

3장. 제어문(조건, 반복)



목 차

1

if문

2

while문

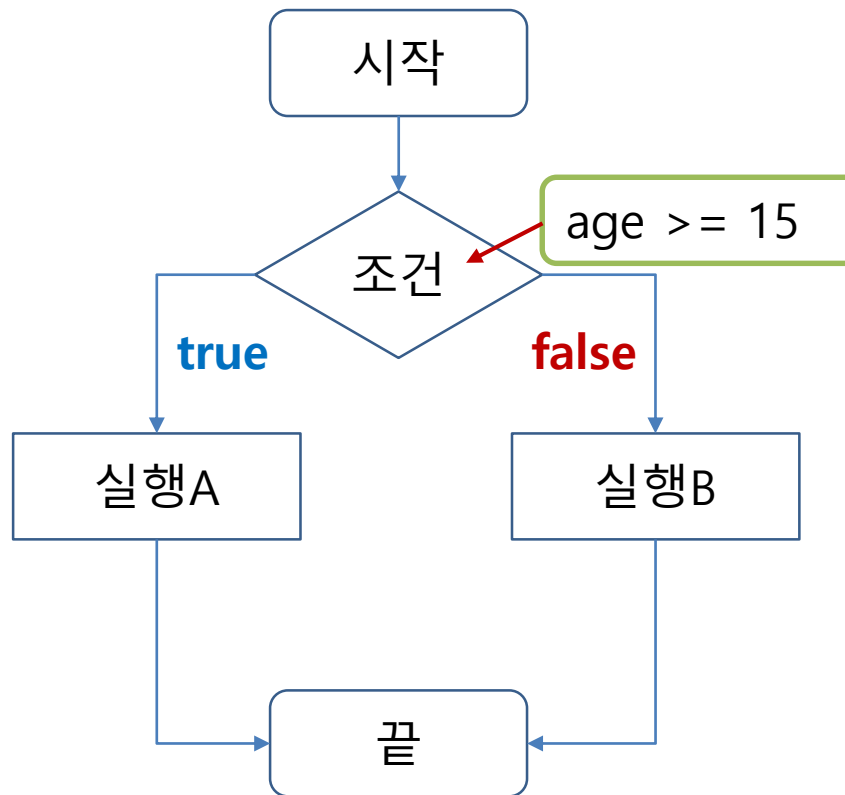
3

for문

조건문

조건문

- 특정한 조건에 의해서 프로그램 진행이 분기되는 문장



조건문 - if

- if문

if 논리식:

실행할 문장

True이면 실행

```
# if문
age = 6
if age < 8:
    print('학교에 가지 않습니다.')
print("나이는", age, "세 입니다.")
```

조건문 – if ~ else

- if ~ else 구문

True/False

if 논리식:

실행문 1

else :

실행문 2

```
# if ~ else 구문
speed = int(input("속도 입력: "))

if speed > 50:
    print("속도 위반, 과태료 10만원")
else:
    print("안전 속도 준수!!")
```

중첩조건 – if ~elif ~ else

▪ if ~ elif ~ else구문

if 논리식:

실행문 1

elif 논리식:

실행문 2

elif 논리식:

실행문 3

else :

실행문 4

♣ 놀이 공원 입장료 계산 ♣
초.중등학생입니다.
입장료는 2,500원입니다.

대 상	입장료
취학전 아동	1,000원
초등학생	2,000원
중.고등학생	2,500원
일반인	3,000원

중첩조건 – if ~elif ~ else

```
print("♣ 놀이 공원 입장료 계산 ♣")

age = 15
charge = 0;
if age < 8:
    print('미취학 아동입니다.')
    charge = 1000
elif age >= 8 and age < 14:
    print('초.중등학생입니다.')
    charge = 2000
elif age >= 14 and age < 20:
    print('초.중등학생입니다.')
    charge = 2500
else:
    print('초.중등학생입니다.')
    charge = 3000

#print("입장료는 " + str(charge) + "원입니다.")
print("입장료는 " + format(charge, ',d') + "원입니다.")
```

