

05

CHAPTER

조건문

학습목표

- 조건문의 대표적인 기본 if 문, 중첩 if 문의 형식과 사용법을 익힌다.
- 조건문을 다양하게 활용하는 방법을 익힌다.
- 조건문과 더불어 간단한 반복문도 다룬다.
- 조건문으로 무지개 색상의 원을 그리는 프로그램과 종합 계산기 프로그램을 만든다.

SECTION 01 이 장에서 만들 프로그램

SECTION 02 기본 if 문

SECTION 03 중첩 if 문

SECTION 04 if 문 응용

요약

연습문제

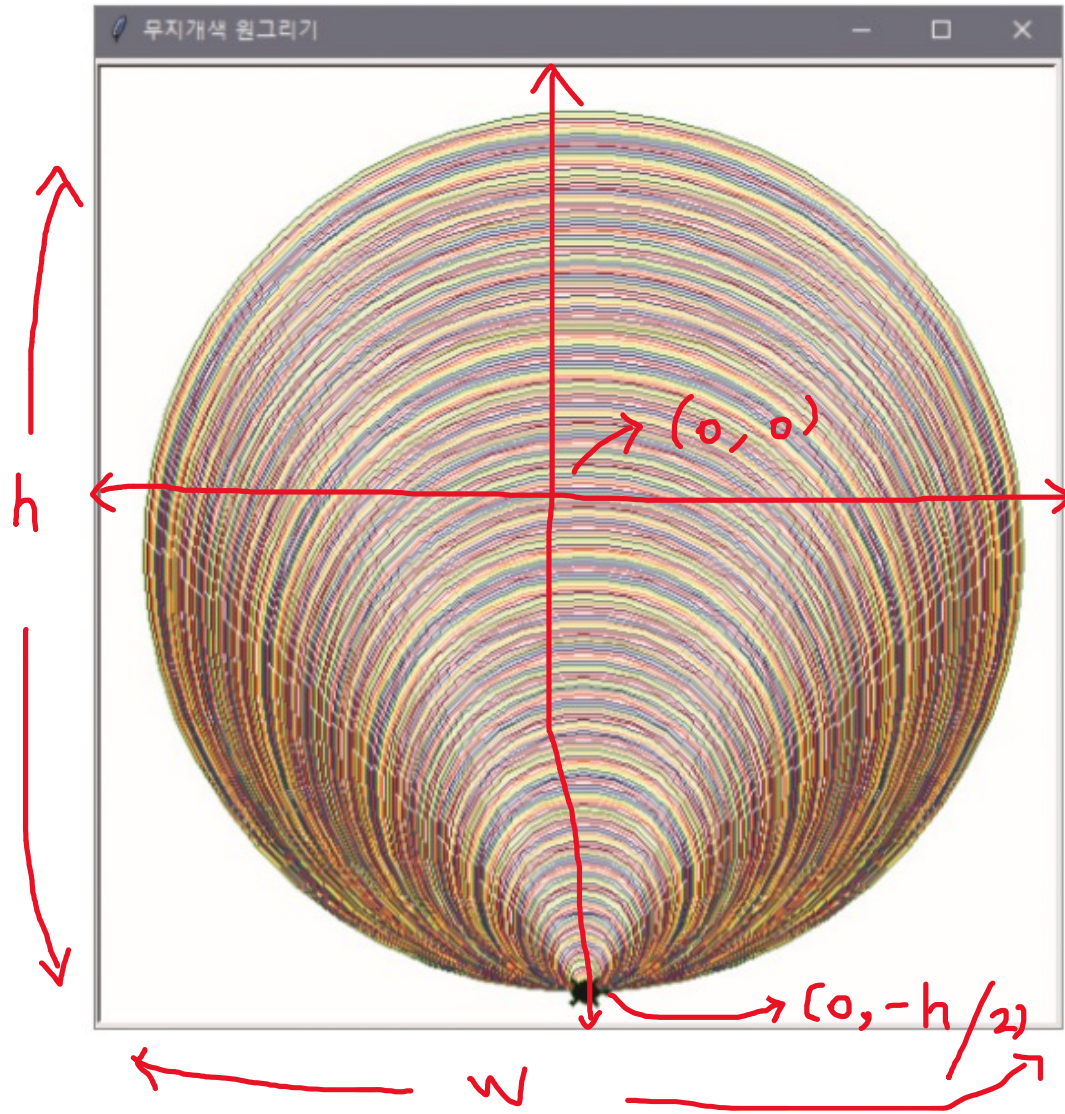
응용예제



Section01 이 장에서 만들 프로그램

■ [프로그램1] 무지개 색상의 원

- if 문을 사용해 터틀 그래픽에서 무지개 색상의 원을 그리는 프로그램



Section01 이 장에서 만들 프로그램

■ [프로그램 2] 종합 계산기

- 기능이 두 가지인 종합 계산기 프로그램

```
Python 3.6.2 Shell
File Edit Shell Debug Options Window Help

===== RESTART: C:\CookPython\Code05-11.py =====
1. 입력한 수식 계산 2. 두 수 사이의 합계 : 1
*** 수식을 입력하세요 : 3*4/2-5
3*4/2-5 결과는 1.0입니다.
>>>

===== RESTART: C:\CookPython\Code05-11.py =====
1. 입력한 수식 계산 2. 두 수 사이의 합계 : 2
*** 첫 번째 숫자를 입력하세요 : 1
*** 두 번째 숫자를 입력하세요 : 10
1+...+10는 55입니다.
>>> |
```

