "Your cards are:")
```

8. 손님과 딜러의 카드 두 장의 합을 각각 계산한다.

```
score_player = count_score(player)
```

```
score_dealer = count_score(dealer)
```

9. 손님 카드의 합 `score_player`가 21이면 블랙잭으로 손님이 이긴다. `chips`에 2를 더한다.

10. 손님 카드 합이 21을 넘지 않는 한 손님이 원하면 카드를 더 준다. 21을 넘으면 손님이 버스트되어 딜러가 이기고 `chips`에서 1을 뺀다.

11. 손님 카드 합이 21을 넘지 않았으면, 딜러의 차례이다. 딜러의 카드 합을 계산하여 16 이하이면 16이 넘을 때까지 무조건 카드를 더 받고, 17 이상이 되는 순간 더 이상 받지 않는다.

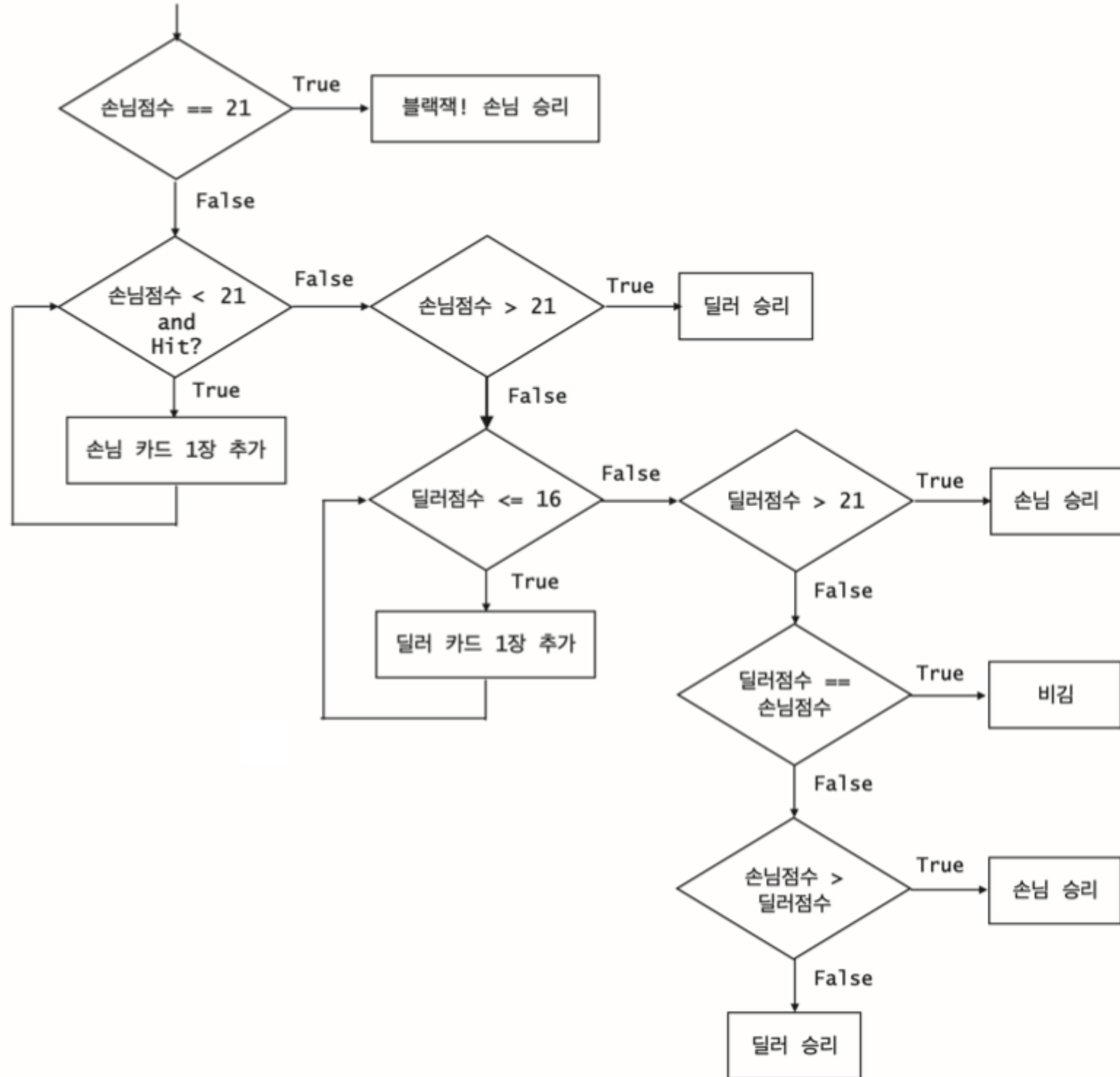
12. 딜러의 카드 합이 21을 넘으면 딜러가 버스트 되어 손님이 이기고 `chips`에 1을 더한다.

13. 둘 다 21이 넘지 않으면 큰 쪽이 이긴다. 손님이 이기면 `chips`에 1을 더하고, 딜러가 이기면 `chips`에서 1을 빼고, 비기면 변동 없다.

14. 더 할지 손님에게 물어봐서 그만하길 원하면 끝낸다.

# 블랙잭 알고리즘 (흐름도)

알고리즘의 단계 9~13을 흐름도로 그리면 다음과 같다.



프로그래밍의 정석  
파이썬

# 9

## 프로젝트 기반 학습 II 카드게임 블랙잭

9.1 집합 · 9.2 카드게임 API 라이브러리 모듈 · 9.3 프로그래밍 프로젝트 1단계 : 블랙잭  
9.4 디셔너리 · 9.5 프로그래밍 프로젝트 2단계 : 블랙잭 (기능 확장)

### CHAPTER 9

## 프로젝트 기반 학습 II 카드게임 블랙잭

### 9.1 집합

### 9.2 카드게임 API 라이브러리 모듈

### 9.3 프로그래밍 프로젝트 1단계 : 블랙잭

### ✓ 9.4 디셔너리

### 9.5 프로그래밍 프로젝트 2단계 : 블랙잭 (기능 확장)

# 딕셔너리

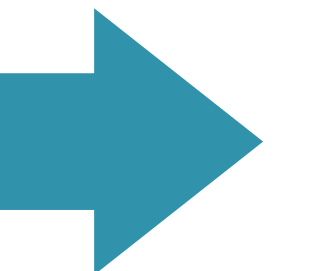
Dictionary

dict

키와 값의 매핑(mapping) 모아놓을 수 있는 컬렉션(collection) 데이터 구조