조건이 둘 이상인 논리연산

▪ if ~ and(or) ~ else구문

```
if 논리식1 and 논리식2:
```

```
    실행문 1
```

```
else :
```

```
    실행문 2
```

```
money = 3000
card = False
if money >= 4000 or card:
    print("택시를 탄다")
elif money >= 2000 or not card:
    print("버스를 탄다")
else:
    print("걸어간다")

print('='*30)

pocket = ['paper', '스마트폰', 'money']
if 'coin' in pocket:
    print('택시를 탄다')
else:
    print('지하철을 탄다')
```


if ~ 내부 if

■ if ~ 내부 if

```
if 논리식1:
    if 논리식:
        실행문1
    else:
        실행문2
else:
    if 논리식:
        실행문3
    else:
        실행문4
```

★ 수를 입력 받아 10보다 큰 수 중 짝수, 홀수를 구분하고, 10 이하의 수 중 짝수, 홀수 구분하는 프로그램

```
수를 입력하세요 :
11
10보다 큰 홀수입니다.
>>>
=====
수를 입력하세요 :
6
10이하의 짝수입니다.
```

if ~ 내부 if

```
print("수를 입력하세요 : ")
n = int(input())

if n > 10:
    #print("10보다 큰 수입니다.")
    if n % 2 == 0:
        print("10보다 큰 짝수입니다.")
    else:
        print("10보다 큰 홀수입니다.")
else:
    #print("10보다 작은 수입니다.")
    if n % 2 == 0:
        print("10이하의 짝수입니다.")
    else:
        print("10이하의 홀수입니다.")
```

```
if n > 10 and n % 2 == 0:
    print("10보다 큰 짝수입니다.")
elif n > 10 and n % 2 == 1:
    print("10보다 큰 홀수입니다.")
elif n < 10 and n % 2 == 0:
    print("10보다 작은 짝수입니다.")
else:
    print("10보다 작은 홀수입니다.")
```

조건문 연습 문제

윤년을 판별하는 프로그램

년도 입력: 2020
2020년은 윤년입니다.

=====
년도 입력: 1900
1900년은 윤년이 아닙니다.

$((year \% 4 == 0) \text{ and } (year \% 100 != 0)) \text{ or } (year \% 400 == 0)$

연도가 4로 나누어떨어진다.

100으로 나누어떨어지는
연도는 제외한다.

400으로 나누어떨어지는
연도는 윤년이다.

```
year = int(input("년도 입력: "))  
  
if ((year % 4 == 0) and (year % 100 != 0)) or (year % 400 == 0):  
    print(str(year) + "년은 윤년입니다.")  
else:  
    print(str(year) + "년은 윤년이 아닙니다.")
```

조건문 연습 문제

자리 배치도 프로그램

입장객수에 따른 좌석 줄 수의 개수 계산하는 프로그램을 작성하세요.