Annotations in the image:

- Red box around `3*4/2-5` with a red arrow pointing to it from the text "사용자가 입력한 값" (Value entered by the user).
- Red text "`eval()` 함수" (eval() function) with a red arrow pointing to the expression `3*4/2-5`.
- Red box around the input `1` with a red arrow pointing to it from the text "사용자가 입력한 값" (Value entered by the user).
- Red box around the input `10` with a red arrow pointing to it from the text "사용자가 입력한 값" (Value entered by the user).

Ln: 14 Col: 4

Section02 기본 if 문

■ if 문

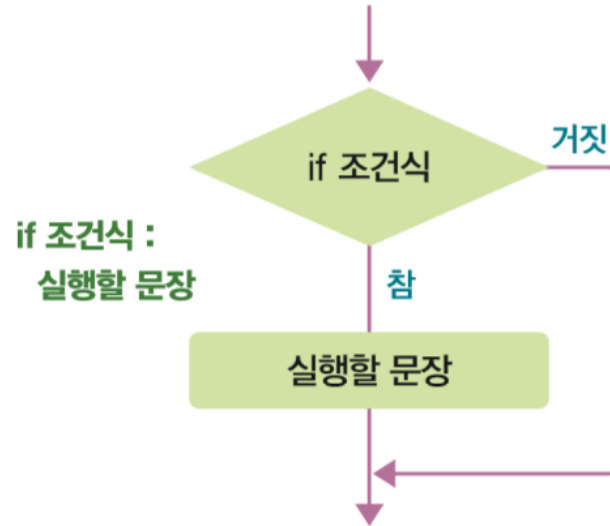


그림 5-1 if 문의 형식과 순서도

```
a = 99
if a < 100 :
    print("100보다 작군요.")
```

출력 결과

100보다 작군요.

Tip • 파이썬은 들여쓰기가 매우 중요. if 문 다음에 '실행할 문장'은 if 문 다음 줄에서 들여쓰기를 해서 작성. 들여쓰기 할 때는 Tab 보다 Space Bar 를 눌러 4칸 정도로 들여쓰기 권장, 대화형 모드에서는 '실행할 문장' 모두 끝나고 Enter 2번 눌러야 if 문이 끝나는 것으로 간주

Section02 기본 if 문

- if 문 실행 과정

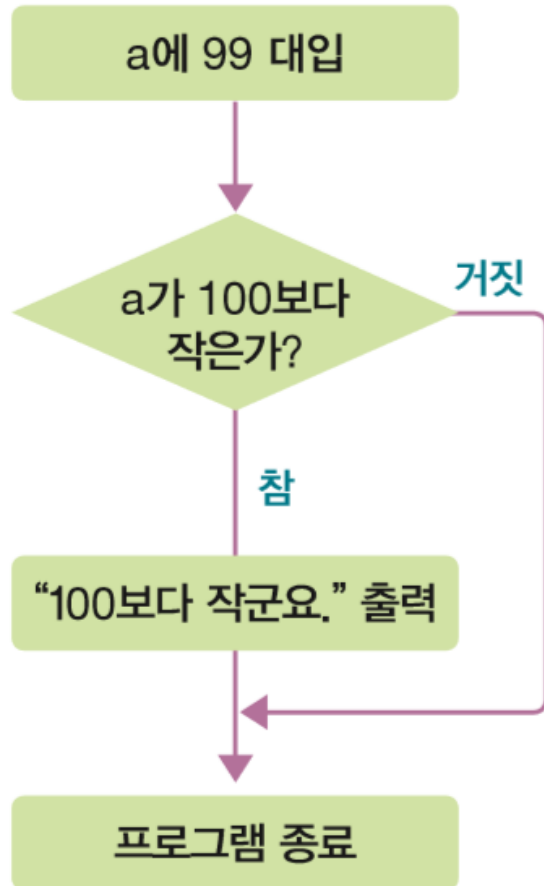


그림 5-2 if 문 실행 과정

Section02 기본 if 문

- 예 : 조건이 참이고 실행할 문장이 2개일 때

Code05-01.py

```
1 a = 200
2
3 if a < 100 :
4     print("100보다 작군요.")
5 print("거짓이므로 이 문장은 안 보이겠죠?")
6
7 print("프로그램 끝")
```

출력 결과

거짓이므로 이 문장은 안 보이겠죠?
프로그램 끝

```
3 if a < 100 :
4     print("100보다 작군요.")
5
6 print("거짓이므로 이 문장은 안 보이겠죠?")
7 print("프로그램 끝")
```

