



파이썬과 친해지기

turtle 모듈, 딕셔너리(사전), 딕셔너리 심화

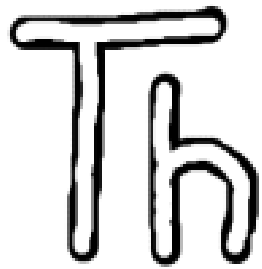
8주차

복습

모듈 = “코드 뭉치”

어떤 역할을 해주는 코드 묶음

“dog 모듈”



dog.py

저는 짖을 줄 알아요!

복습

import A

“A”라는 모듈을 가져와 쓴다.

```
>>> import dog
```

왈왈! (만나서 반가워요!)

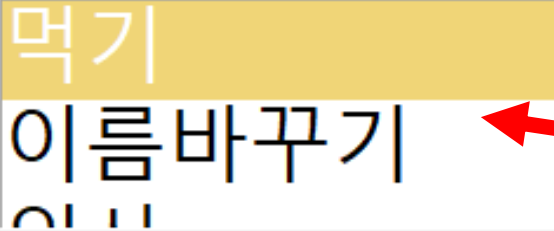
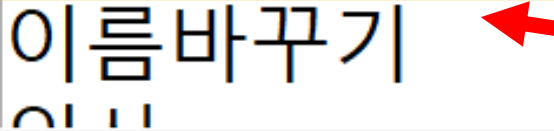
```
>>> dog.짖기()
```

으르릉... 킁킁!

복습

```
>>> dog.짖기()  
으르릉... 쿵쿵!  
'쿠키'은(는) 짖어댔다.
```

앞에 모듈 이름 + “.” 을 붙이자.
dog의 짖기() 함수를 부른 것!

```
>>>  
>>>   
>>>   
>>> dog.|
```

모듈 이름 + “.” 을 쓰고
〈Tab〉을 누르면 모듈이
할 수 있는 것들이 뜬다.

복습

```
>>> import random
```

```
>>> random.random() 0~1 사이의 랜덤한 수를 뽑음.  
0.11220638866309429
```

```
>>> random.randrange(1,10) 1 이상, 10 미만의  
8 정수 중 하나를 랜덤하게 뽑음.
```

```
>>> random.choice(["앞면", "뒷면"])  
'뒷면'
```

어떤 리스트 안에 있는 것을 랜덤하게 하나 뽑아 뽑음.

import A as B

import A as B

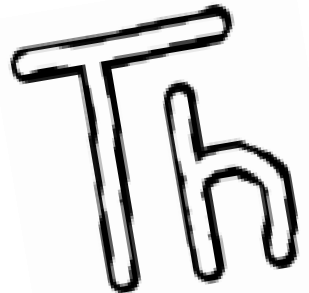
A라는 모듈을 가져와 쓰지만 이름만 B로 바꾼다.

```
>>> import random as r
>>> r.choice("안녕하세요?")
'요'
```

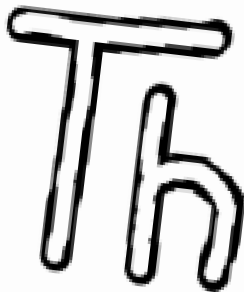
모듈


random.py

숫자를 임의로 뽑는다던지,
뭔가 랜덤한 일을 할 수 있는 모듈!


math.py

복잡한 수학 계산을 대신해주는 모듈!
제곱, 삼각함수 등...

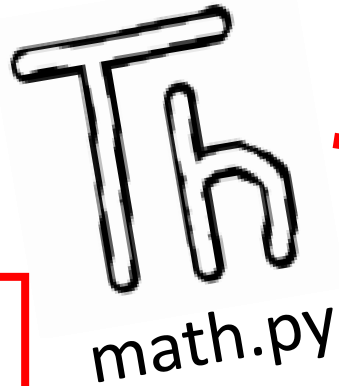

turtle.py

거북이로 그림을 그려주는 모듈?

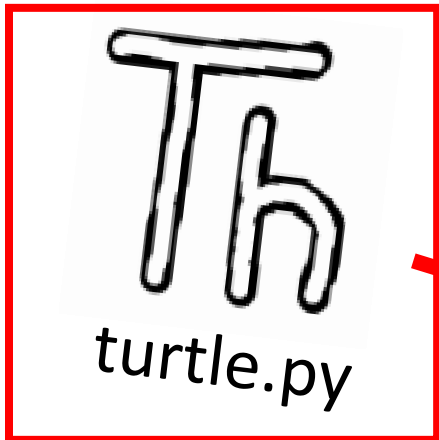
모듈



숫자를 임의로 뽑는다던지,
뭔가 랜덤한 일을 할 수 있는 모듈!



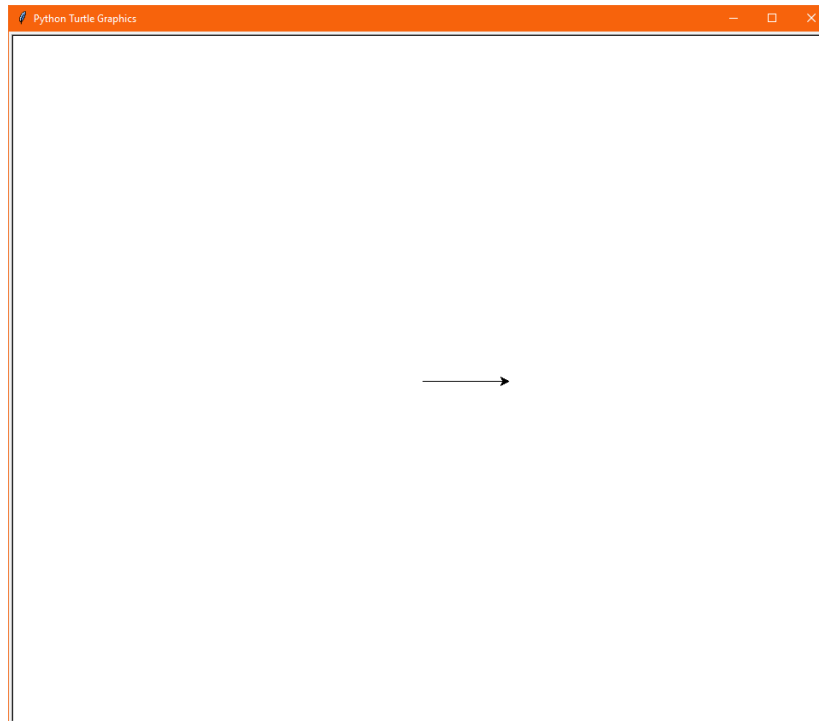
복잡한 수학 계산을 대신해주는 모듈!
제곱, 삼각함수 등...



거북이로 그림을 그려주는 모듈?