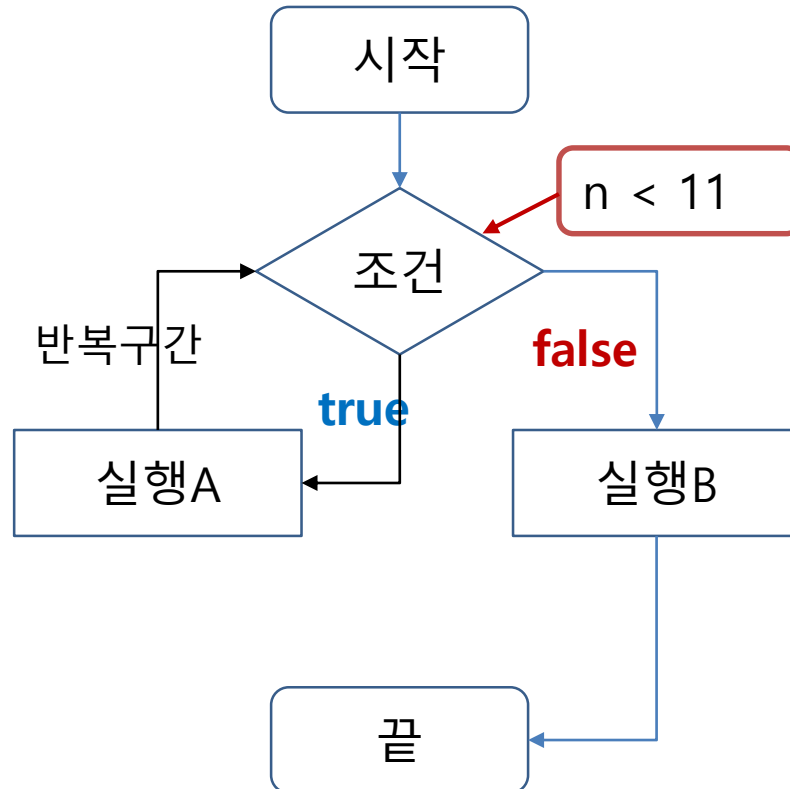
```
입장객 수 : 44  
열 수 : 5  
9개의 줄이 필요합니다.
```

```
customer_num = int(input("입장객 수 입력 : "))  
col_num = int(input("좌석 열의 수 입력: "))  
  
if customer_num % col_num == 0:  
    row_num = customer_num / col_num  
else:  
    row_num = customer_num // col_num + 1  
    #row_num = int(customer_num / col_num) + 1  
  
print("%d개의 줄이 필요합니다." % row_num)
```

반복문

반복문

- 주어진 조건이 만족할 때까지 실행문을 반복적으로 수행
- while문과 for문을 사용함



반복문

while문

초기값

while 조건문:

실행문1

증감값

```
i = 0
while i < 10:
    i += 1
    print("Hello~ ")
```

반복문 - while

1부터 10까지 더하는 프로그램

```
sum = 0
sum = sum + 1;
sum = sum + 2;
sum = sum + 3;
sum = sum + 4;
sum = sum + 5;

print(sum)
```

```
i = 0
sum = 0
while i < 10:
    i += 1
    sum += i
    print("i =", i, ", sum = ", sum)

print(sum)
```

```
i = 1 , sum = 1
i = 2 , sum = 3
i = 3 , sum = 6
i = 4 , sum = 10
i = 5 , sum = 15
i = 6 , sum = 21
i = 7 , sum = 28
i = 8 , sum = 36
i = 9 , sum = 45
i = 10 , sum = 55
55
```

반복문 - while

- 무한 반복 - break문

```
while True:  
    수행문  
    if 조건 :  
        break
```

```
i = 0  
while True:  
    i += 1  
    if i > 10:  
        break  
    print(i)
```

```
1  
2  
3  
4  
5  
6  
7  
8  
9  
10
```


반복문 - while

- 무한 반복 - break문

```
반복을 계속 할까요?(y/n)
y
반복을 계속 합니다.
반복을 계속 할까요?(y/n)
y
반복을 계속 합니다.
반복을 계속 할까요?(y/n)
k
정상 답변이 아닙니다.
반복을 계속 할까요?(y/n)
n
```

```
while True:
    answer = input('반복을 계속 할까요?(y/n)')

    if answer == 'y' or answer == 'Y':
        print('반복을 계속 합니다.')
    elif answer == 'n' or answer == 'N':
        print('반복을 중단 합니다.')
        break
    else:
        print('정상 답변이 아닙니다.')
```