Section02 기본 if 문

- 예 : if 문에서 두 문장 이상을 실행하고자 할 때

Code05-02.py

```
1 a = 200
2
3 if a < 100 :
4     print("100보다 작군요.")
5     print("거짓이므로 이 문장은 안 보이겠죠?")
6
7 print("프로그램 끝")
```

출력 결과

프로그램 끝

Tip • 들여쓰기 오류 예

```
if a < 100 :
    print("100보다 작군요.")
print("거짓이므로 이 문장은 안 보이겠죠?")
```

Section02 기본 if 문

■ if~else 문

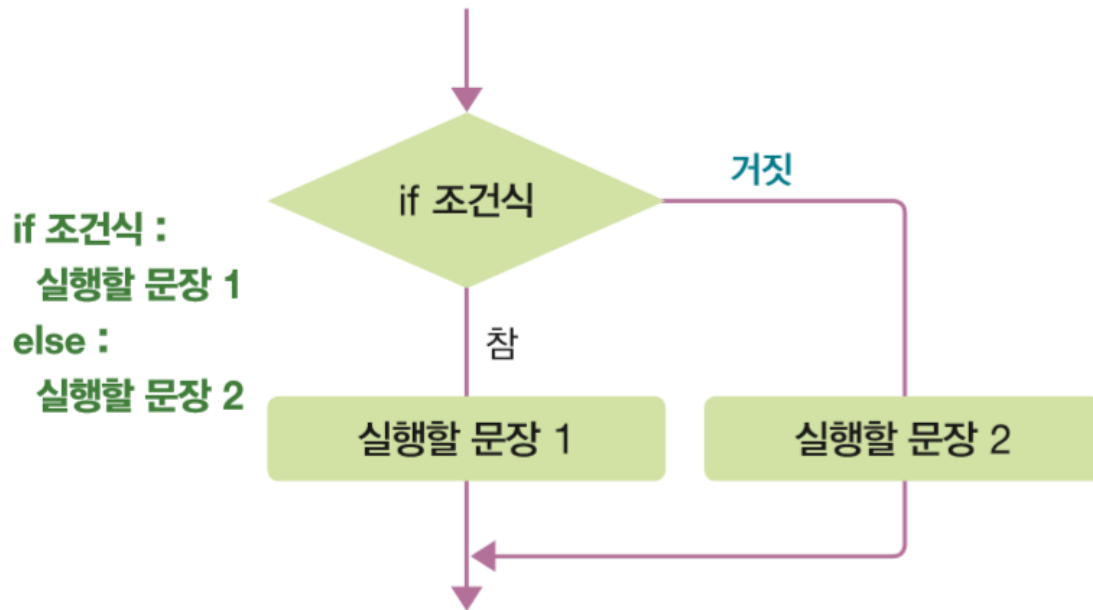


그림 5-3 if~else 문의 형식과 순서도

- 조건이 참일 때와 거짓일 때 실행할 문장이 다름

Section02 기본 if 문

Code05-03.py

```
1 a = 200
2
3 if a < 100 :
4     print("100보다 작군요.")
5 else :
6     print("100보다 크군요.")
```

출력 결과

100보다 크군요.

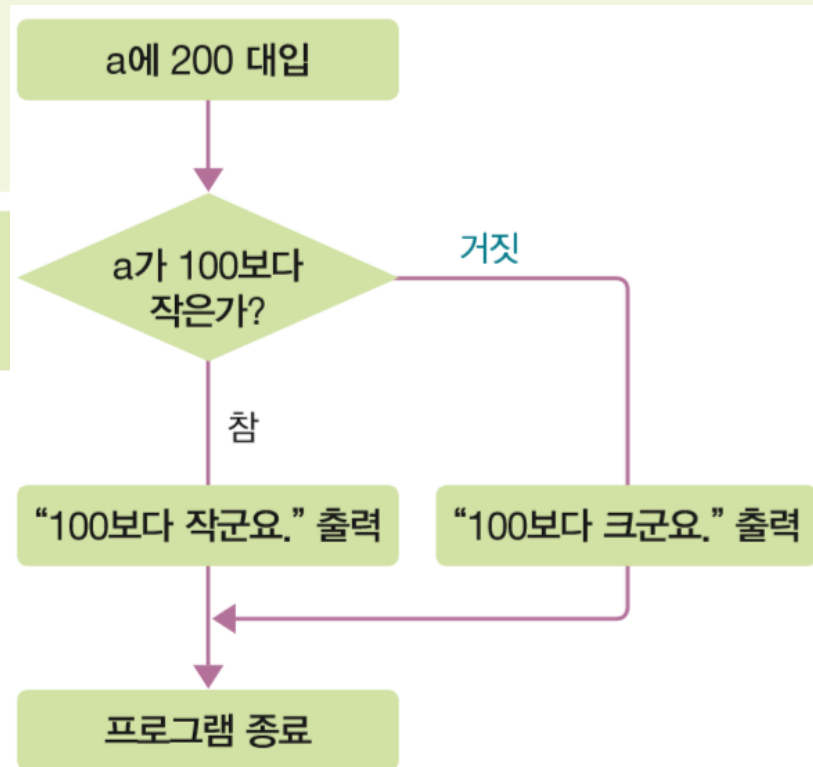


그림 5-4 Code05-03.py 실행 과정

Section02 기본 if 문

Code05-04.py

```
1  a = 200
2
3  if a < 100 :
4      print("100보다 작군요.")
5      print("참이면 이 문장도 보이겠죠?")
6  else :
7      print("100보다 크군요.")
8      print("거짓이면 이 문장도 보이겠죠?")
9
10 print("프로그램 끝")
```