같이 해보기

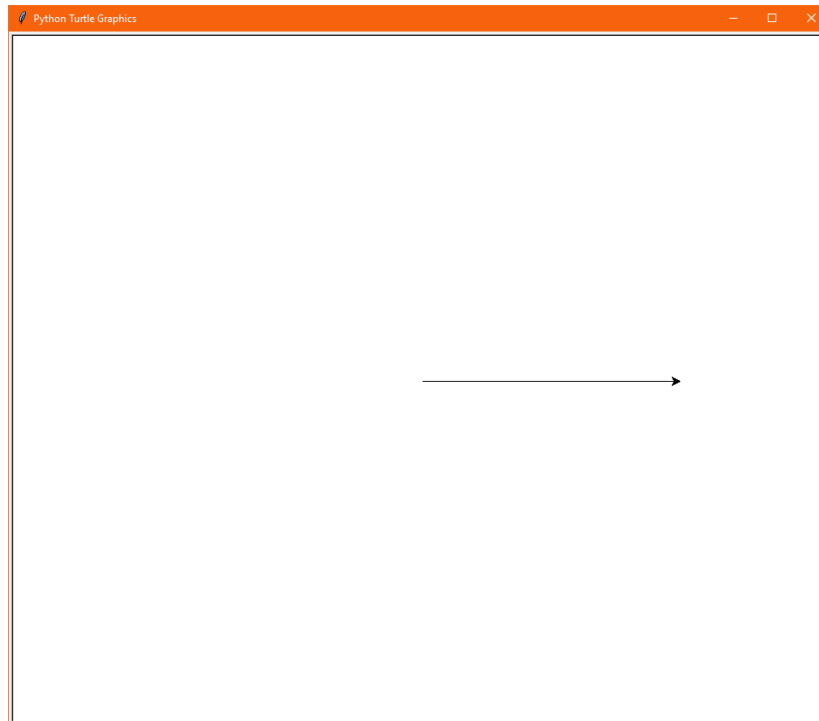
```
>>> import turtle as t  
>>> t.forward(100)
```



같이 해보기

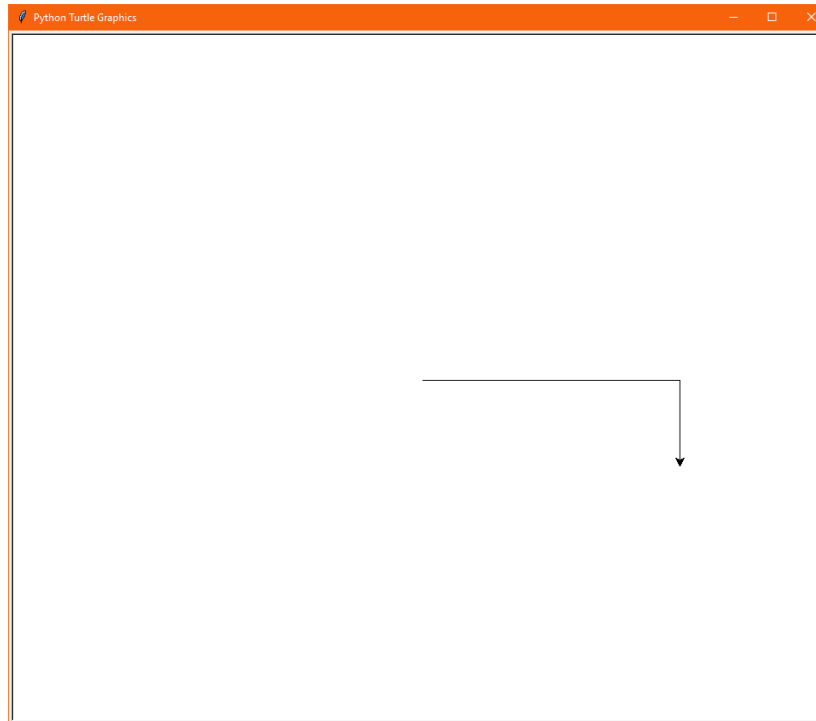
```
>>> t.forward(200)
```

200만큼 앞으로 간다.



```
>>> t.right(90)
>>> t.forward(100)
```

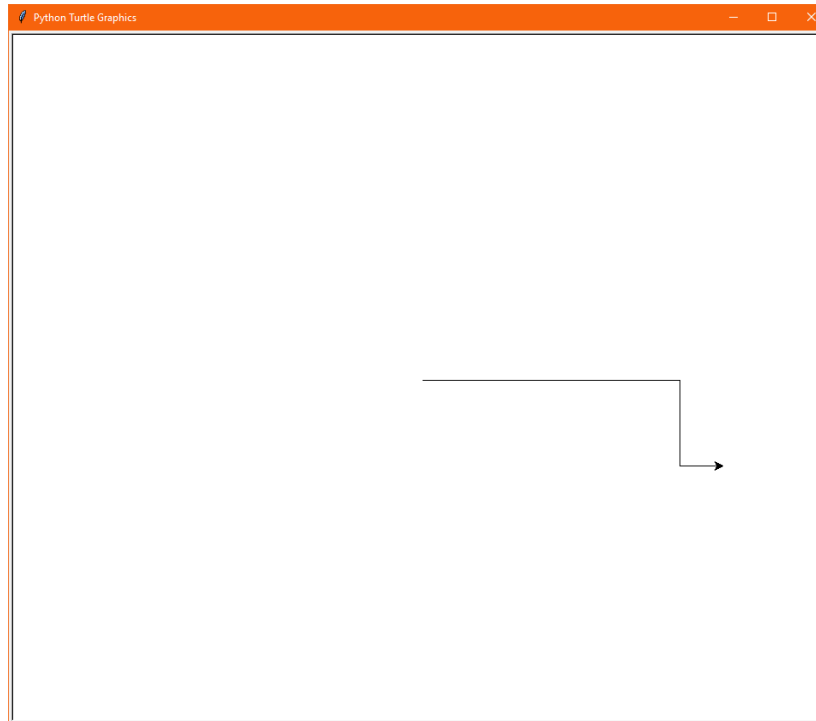
오른쪽으로 90도 회전한다.
앞으로 100만큼 간다.



```
>>> t.left(90)
```

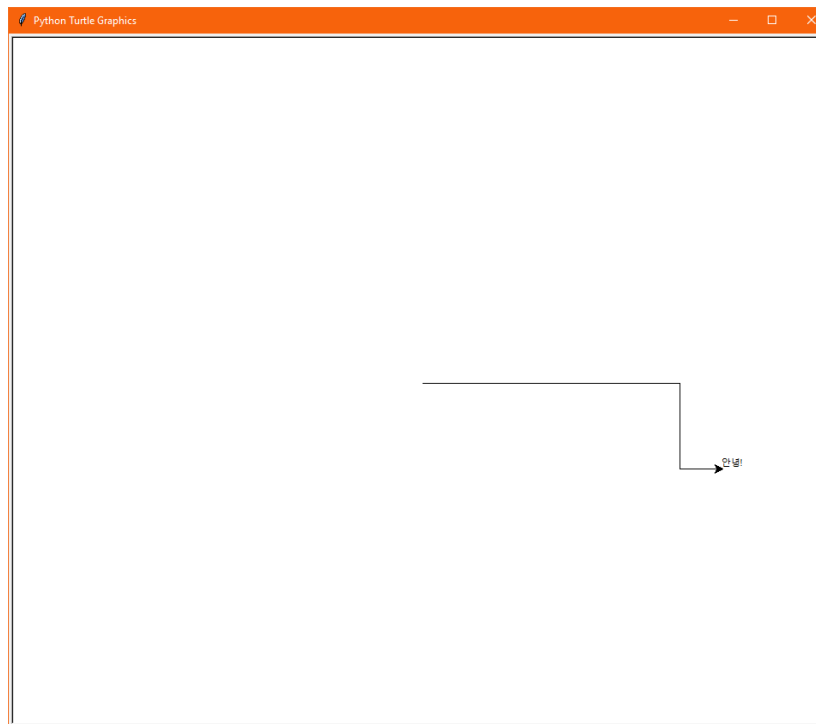
```
>>> t.forward(50)
```

왼쪽으로 90도 회전한다.
앞으로 50만큼 간다.



```
>>> t.write("안녕!")
```

현재 위치에 “안녕!” 이라고 쓴다.