```
{'이름': '조상만', '생년': 2002, '이메일': 'chosm@softopia.co.kr'}
```

"이름"	"생년"	"이메일"
"조상만"	2002	"chosm@softopia.co.kr"



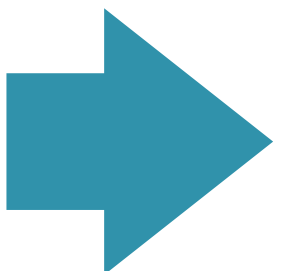
# 딕셔너리

## Dictionary

### dict

키와 값의 매핑(mapping), 모아놓을 수 있는 컬렉션(collection) 데이터 구조

딕셔너리 메소드	실행 의미
<code>d.get(key)</code>	키 <code>key</code> 가 딕셔너리 <code>d</code> 에 있으면 그 값을 리턴하고, 없으면 <code>None</code> 을 리턴한다.
<code>d.get(key, default)</code>	키 <code>key</code> 가 딕셔너리 <code>d</code> 에 있으면 그 값을 리턴하고, 없으면 <code>default</code> 를 리턴한다.



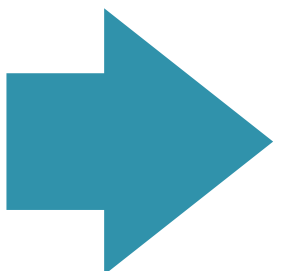
# 딕셔너리

## Dictionary

### dict

키와 값의 매핑<sup>mapping</sup>, 모아놓을 수 있는 컬렉션<sup>collection</sup> 데이터 구조

딕셔너리 메소드	실행 의미
<code>d.keys()</code>	딕셔너리 <code>d</code> 의 키를 리스트로 모아 뷰 객체를 만들어 리턴한다.
<code>d.values()</code>	딕셔너리 <code>d</code> 의 값을 리스트로 모아 뷰 객체를 만들어 리턴한다.
<code>d.items()</code>	딕셔너리 <code>d</code> 의 아이템을 리스트로 모아 뷰 객체를 만들어 리턴한다. 여기서 아이템은 (키, 값) 튜플을 말한다.



프로그래밍의 정석  
파이썬

# 9

## 프로젝트 기반 학습 II 카드게임 블랙잭

9.1 집합 · 9.2 카드게임 API 라이브러리 모듈 · 9.3 프로그래밍 프로젝트 1단계 : 블랙잭  
9.4 디셔너리 · 9.5 프로그래밍 프로젝트 2단계 : 블랙잭 (기능 확장)

### CHAPTER 9

## 프로젝트 기반 학습 II 카드게임 블랙잭

9.1 집합

9.2 카드게임 API 라이브러리 모듈

9.3 프로그래밍 프로젝트 1단계 : 블랙잭

9.4 디셔너리

✓ 9.5 프로그래밍 프로젝트 2단계 : 블랙잭 (기능 확장)



확장한 블랙잭 알고리즘은 다음과 같다. 추가된 부분인 2와 15~17을 제외하고 알고리즘의 다른 부분은 앞과 거의 동일하다.

1. 환영인사를 프린트한다.