출력 결과

100보다 크군요.
거짓이면 이 문장도 보이겠죠?
프로그램 끝

Section02 기본 if 문

- 예 : 입력 숫자가 짝수인지 홀수인지 계산

Code05-05.py

```
1 a = int(input("정수를 입력하세요 : "))
2
3 if a % 2 == 0 :
4     print("짝수를 입력했군요.")
5 else :
6     print("홀수를 입력했군요.")
```

출력 결과

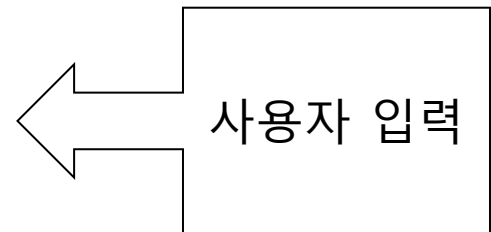
정수를 입력하세요 : 125

홀수를 입력했군요.

실습 문제

- 정수를 입력 받아 짝수인지, 홀수인지 판별하여 출력하시오. 'q'를 입력하면 프로그램이 종료 되도록!

정수를 입력하세요: 123
홀수 입니다



Section03 중첩 if 문

■ if~else~if~else 문

- if 문을 한 번 실행한 후 그 결과에서 if 문을 다시 실행하는 것

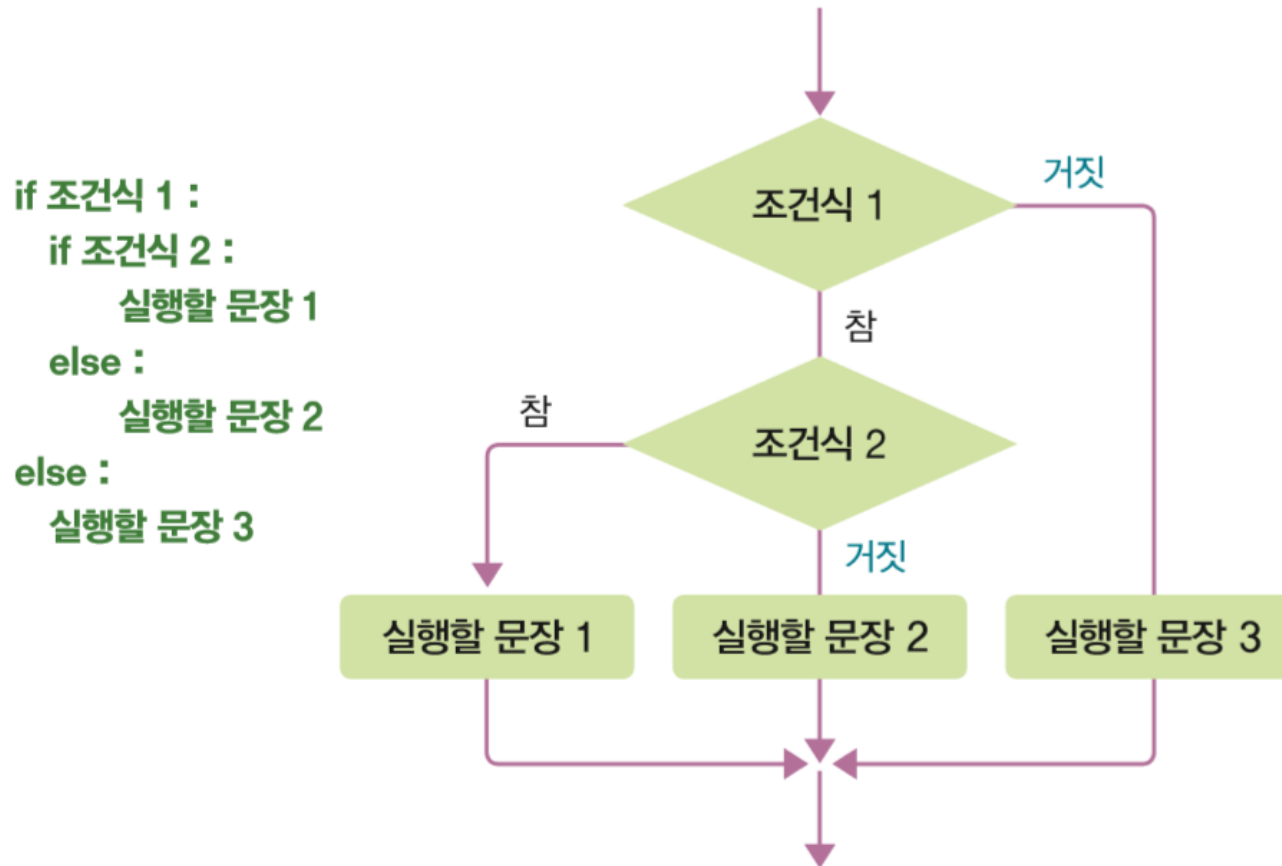


그림 5-5 중첩 if 문의 형식과 순서도

Section03 중첩 if 문

Code05-06.py

```
1  a = 75
2
3  if a > 50 :