거북이 움직이기

▲ t.forward(100)

100만큼 앞으로 가기

▼ t.backward(50)

50만큼 뒤로 가기

◀ t.left(90)

왼쪽으로 90도 돌기

▶ t.right(90)

오른쪽으로 90도 돌기

♡ t.write("안녕!")

"안녕!"이라고 쓰기

○ t.circle(50)

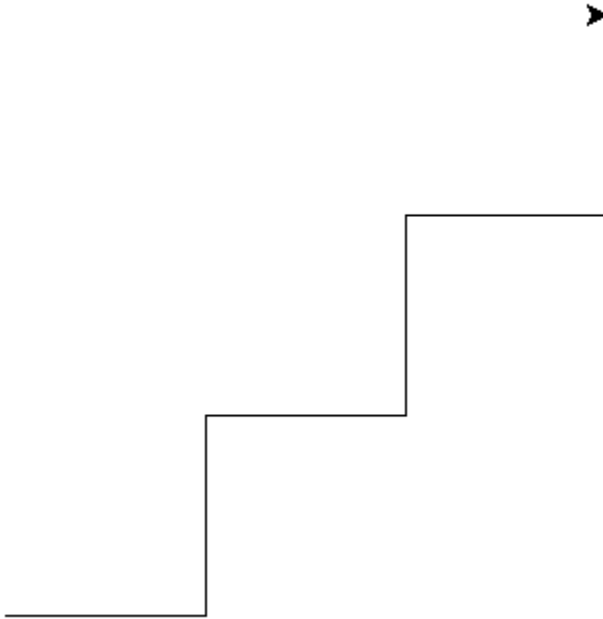
크기 50의 원 그리기

☆ t.reset()

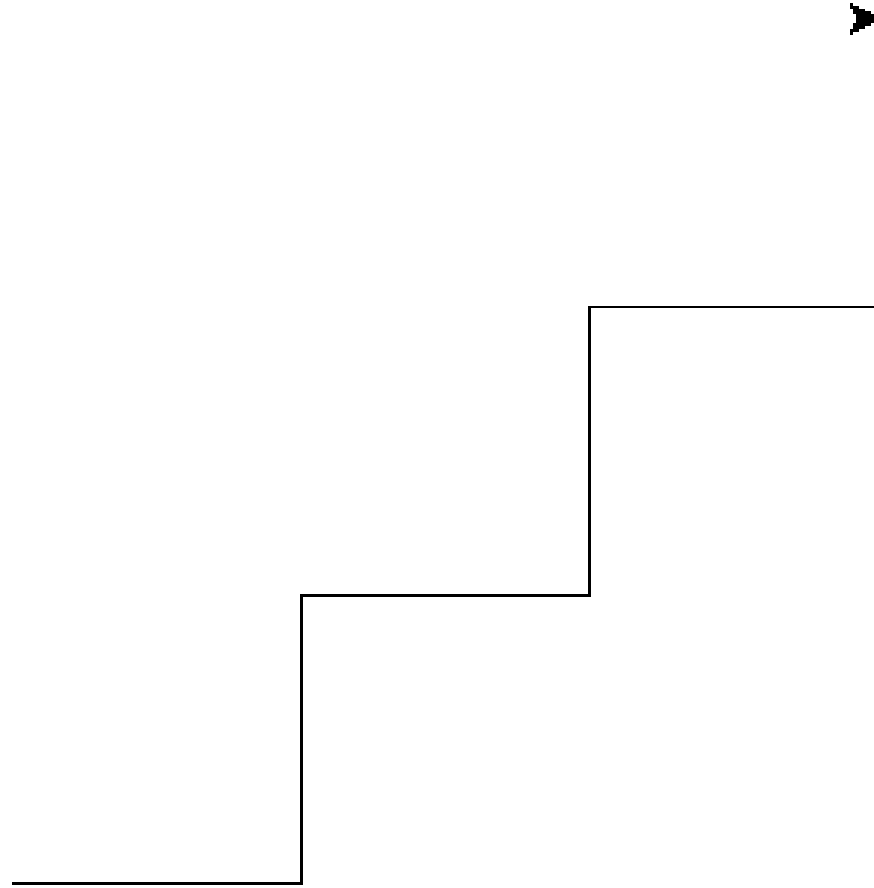
모두 지우고 처음으로 돌아간다.

같이 해보기

지그재그 모양을 그려보자.

