

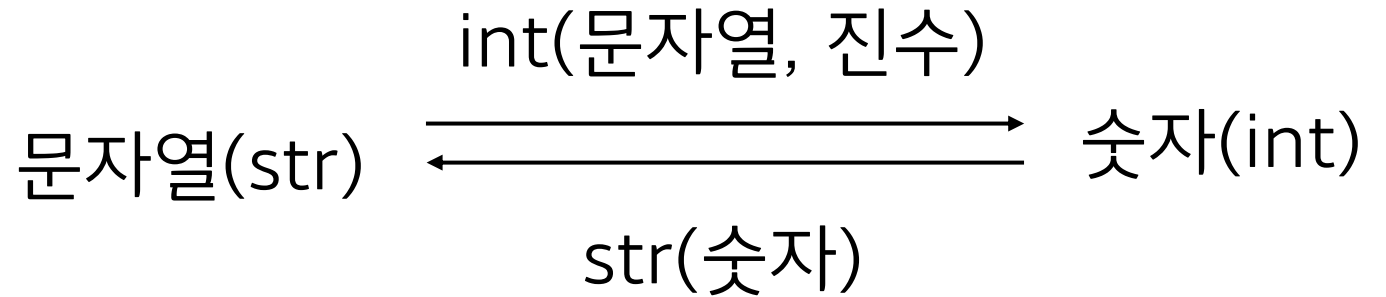
1장 알고리즘 기초

강주란

1. input()함수와 형 변환(broad casting)

input() 함수는
문자열(str)을 반환함

```
1 a = input()  
10  
  
1 print(a)  
10  
  
1 type(a)  
str
```



2. 복합문의 구조

헤더

if식 : 스위트
else식 : 스위트
else식 : 스위트

스위트

- 행마다 같은 수준의 들여쓰기를 해야함
- 스위트가 단순문인 경우, 헤더와 같은 행에 쓸 수 있음
- 스위트가 복합문인 경우, 헤더와 같은 행에 쓸 수 없음

3. 연산자

조건 연산자인 if~else 문은 파이썬의 유일한 삼항 연산자

- 피연산자가 3개
ex. a if b else c

```
a = x if x>y else y  
  
print('c는 0입니다.' if c == 0 else 'c는 0이 아닙니다.')
```

4-1. while문 반복 - 구조

어떤 조건이 성립하는 동안 반복해서 처리하는 것

While 문은 판단 반복 구조

실행하기 전에 반복을 계속할 것인지를 판단

While 조건식 : 명령문

루프 본문

4-2. while문 반복 - 무한 루프와 break 문

무한 루프

의도적으로 while 문이 무한 반복되도록 설계

```
while True:
    n = int(input())
    if n > 0:
        break
```

break 문을 사용해 무한루프를 빠져나옴

* for문에서도 break문을 사용 가능

4-3. while문 반복 - continue 문

```
area = int(input())

for i in range(1, area+1):
    if i*i > area: break
    if area % i : continue
    print(f'{i}' x {area//i})
```

break 문을 사용해 루프를 빠져나옴

continue 문을 사용해
루프 본문의 나머지 부분을 건너뛰고
print(f'{i} x {area // i}')
조건식인 for문으로 돌아감

5-1. for문 반복 - 구조

리스트나 튜플, 문자열의 첫 번째 요소부터 마지막 요소까지
차례로 변수에 대입되어
"수행할 문장1", "수행할 문장2" 등이 수행된다

```
for 변수 in 리스트(또는 튜플, 문자열):  
    수행할 문장1  
    수행할 문장2  
    ...
```


5-2. for문 반복 - break 문

```
for i in range(10000):  
    print(i)  
    if i == 100:  
        break
```

break 문을 사용해 루프를 빠져나옴

5-3. for문 반복 - continue 문

```
for i in range(1, 13):  
    if i == 8:  
        continue  
    print(i, end = ' ')  
  
print()
```

continue 문을 사용해
루프 본문의 나머지 부분을 건너뛰고
조건식인 for문으로 돌아감

5-4. for문 반복 - 다중루프

```
n = int(input())
```

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

```
    for j in range(i+1):  
        print('*', end='')
```

```
    print()
```

행 루프

열 루프

6. 파이썬의 변수

전역 변수

프로그램 전체에서 사용 가능

지역 변수

함수 내부에서 정의한 변수로,
함수 내부에서만 사용 가능

```
n = 1
```

```
def put_id():
```

```
    x = 1
```

```
    print(id(x))
```