

Python

제어문(조건문, 반복문)

“

스몰사이즈(반지름 약 4인치) 피자를 8등분하여 두 조각을 담은 접시가 있다. 그리고 라지사이즈(반지름 약 8인치) 피자를 마칭가지로 8등분해서 한 조각을 담은 접시가 있다. 배고픈 당신은 양이 많은 것을 선택하려고 한다. 어/떤 것을 선택하겠는가?



”

스몰사이즈 = 반지름 약 4인치

라지사이즈 = 반지름 약 8인치

스몰사이즈의 넓이(원의 넓이) = $\pi \times r^{**2}$

스몰사이즈의 한 조각크기 = 스몰사이즈의 넓이 / 8

스몰사이즈의 두 조각크기 = 스몰사이즈의 한 조각크기 \times 2

라지사이즈의 넓이(원의 넓이) = $\pi \times r^{**2}$

라지사이즈의 한 조각크기 = 라지사이즈의 넓이 / 8

$\pi = 3.14$

small_pizza = 4

large_pizza = 8

area_small = $\pi * \text{small_pizza} ** 2$

area_large = $\pi * \text{large_pizza} ** 2$

one_piece_small = area_small / 8

one_piece_large = area_large / 8

print("I want to eat", "Large Pizza" if one_piece_small * 2 < one_piece_large else "Small Pizza")

print("I want to eat", one_piece_small * 2 < one_piece_large and "Large Pizza" or "Small Pizza")

```
import math
```

```
small_pizza = 4
```

```
large_pizza = 8
```

```
area_small = math.pi * small_pizza ** 2
```

```
area_large = math.pi * large_pizza ** 2
```

```
one_piece_small = area_small / 8
```

```
one_piece_large = area_large / 8
```

```
print("I want to eat", "Large Pizza" if one_piece_small * 2 <  
one_piece_large else "Small Pizza")
```

```
print("I want to eat", one_piece_small * 2 < one_piece_large and "Large  
Pizza" or "Small Pizza")
```

제어문

```
if boolean_value :  
    boolean_value가 True일 때 처리할 문장들  
...
```

if *boolean_value* :
 boolean_value가 True일 때 처리할 문장들 } Block
 ...

들여쓰기 indentation

Blanks (space bar or tab)

4 blanks를 권장

if *boolean_value* :

→ *boolean_value*가 True일 때 처리할 문장들 } Block
...

else:

→ *boolean_value*가 False일 때 처리할 문장들 } Block
...

→ 여백! indentation

Blanks (space bar or tab)

if *boolean_value_1* :

→ *boolean_value_1*가 True일 때 처리할 문장들
...

} Block

elif *boolean_value_2*:

→ *boolean_value_2*가 True일 때 처리할 문장들
...

} Block

else:

→ *boolean_value_1*과 *boolean_value_2*가 모두 False일 때
...

} Block

```
import math
```

```
small_pizza = 4
```

```
large_pizza = 8
```

```
area_small = math.pi * small_pizza ** 2
```

```
area_large = math.pi * large_pizza ** 2
```

```
one_piece_small = area_small / 8
```

```
one_piece_large = area_large / 8
```

```
print("I want to eat", "Large Pizza" if one_piece_small * 2 <  
one_piece_large else "Small Pizza")
```

이 부분을 if문을 사용해서 바꿔보자

“

아이스크림을 사서 동네 마트에 온 한 학생, 마침 좋아하는
큰 아이스크림이 2 + 1 행사를 하고 있습니다. 그래서 아이스크림
3개를 장바구니에 넣으려는데, 옆을 보니 다른 컵 아이스크림이
30% 할인하고 있습니다. 그럼 2 + 1을 사야 할까요? 30%
할인제품을 사야 할까요?

”

```
price = 1000
```

```
price_2_1 = 2000
```

```
price_30dc = 3000 * (1 - 30/100)
```

```
print(price_2_1 < price_30dc and  
"2 + 1" or "30%dc", "is cheaper ")
```

} 이 부분을 if문을 사용해서 바꿔보자

“

매일 아침 출근길에 커피전문점에 들러 아메리카노를 한 잔을
사는 박 대리. 아주 고민 없이 커피 한잔을 사는데 문득 생각
이 스칩니다. 가장 작은 사이즈(8온스)는 3600원이고 가장 큰
사이즈(20온스)는 5100원인데 어느 것을 사는 것이 이득 일까요?
(1온스는 약 30ml)

”

```
size_large = 20
size_small = 8
price_large = 5100
price_small = 3600
price_peroz_large = price_large / size_large
price_peroz_small = price_small / size_small

print("Large" if price_peroz_large <
price_peroz_small else "Small", "is cheaper") }
```

이 부분을 if문을 사용해서 바꿔보자

“

동일한 물건은 쇼핑몰 A에서는 20%할인인 쿠폰으로 구매 할 수 있고, 쇼핑몰 B에서는 10%와 11%할인인 쿠폰을 중복사용하여 구매할 수 있습니다. 10,000원짜리 제품을 어느 쇼핑몰에서 더 싸게 구매 할 수 있을까요?