반복 조건문 – while ~ if

커피 자동판매기 프로그램

- 돈을 입력하면 커피가 나온다. (커피 가격 : 400원)
- 돈이 커피 가격을 초과하면 거스름돈을 돌려준다.
- 돈이 커피 가격보다 작으면 돈은 돌려주고 커피가 나오지 않는다.
- 커피의 수량이 소진되면 판매가 중지된다. (커피 수량 : 5개)

```
돈을 넣어주세요: 400
커피가 나옵니다
남은 커피의 양은 4개 입니다.
돈을 넣어주세요: 400
커피가 나옵니다
남은 커피의 양은 3개 입니다.
돈을 넣어주세요: 500
커피가 나오고, 거스름돈 100원을 돌려 받습니다.
남은 커피의 양은 2개 입니다.
돈을 넣어주세요: 300
커피가 나오지 않습니다.
남은 커피의 양은 2개 입니다.
돈을 넣어주세요: 400
커피가 나옵니다
남은 커피의 양은 1개 입니다.
돈을 넣어주세요: 400
커피가 나옵니다
남은 커피의 양은 0개 입니다.
커피가 모두 소진되었습니다. 판매를 중지합니다.
```

반복 조건문 – while ~ if

커피 자동판매기 프로그램

```
coffee = 5      #전역 변수
price = 400
while True:
    money = int(input("돈을 넣어주세요: "))
    if money == price:
        print("커피가 나옵니다")
        coffee = coffee - 1
    elif money > price:
        print("커피가 나오고, 거스름돈 " + str(money - price) + "원을 돌려 받습니다.")
        coffee -= 1
    else:
        print("커피가 나오지 않습니다.")
    print("남은 커피의 양은 " + str(coffee) + "개 입니다.")

    if coffee == 0:
        print("커피가 모두 소진되었습니다. 판매를 중지합니다.")
        break
```

반복문 - for

for문

순서열의 각 원소를 처음부터 순회하면서 반복변수에 담아 낸다. 순서열은 **리스트, 튜플, 문자열** 등을 사용

for 반복변수 **in** 순서열:
코드블록

```
>>> for i in (1, 2, 3, 4, 5):  
    print(i)
```

```
1  
2  
3  
4  
5
```

```
>>> for i in range(1, 6, 1):  
    print(i)
```

```
1  
2  
3  
4  
5
```

반복문 - for

- range() 함수 사용하기

for x in range(**시작값, 종료값, 증감값**)

시작값을 생략하면 0부터 시작하고, 종료값은 (종료값-1) 이다.

```
# 1 ~ 10 출력
for i in range(1, 11, 1):
    print(i)
print()

# 1 ~ 10 홀수
for i in range(1, 11, 2):
    print(i, end = ' ')
print()

for i in range(1, 11):
    if i % 2 != 0:
        print(i, end = ' ')
print()
```

```
# 1 ~ 10 합계
sum = 0
for i in range(1, 11):
    sum += i

print("합계 :", sum)
```

구구단

단을 입력받아 구구단 출력하기

단을 입력하세요 : 4

```
4 x 1 = 4
4 x 2 = 8
4 x 3 = 12
4 x 4 = 16
4 x 5 = 20
4 x 6 = 24
4 x 7 = 28
4 x 8 = 32
4 x 9 = 36
```

```
x = int(input("단을 입력하세요 : "))
dan = int(x)

for i in range(1, 10):
    #print(dan, 'x', i, '=', dan*i)
    print("%d x %d = %d" % (dan, i, dan*i))
```

Continue문

- continue 문

for ~ in :
 if 조건 :
 continue
수행문

반복하다 조건에 맞으면
수행문을 처리하지 않고
다시 반복한다.

```
for i in range(1, 11, 1):  
    if i==5 or i==7:  
        continue  
    print(i)
```

```
# 1부터 10까지의 짝수 합계  
sum = 0  
for i in range(1, 11):  
    if i % 2 == 1:  
        continue  
    print(i, end = " ")  
    sum = sum + i  
  
print("합계 : ", sum)
```

반복문

이중 for문

- 행, 열 구현하기

```
for i in range(1, 6)
```

```
    for j in range(1, 6)
```

```
        실행문;
```

```
    print()
```

행

열

5행 5열

	열1	열2	열3	열4	열5
행1					
행2					
행3					
행4					
행5					

반독점

이중 for문

- 행, 열 구현하기

가
가
가
가
가

```
for i in range(5):
    for j in range(5):
        print('가', end='')
    print()
```