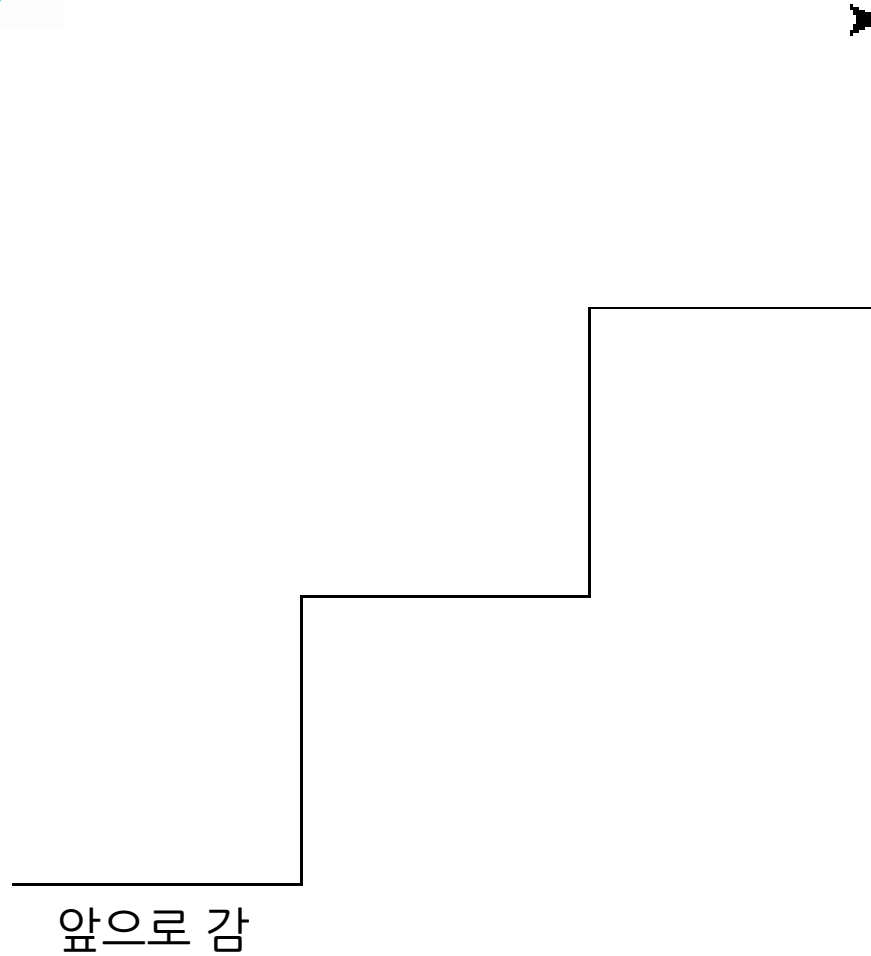


같이 해보기



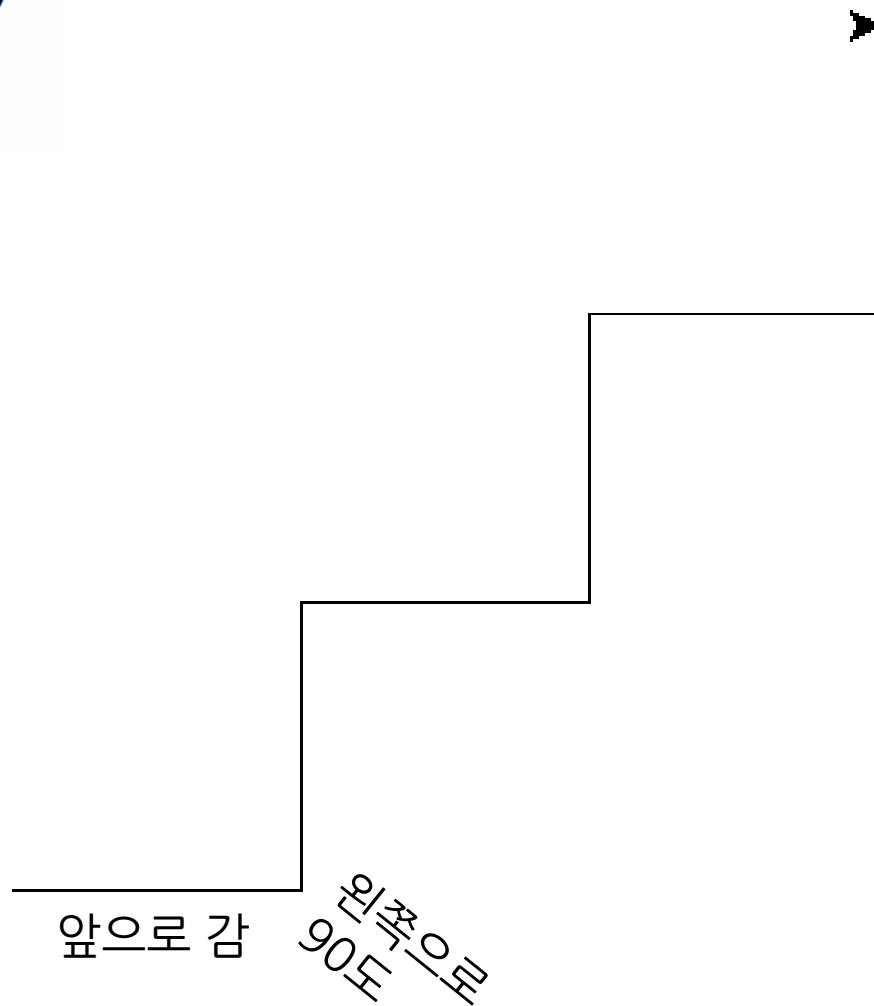
같이 해보기

```
t.forward(100)
```



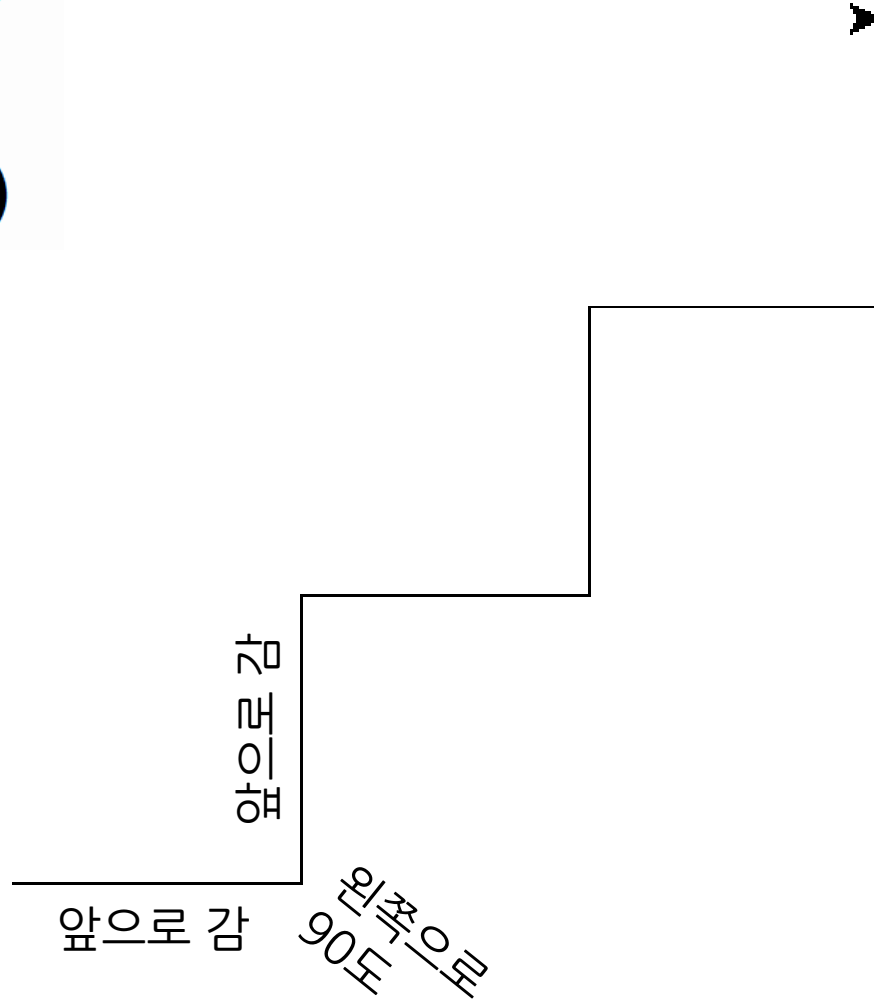
같이 해보기

```
t.forward(100)  
t.left(90)
```