4      if a < 100 :
5          print("50보다 크고 100보다 작군요.")
6
7      else :
8          print("와~~ 100보다 크군요.")
9  else :
10     print("에고~ 50보다 작군요.")
```

출력 결과

50보다 크고 100보다 작군요.

Section03 중첩 if 문

- Code05-07.py를 그림으로 표현

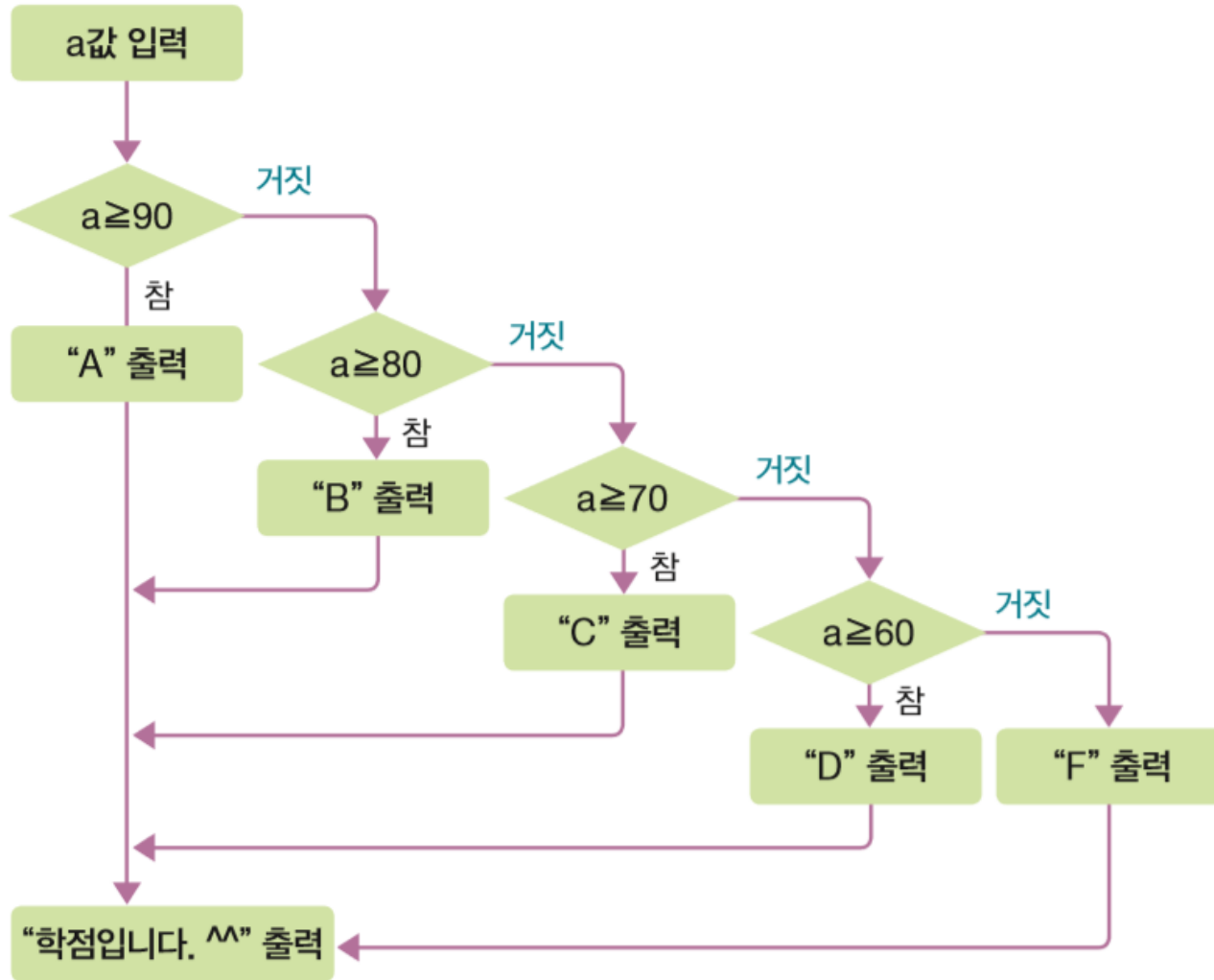


그림 5-6 Code05-07.py 실행 과정

Section03 중첩 if 문

■ if~elif~else 문

Code05-08.py

```
1 score = int(input("점수를 입력하세요 : "))
2
3 if score >= 90 :
4     print("A")
5 elif score >= 80 :
6     print("B")
7 elif score >= 70 :
8     print("C")
9 elif score >= 60 :
10    print("D")
11 else:
12    print("F")
13
14 print("학점입니다. ^^")
```