2중 for문 - 별 찍기

▣ 별로 삼각형 모양만들기

```
$  
$$  
$$$  
$$$$  
$$$$$  
  
=====:  
$$$$$  
$$$$$  
$$$  
$$  
$
```

```
for i in range(0, 5):  
    for j in range(0, i+1):  
        print("$", end='')  
    print()  
print("="*20)  
  
for i in range(0, 5):  
    for j in range(0, 5-i):  
        print("$", end='')  
    print()  
print("="*20)
```

2중 for문 - 별 찍기

▣ 별로 삼각형 모양만들기

```
1 2 3 4 5
6 7 8 9 10
1112131415
1617181920
2122232425
```

```
for i in range(0, 5):
    for j in range(1, 6):
        n = i*5+j
        print(n, end=' ')
    print()
```

이중 for문 – 구구단 전체

■ 구구단 전체 출력 프로그램

```
for i in range(1, 10):  
    print("[", i, "단"])  
    for j in range(1, 10):  
        print("%d x %d = %d" % (i, j, i*j))  
    print()
```

```
[ 2 단]  
2 x 1 = 2  
2 x 2 = 4  
2 x 3 = 6  
2 x 4 = 8  
2 x 5 = 10  
2 x 6 = 12  
2 x 7 = 14  
2 x 8 = 16  
2 x 9 = 18
```

```
[ 3 단]  
3 x 1 = 3  
3 x 2 = 6  
3 x 3 = 9  
3 x 4 = 12  
3 x 5 = 15  
3 x 6 = 18  
3 x 7 = 21  
3 x 8 = 24  
3 x 9 = 27
```

이중 for문 – 구구단 전체

■ 구구단 전체 출력 프로그램

```
for i in range(1, 10):  
    for j in range(2, 10):  
        print("%d x %d = %d" % (j, i, i*j), end='  ')  
    print()
```

2 x 1 = 2	3 x 1 = 3	4 x 1 = 4	5 x 1 = 5
2 x 2 = 4	3 x 2 = 6	4 x 2 = 8	5 x 2 = 10
2 x 3 = 6	3 x 3 = 9	4 x 3 = 12	5 x 3 = 15
2 x 4 = 8	3 x 4 = 12	4 x 4 = 16	5 x 4 = 20
2 x 5 = 10	3 x 5 = 15	4 x 5 = 20	5 x 5 = 25
2 x 6 = 12	3 x 6 = 18	4 x 6 = 24	5 x 6 = 30
2 x 7 = 14	3 x 7 = 21	4 x 7 = 28	5 x 7 = 35
2 x 8 = 16	3 x 8 = 24	4 x 8 = 32	5 x 8 = 40
2 x 9 = 18	3 x 9 = 27	4 x 9 = 36	5 x 9 = 45

자리 배치도

```
입장객 수 : 44
열 수 : 5
좌석1 좌석2 좌석3 좌석4 좌석5
좌석6 좌석7 좌석8 좌석9 좌석10
좌석11 좌석12 좌석13 좌석14 좌석15
좌석16 좌석17 좌석18 좌석19 좌석20
좌석21 좌석22 좌석23 좌석24 좌석25
좌석26 좌석27 좌석28 좌석29 좌석30
좌석31 좌석32 좌석33 좌석34 좌석35
좌석36 좌석37 좌석38 좌석39 좌석40
좌석41 좌석42 좌석43 좌석44
```

```
customer_num = int(input("입장객 수 입력 : "))
col_num = int(input("좌석 열의 수 입력: "))

if customer_num % col_num == 0:
    row_num = customer_num / col_num
else:
    row_num = customer_num // col_num + 1
    #row_num = int(customer_num / col_num + 1)
print("줄의 수: ", row_num)

print('자리 배치도')

for i in range(0, row_num):
    for j in range(1, col_num+1):
        seat_num = i*col_num+j
        if seat_num > customer_num:
            break
        print(seat_num, end=' ')

    print()
```