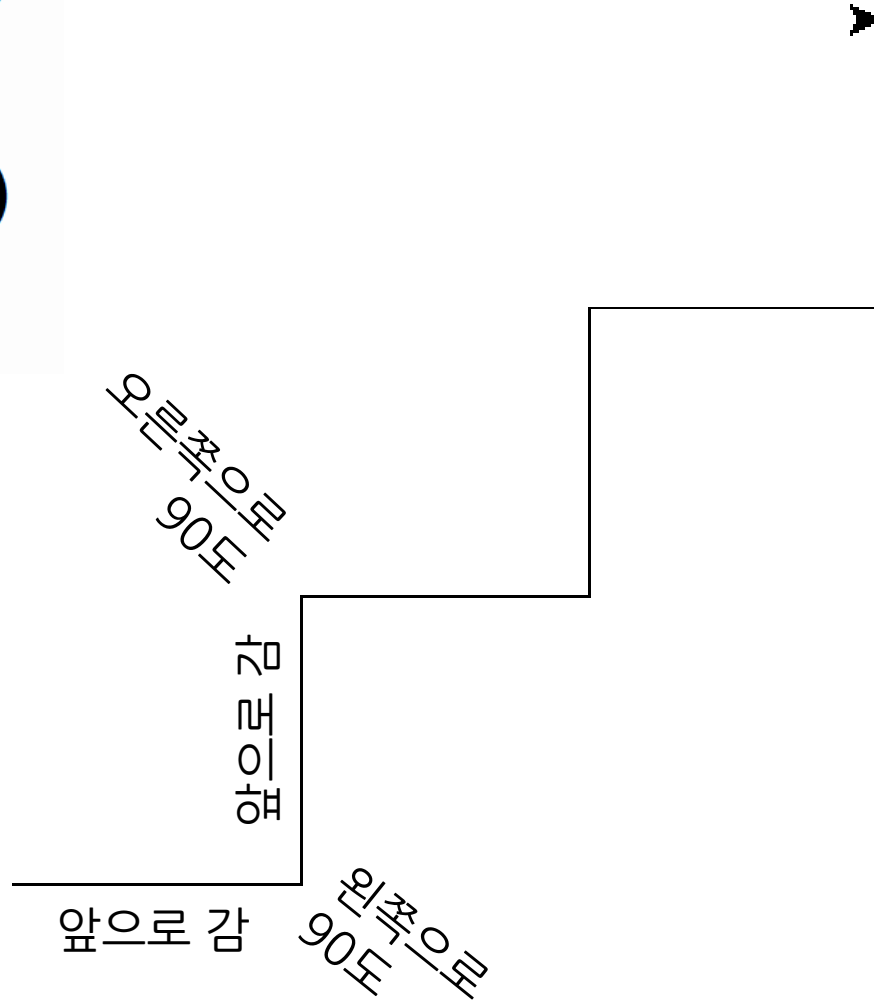
같이 해보기

```
t.forward(100)  
t.left(90)  
t.forward(100)
```



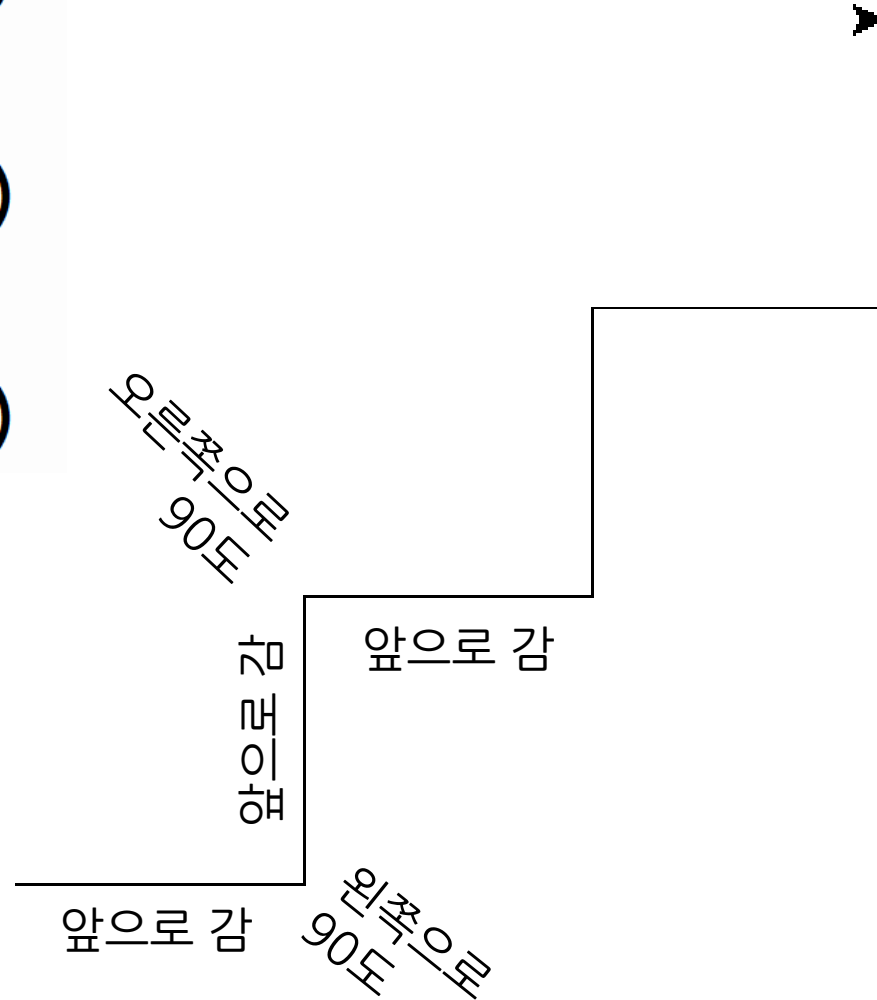
같이 해보기

```
t.forward(100)  
t.left(90)  
t.forward(100)  
t.right(90)
```