```
print("Welcome to Softopia Casino")
```

2. `members.csv` 파일에서 멤버 기록을 읽고 로그인 절차를 통해서 사용자이름, 게임시도 횟수, 이긴 횟수, 칩 보유개수, 전체 멤버 디셔너리 정보를 수집한다.

```
username, tries, wins, chips, members = login(load_members())
```

3. 잘 섞은 카드 1벌을 준비한다.

```
deck = fresh_deck()
```

15. 게임이 진행되는 동안 승패 횟수와 칩의 획득 개수를 추적하여, 게임이 끝난 뒤 결과를 멤버 디셔너리에 적용하여 수정하고, `members.csv` 파일에 저장한다.

16. 해당 세션의 게임 결과를 다음과 같이 요약하여 보여준다.

```
You played 21 games and won 11 of them.  
Your winning pecentage today is 52.4%
```

17. 지금까지의 칩 최다 보유 멤버 5명까지 보여준다.

```
show_top5(members)
```

# 프로그래밍 프로젝트 2단계 : 블랙잭 (기능 확장)

pp.444~447



**실습 9.4** 파일에서 게임 기록 정보 읽기 함수



**실습 9.5** 파일에 게임 기록 정보 쓰기 함수

# 프로그래밍 프로젝트 2단계 : 블랙잭 (기능 확장)

pp.448~449



**실습 9.6** 로그인 함수

# 프로그래밍 프로젝트 2단계 : 블랙잭 (기능 확장)

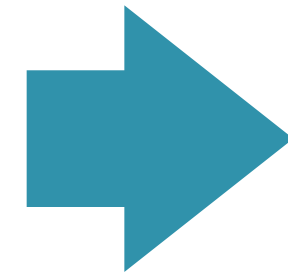
pp.450~452



**실습 9.7** Top 5 보여주기 함수

# 딕셔너리 정렬하기

**sorted**



# 프로그래밍 프로젝트 2단계 : 블랙잭 (기능 확장)

pp.452~455



**실습 9.8**    **블랙잭 확장 버전 완성**

>>>>>>>>>>> 제어 구조의 설계 원리를 중심으로 배우는 >>>>>>>>>>>>

# 프로그래밍의 정석

# 과이썬

도경구 지음



CHAPTER 9

프로젝트 기반 학습 II

카드게임 블랙잭