Handwritten annotations in red:

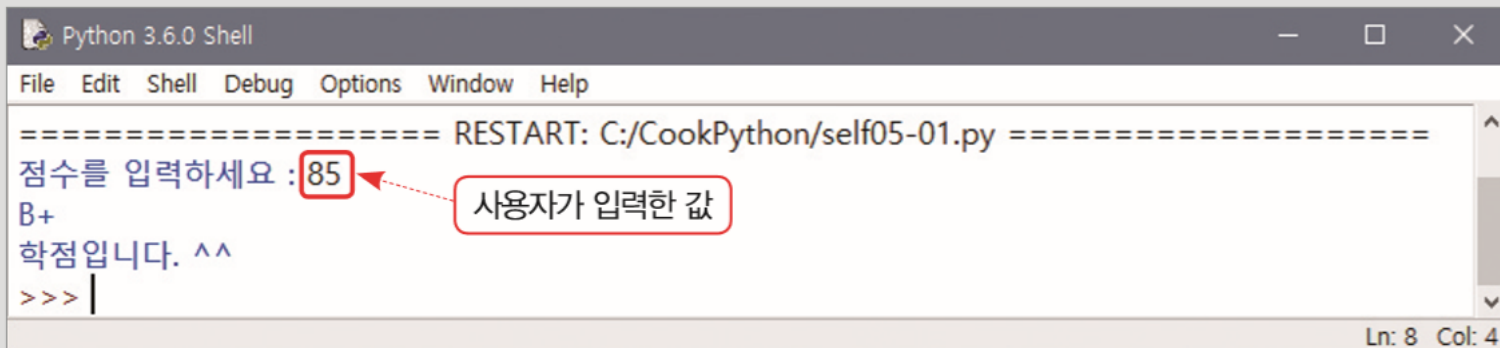
- 90 이상 (points to line 3)
- 80 ~ 89 (points to line 5)
- 70 ~ 79 (points to line 7)
- 60 ~ 69 (points to line 9)
- 59 이하 (points to line 11)

SELF STUDY 5-1

Code05-08.py를 다음 조건처럼 좀 더 세분화해 보자.

95점 이상 : A+, 90점 이상 : A0, 85점 이상 : B+, 80점 이상 : B0,

75점 이상 : C+, 70점 이상 : C0, 65점 이상 : D+, 60점 이상 : D0, 60점 미만 : F



```
Python 3.6.0 Shell
File Edit Shell Debug Options Window Help
===== RESTART: C:/CookPython/self05-01.py =====
점수를 입력하세요 : 85
B+
학점입니다. ^^
>>>
```

사용자가 입력한 값

Ln: 8 Col: 4

Section03 중첩 if 문

■ 삼항 연산자를 사용한 if 문

```
1 jumsu = 55
2 res = ''
3 if jumsu >= 60 :
4     res = '합격'
5 else :
6     res = '불합격'
7 print(res)
```

■ 3~6행 줄임

```
res = '합격' if jumsu >= 60 else '불합격'
```



Section03 중첩 if 문

■ [프로그램 1]의 완성

Code05-09.py

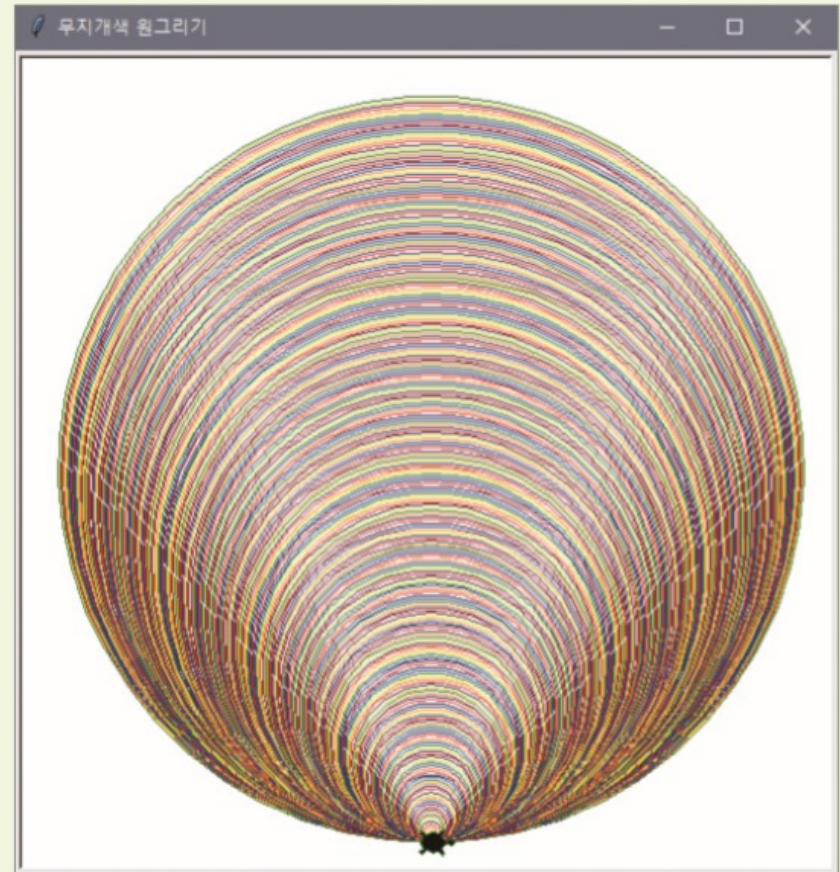
```
1 import turtle
2
3 ## 전역 변수 선언 부분 ##
4 swidth, sheight = 500, 500
5
6 ## 메인 코드 부분 ##
7 turtle.title('무지개색 원그리기')
8 turtle.shape('turtle')
9 turtle.setup(width = swidth + 50, height = sheight + 50)
10 turtle.screensize(swidth, sheight)
11 turtle.penup()
12 turtle.goto(0, -sheight / 2)
13 turtle.pendown()
14 turtle.speed(10)
15
```