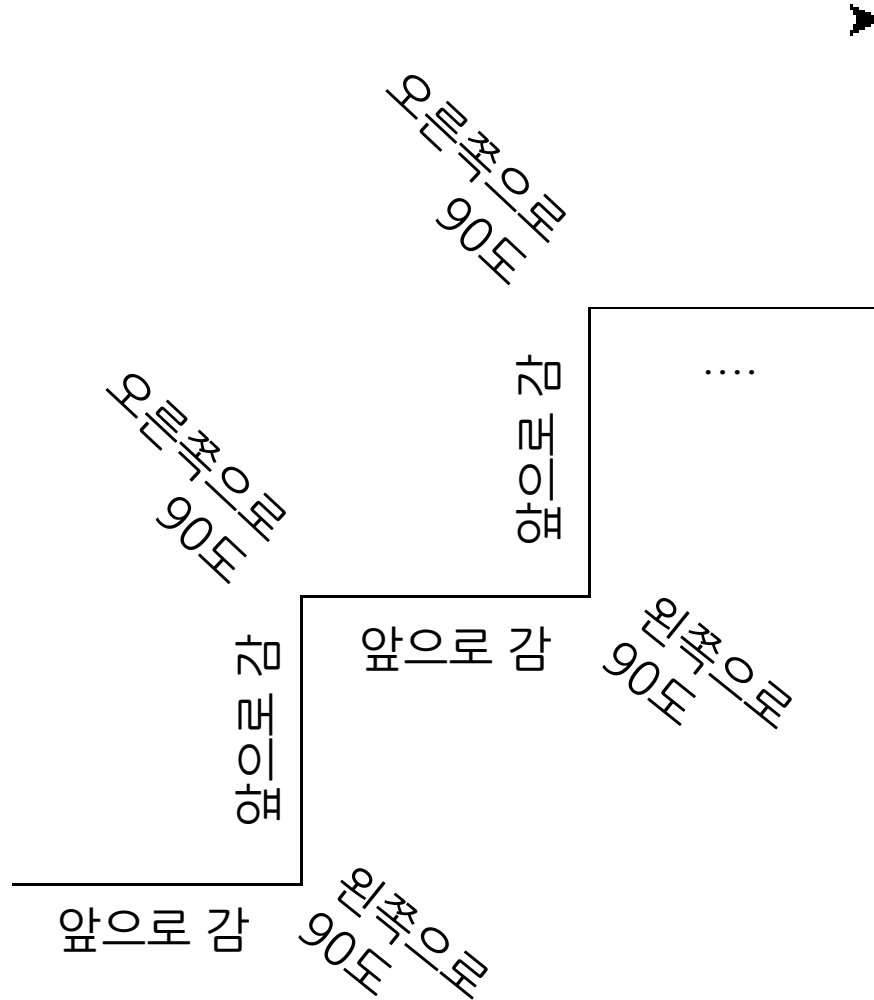
같이 해보기

```
t.forward(100)  
t.left(90)  
t.forward(100)  
t.right(90)  
t.forward(100)
```



같이 해보기

```
t.forward(100)
t.left(90)
t.forward(100)
t.right(90)
t.forward(100)
t.left(90)
t.forward(100)
t.right(90)
t.forward(100)
t.left(90)
....
```



직접 해보기

1. 거북이로 정사각형을 그려보자.

힌트: 아래 명령들 중 알맞는 것을 사용하면 된다.

`t.forward(200)`

200만큼 앞으로 가기

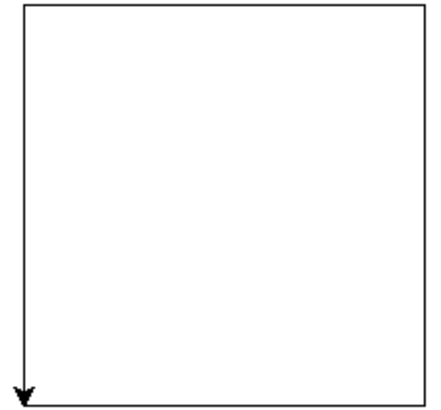
`t.left(90)`

왼쪽으로 90도 돌기

`t.right(90)`

오른쪽으로 90도 돌기

뭔가 잘못 했을 때에는 `t.reset()` 으로 처음부터!



직접 해보기

2. 거북이로 정삼각형을 그려보자.

힌트: 아래 명령들 중 알맞는 것을 사용하면 된다.

`t.forward(200)`

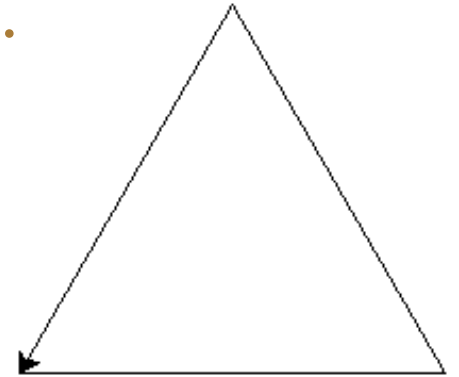
200만큼 앞으로 가기

`t.left(120)`

왼쪽으로 120도 돌기

`t.right(120)`

오른쪽으로 120도 돌기

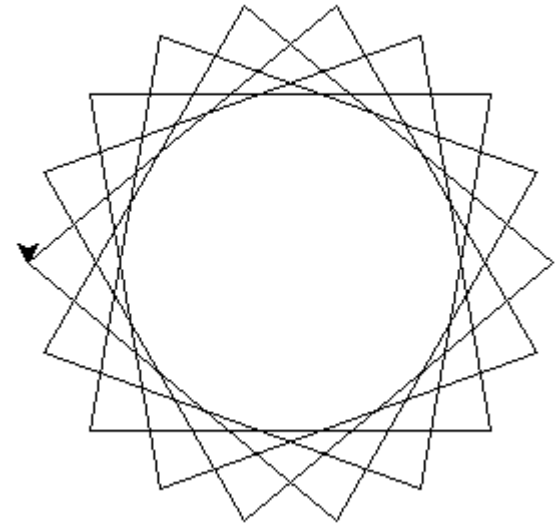


뭔가 잘못 했을 때에는 `t.reset()` 으로 처음부터!

직접 해보기

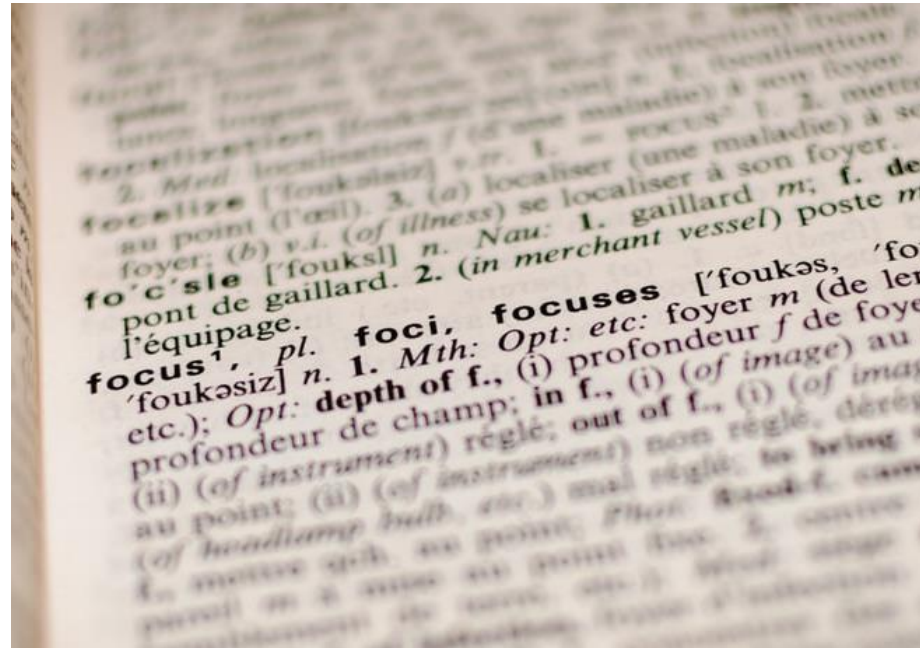
3. 아래 코드를 자유롭게 수정해 모양을 만들어보자.

```
import turtle as t
for a in range(0, 1000):
    t.forward(???)
    t.left(???)
```



10, 50, 100 같은 숫자나 a , $a*2$, $a+50$ 같은 식을 넣어보자.

딕셔너리란?