”


```
price = 10000
rate_a = 1 - 20 / 100
rate_b1 = 1 - 10 / 100
rate_b2 = 1 - 11 / 100
price_a = price * rate_a
price_b = price * rate_b1
price_b *= rate_b2
```

이 부분을 if문을 사용해서 바꿔보자

```
print( price_a < price_b and "A  
shopping Mall" or "B shopping Mall") }
```

```
while boolean_value :  
    boolean_value가 True인 동안 처리할 문장들 } Block  
    ...
```

```
i = 10
while i < 100 :
    i += 1
print(i)
```

```
while boolean_value :  
    boolean_value가 True인 동안 처리할 내용  
    ...  
else:  
    while 반복이 정상적으로 끝난 후 처리할 내용
```

```
i = 10
while i < 100 :
    i += 1
else:
    i = 200
print(i)
```

while *boolean_value* :

*boolean_value*가 True인 동안 처리할 내용

if *boolean_value* :

break

} Block

else:

while 반복이 정상적으로 끝난 후 처리할 내용

} Block

가장 가까운 loop block 밖으로 빠져나간다

```
i = 10
while i < 100 :
    i += 1
    if i > 70:
        break
else:
    i = 200
print(i)
```

“

철수는 2000년 11월 3일에 태어났다.
오늘(2020년 11월 21일)은 태어난지 20개월째 되는 날일까?

”


```
print("오늘")
yearOfToday = input("Year : ")
monthOfToday = input("Month : ")
dayOfToday = input("Day : ")
yearOfToday = int(yearOfToday)
monthOfToday = int(monthOfToday)
dayOfToday = int(dayOfToday)

print("생년월일 :")
yearOfBirthday = input("Year : ")
monthOfBirthday = input("Month : ")
dayOfBirthday = input("Day : ")
yearOfBirthday = int(yearOfBirthday)
monthOfBirthday = int(monthOfBirthday)
dayOfBirthday = int(dayOfBirthday)
```

첫번째 힌트.

먼저 태어난 달의 남을 일수 계산을 위해 다음과 같은 코드가 필요하다.

days는 우리가 구하려는 값이다.

```
days = 0
```

```
if ( monthOfBirthday == 1 ) :  
    days = days + 31 - dayOfBirthday  
elif ( monthOfBirthday == 2 ) :  
    days = days + 28 - dayOfBirthday  
elif ( monthOfBirthday == 3 ) :  
    days = days + 31 - dayOfBirthday  
elif ( monthOfBirthday == 4 ) :  
    days = days + 30 - dayOfBirthday  
elif ( monthOfBirthday == 5 ) :  
    days = days + 31 - dayOfBirthday  
elif ( monthOfBirthday == 6 ) :  
    days = days + 30 - dayOfBirthday
```

```
elif ( monthOfBirthday == 7 ) :  
    days = days + 31 - dayOfBirthday  
elif ( monthOfBirthday == 8 ) :  
    days = days + 31 - dayOfBirthday  
elif ( monthOfBirthday == 9 ) :  
    days = days + 30 - dayOfBirthday  
elif ( monthOfBirthday == 10 ) :  
    days = days + 31 - dayOfBirthday  
elif ( monthOfBirthday == 11 ) :  
    days = days + 30 - dayOfBirthday  
else: # is 12  
    days = days + 31 - dayOfBirthday
```

두번째 힌트.

태어난 달의 다음 달부터 그 해 남은 달의 날짜를 계산해야 한다.

```
currentMonth = monthOfBirthday + 1
while (currentMonth <= 12) :
    if (currentMonth == 1) :
        days = days + 31
    elif (currentMonth == 2) :
        days = days + 28
    elif (currentMonth == 3) :
        days = days + 31
    elif (currentMonth == 4) :
        days = days + 30
    elif (currentMonth == 5) :
        days = days + 31
    elif (currentMonth == 6) :
        days = days + 30
```

```
    elif (currentMonth == 7) :
        days = days + 31
    elif (currentMonth == 8) :
        days = days + 31
    elif (currentMonth == 9) :
        days = days + 30
    elif (currentMonth == 10) :
        days = days + 31
    elif (currentMonth == 11) :
        days = days + 30
    else: # is 12
        days = days + 31

currentMonth+=1
```

세번째 힌트.

이제 온전히 한 해의 날 수를 갖는 태어난 해와 올해의 사이에 있는 해의 날 수들만
더해주면 된다.

```
yearOfBirthday += 1  
while (yearOfBirthday < yearOfToday) :  
    days = days + 365
```


네번째 힌트.

오늘이 속한 달 이전 달까지의 날짜 수를 합한다.

```
currentMonth = 1
while (currentMonth <= monthOfToday) :
    if (currentMonth == 1 ) :
        days = days + 31
    elif (currentMonth == 2 ) :
        days = days + 28
    elif (currentMonth == 3 ) :
        days = days + 31
    ...
    else: # is 12
        days = days + 31

currentMonth+=1
```

다섯번째 힌트.

오늘 날짜를 더하면 된다.

```
days += dayOfToday
```

운년을 고려해야 한다.

윤년 ?

1. 연도를 4로 나누어 떨어진다.
2. 단, 100으로 나누어 떨어지는 경우는 제외한다.
3. 하지만 연도가 400으로 나누어지면 포함시킨다.