4행 : 창 크기에 사용할 변수 준비
7~10행 : 윈도우창 설정
11~13행 : 거북이를 가운데 아래쪽으로 이동
14행 : 거북이 속도 설정

Section03 중첩 if 문

```
16 for radius in range(1, 250) :
17     if radius % 6 == 0 :
18         turtle.pencolor('red')
19     elif radius % 5 == 0 :
20         turtle.pencolor('orange')
21     elif radius % 4 == 0 :
22         turtle.pencolor('yellow')
23     elif radius % 3 == 0 :
24         turtle.pencolor('green')
25     elif radius % 2 == 0 :
26         turtle.pencolor('blue')
27     elif radius % 1 == 0 :
28         turtle.pencolor('navyblue')
29     else :
30         turtle.pencolor('purple')
31
32     turtle.circle(radius)
33
34 turtle.done()
```

16~32행 : 반지름(radius) 1에서 249까지 원을 반복해 그림
17~30행 : 반지름에 따라 빨주노초파남보 색상이 반복 설정
32행 : 거북이 원을 그림



Section04 if 문 응용

■ 리스트와 함께 사용 : List append(), pop()

- 리스트(List) : 데이터 여러 개를 한곳에 담아 놓은 것
- 방법 : 대괄호 []로 묶고 그 안에 필요한 것들을 한꺼번에 넣음
- 예 : fruit 변수에 값 4개를 리스트 하나로 묶어 대입

```
fruit = ['사과', '배', '딸기', '포도']  
print(fruit)
```

출력 결과

```
['사과', '배', '딸기', '포도']
```

• 추가

```
fruit.append('귤')  
print(fruit)
```

출력 결과

```
['사과', '배', '딸기', '포도', '귤']
```

Section04 if 문 응용

- if 항목 in 리스트 : 리스트에 해당 항목이 있다면 True를 반환

출력 결과

딸기가 있네요. ^^

```
if '딸기' in fruit :  
    print("딸기가 있네요. ^^")
```

Section04 if 문 응용

- 예 : 0부터 9까지 숫자 중에서 리스트 안에 없는 숫자 찾기

Code05-10.py

```
1 import random
2
3 numbers = []
4 for num in range(0, 10) :
5     numbers.append(random.randrange(0, 10))
6
7 print("생성된 리스트", numbers)
8
9 for num in range(0, 10) :
10     if num not in numbers :
11         print("숫자 %d는(은) 리스트에 없네요." %num)
```

3행 : 빈 리스트인 numbers 준비
4행과 9행 : 각각 0부터 9까지 총 10회를 반복
5행 : 0~9의 숫자 총 10개를 numbers 리스트에 만듦
7행 : 생성된 리스트 출력
9행 : 0부터 9까지의 숫자를 num에 넣음
10행 : numbers 리스트에 해당 숫자가 없다면 11행에서 숫자 없다는 메시지 출력

출력 결과

생성된 리스트 [5, 8, 8, 7, 8, 1, 9, 0, 0, 4]

숫자 2는(은) 리스트에 없네요.

숫자 3는(은) 리스트에 없네요.

숫자 6는(은) 리스트에 없네요.