“사전”

단어와 뜻이 짝을 지어있다.

딕셔너리란?

apple



사과. 사과나무의 열매.

bear



곰. 갈색의 털북숭이 포유류.

car



자동차. 바퀴가 달린 이동수단.

key

열쇠

value

값

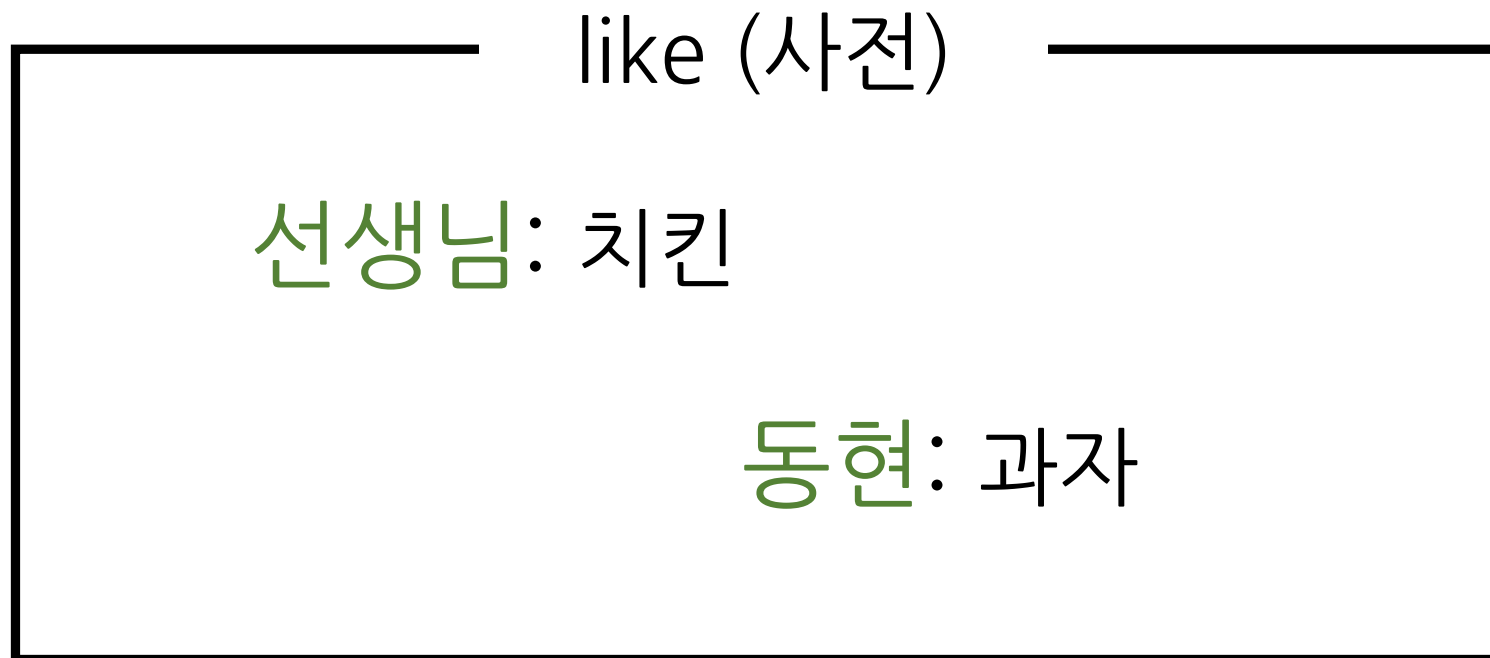
딕셔너리

딕셔너리, 즉 사전은 중괄호를 사용한다.

```
word = {"apple": "사과"}  
like = {"선생님": "치킨",  
        "동현": "과자"}  
empty = {}
```

딕셔너리

```
like = {"선생님": "치킨",  
        "동현": "과자"}
```



hello[A]

hello[A]

hello 사전의 A 에 해당하는 값을 뱉는다.

```
>>> like = {"선생님": "치킨",  
            "동현": "과자"}
```

```
>>> like["동현"]  
'과자'
```

```
>>> like["선생님"]  
'치킨'
```

딕셔너리

```
like = {"선생님": "치킨",  
        "동현": "과자"}
```



hello[A] = B

hello[A] = B

hello 사전의 A 의 값을 B 로 한다.

```
>>> a = {}  
>>> a["치킨"] = "맛있음"  
>>> a["샐러드"] = "상쾌함"  
>>> a  
{ '치킨': '맛있음', '샐러드': '상쾌함' }
```


영어사전 만들기

```
eng = {}  
eng["apple"] = "사과"  
eng["bike"] = "자전거"  
eng["carrot"] = "당근"  
eng["diet"] = "다이어트"  
eng["elephant"] = "코끼리"
```

영어사전 만들기

```
eng["elephant"] = "코끼리"
```

```
a = input("영어 단어를 입력하세요.")  
print(eng[a])
```



```
>>> %Run a.py  
영어 단어를 입력하세요. carrot  
당근
```

음식점 메뉴 만들기

```
menu = {}  
menu["치킨"] = 18000  
menu["라면"] = 3000  
menu["공깃밥"] = 1000  
  
a = input("무슨 메뉴를 시키시겠습니까?")  
print(a, "의 가격은", menu[a], "입니다.")
```

음식점 메뉴 만들기

```
menu["공깃밥"] = 1000
```

```
a = input("무슨 메뉴를 시키시겠습니까?")  
print(a, "의 가격은", menu[a], "입니다.")
```



```
>>> %Run a.py
```

```
무슨 메뉴를 시키시겠습니까?공깃밥  
공깃밥 의 가격은 1000 입니다.
```

직접 해보기

1. 자신이 아는 영어 단어를 이용해 한글→영어 사전을 만들어보자. 단 아래와 같이 단어를 알려줘야 한다. (중요: 코드를 저장해두자!)

힌트: 딕셔너리(사전)과 input 함수를 잘 활용해보자.

```
>>> %Run a.py
```

```
한글 단어를 입력해주세요. 사과  
영어로 apple 입니다.
```

2. 음식을 열쇠로, 가격을 값으로 가지고 있는 “메뉴” 사전을 만들고 음식을 여러 개 주문하면 총 가격이 얼마인지 알려주는 프로그램을 만들자.

힌트: 딕셔너리(사전), input 함수, int 변환을 잘 활용해보자.

```
>>> %Run a.py
```

```
무얼 주문하시겠습니까?치킨  
몇인분 주문하시겠습니까?2  
가격은 36000 원입니다.
```

3. 2번 문제에서 만들었던 프로그램의 앞부분에 메뉴 소개를 넣어보자. 단 for문을 사용해서 “메뉴” 사전 안에 있는 것들을 반복해 출력하자.

```
>>> %Run a.py
```

```
김밥지옥 가격표
```

```
치킨 18000
```

```
공깃밥 1000
```

```
라면 3000
```

```
무얼 주문하시겠습니까? 라면
```

```
몇인분 주문하시겠습니까? 3
```

```
가격은 9000 원입니다.
```

복습

A in B

B 안에 A가 있으면 True, 없으면 False를 뱉는다.

```
>>> "감자" in "감자튀김"
```

```
True
```

```
>>> "d" in ["a", "b", "c"]
```

```
False
```



```
a = input("오늘 저녁엔 뭘 먹을까? ")
if "고기" in a:
    print("오늘 저녁은 푸짐하군!")
else:
    print("아쉽다... 내일 저녁은 고기죠?")
```



```
>>> %Run ye.py
오늘 저녁엔 뭘 먹을까? 소고기김치찌개
오늘 저녁은 푸짐하군!
```

```
>>> %Run ye.py
오늘 저녁엔 뭘 먹을까? 두부조림
아쉽다... 내일 저녁은 고기죠?
```

in 연산자를 딕셔너리(사전)에서!

```
>>> 메뉴 = {"치킨": 18000,  
            "공깃밥": 1000,  
            "라면": 3000}
```

```
>>> "치킨" in 메뉴
```

```
True
```

```
>>> "샐스핀" in 메뉴
```

```
False
```

$A \text{ in } B$

A라는 열쇠가 B에 있는가?

in 연산자를 딕셔너리(사전)에서!

```
if a in 메뉴:  
    print(메뉴[a], "원입니다.")  
else:  
    print("죄송하지만 그런 음식은 안팝니다.")
```



```
>>> %Run a.py
```

```
뭘 시키시겠습니까? 지옥불갈창볶음  
죄송하지만 그런 음식은 안팝니다.
```

for 문을 딕셔너리(사전)에서!

```
>>> 메뉴 = {"치킨": 18000,  
            "공깃밥": 1000,  
            "라면": 3000}
```

```
>>> for a in 메뉴:  
    print(a)
```

```
치킨  
공깃밥  
라면
```

for a in 메뉴
“메뉴” 사전 속 모든 열쇠를
a 에 한번 씩 담아 반복

같이 해보기

1. 로그인 프로그램을 만들어보자. 단 ID와 패스워드는 다음과 같은 사전으로 저장한다.

```
users = {"선생님": "teacher01",  
         "현빈": "hb0808",  
         "동현": "ohdonghyeon",  
         "지완": "gwanman@"}
```

```
users = {"선생님": "teacher01",  
         "현빈": "hb0808",  
         "동현": "ohdonghyeon",  
         "지완": "gwanman@"}
```

```
아이디 = input("아이디를 입력해주세요.")  
비번 = input("비밀번호를 입력해주세요.")
```

```
users = {"선생님": "teacher01",  
         "현빈": "hb0808",  
         "동현": "ohdonghyeon",  
         "지완": "gwanman@"}
```

```
아이디 = input("아이디를 입력해주세요.")  
비번 = input("비밀번호를 입력해주세요.")  
if
```

```
users = {"선생님": "teacher01",  
         "현빈": "hb0808",  
         "동현": "ohdonghyeon",  
         "지완": "gwanman@"}
```

```
아이디 = input("아이디를 입력해주세요.")  
비번 = input("비밀번호를 입력해주세요.")  
if users[아이디] == 비번:
```



```
users = {"선생님": "teacher01",  
         "현빈": "hb0808",  
         "동현": "ohdonghyeon",  
         "지완": "gwanman@"}
```

```
아이디 = input("아이디를 입력해주세요.")  
비번 = input("비밀번호를 입력해주세요.")  
if users[아이디] == 비번:  
    print("로그인 성공.")
```

```
users = {"선생님": "teacher01",  
         "현빈": "hb0808",  
         "동현": "ohdonghyeon",  
         "지완": "gwanman@"}
```

```
아이디 = input("아이디를 입력해주세요.")  
비번 = input("비밀번호를 입력해주세요.")  
if users[아이디] == 비번:  
    print("로그인 성공.")  
else:  
    print("로그인 실패.")
```

```
>>> %Run a.py
```

아이디를 입력해주세요. 선생님
비밀번호를 입력해주세요. teacher01
로그인 성공.

```
>>> %Run a.py
```

아이디를 입력해주세요. 선생님
비밀번호를 입력해주세요. 에뷔뵐뵐뵐
로그인 실패.

```
>>> %Run a.py
```

아이디를 입력해주세요.비둘기

비밀번호를 입력해주세요.비둘비둘

Traceback (most recent call last):

File "C:\Users\lumer\Desktop\a.py", line 8, in <module>
≥

```
    if users[아이디] == 비번:
```

KeyError: '비둘기'



“열쇠 오류”

어디서 오류가 나는걸까?

```
users = {"선생님": "teacher01",  
         "현빈": "hb0808",  
         "동현": "ohdonghyeon",  
         "지완": "gwanman@"}
```

```
아이디 = input("아이디를 입력해주세요.")  
비번 = input("비밀번호를 입력해주세요.")  
if users[아이디] == 비번:  
    print("로그인 성공.")  
else:  
    print("로그인 실패.")
```

```
users = {"선생님": "teacher01",  
         "현빈": "hb0808",  
         "동현": "ohdonghyeon",  
         "지완": "gwanman@"}
```

```
아이디 = input("아이디를 입력해주세요.")  
비번 = input("비밀번호를 입력해주세요.")  
if users[아이디] == 비번:  
    print("로그인 성공.")  
else:  
    print("로그인 실패.")
```

users에 “비둘기”라는
열쇠가 없다.

즉 열쇠 오류가 발생한다.

```
s = {"선생님": "teacher01",  
     "현빈": "hb0808",  
     "동현": "ohdonghyeon",  
     "지완": "gwanman@"}
```

```
id = input("아이디를 입력해주세요.")  
pw = input("비밀번호를 입력해주세요.")  
if s[id] == pw:  
    print("로그인 성공.")  
else:  
    print("로그인 실패.")
```

열쇠가 있는지
확인부터 해야된다!

```
아이디 = input("아이디를 입력해주세요. ")
비번 = input("비밀번호를 입력해주세요.")
if 아이디 not in users:
    print("아이디가 존재하지 않습니다.")
elif users[아이디] == 비번:
    print("로그인 성공.")
else:
    print("로그인 실패 ")
```


직접 해보기

1. 아까 만들었던 한글→영어 사전 프로그램을 수정하자. 사전에 없는 단어를 입력해도 오류가 나지 않도록 if문을 적절히 추가해보자.

힌트: if문과 not in 문을 이용해보자.

```
>>> %Run a.py
```

한글 단어를 입력해주세요. 에뮬뵈레뵈
알 수 없는 단어입니다.

직접 해보기

2. 다음을 참고하여 사전을 만들고 for문으로 아래처럼 출력해보자.

열쇠	값
양념치킨	맛있음
신라면	매움
지우개	큰일남

```
>>> %Run a.py
```

```
양념치킨 은 먹으면 맛있음  
신라면 은 먹으면 매움  
지우개 은 먹으면 큰일남
```



파이썬과 친해지기

turtle 모듈, 딕셔너리(사전), 딕셔너리 심화

8주차

References

왕초보를 위한 Python 2.7
뱀 인형 이미지
Stock Photos

<https://wikidocs.net/145>
<https://bit.ly/2WINL65>
<https://unsplash.com/>