List : pop()

```
count = 0
while (count < 3):
    count = count + 1
    print("Hello Python")
```

```
print('\n')
```

```
a = [1, 2, 3, 4]
while a:
    print(a.pop())
```

→ pop 순서 (stack)

output:

```
Hello Python
Hello Python
Hello Python
```

```
4
3
2
1
```

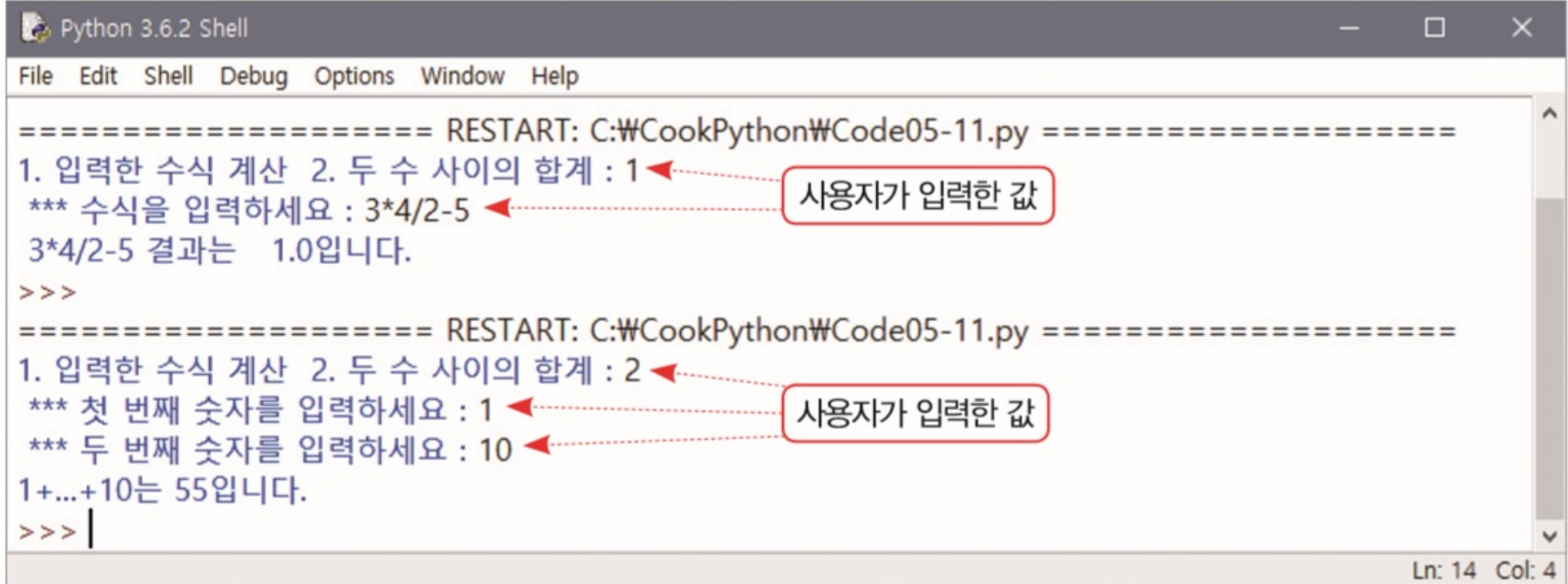

Section04 if 문 응용

■ [프로그램 2]의 완성

Code05-11.py

```
1  ## 변수 선언 부분 ##
2  select, answer, numStr, num1, num2 = 0, 0, "", 0, 0
3
4  ## 메인 코드 부분 ##
5  select = int(input("1. 입력한 수식 계산 2. 두 수 사이의 합계 : "))
6
7  if select == 1 :
8      numStr = input(" *** 수식을 입력하세요 : ")
9      answer = eval(numStr)
10     print(" %s 결과는 %5.1f입니다. " % (numStr, answer))
11 elif select == 2 :
12     num1 = int(input(" *** 첫 번째 숫자를 입력하세요 : "))
13     num2 = int(input(" *** 두 번째 숫자를 입력하세요 : "))
14     for i in range(num1, num2 + 1) :
15         answer = answer + i
16     print("%d+...+%d는 %d입니다. " % (num1, num2, answer))
17 else :
18     print("1 또는 2만 입력해야 합니다.")
```

Section04 if 문 응용



```
Python 3.6.2 Shell
File Edit Shell Debug Options Window Help

===== RESTART: C:\CookPython\Code05-11.py =====
1. 입력한 수식 계산 2. 두 수 사이의 합계 : 1
*** 수식을 입력하세요 : 3*4/2-5
3*4/2-5 결과는 1.0입니다.
>>>

===== RESTART: C:\CookPython\Code05-11.py =====
1. 입력한 수식 계산 2. 두 수 사이의 합계 : 2
*** 첫 번째 숫자를 입력하세요 : 1
*** 두 번째 숫자를 입력하세요 : 10
1+...+10는 55입니다.
>>> |
```

사용자가 입력한 값

사용자가 입력한 값

Ln: 14 Col: 4

Section04 if 문 응용

SELF STUDY 5-2

[프로그램 2]의 두 번째 기능처럼 두 숫자를 입력받고 두 숫자 사이의 합계를 구하는 프로그램을 만들어 보자. 단 1씩 증가하지 않고 증가하는 숫자도 입력받는다. 예를 들어 1, 100, 3을 입력하면 $1+4+\dots+100$ 의 합계를 구한다.

힌트 range(시작값, 끝값+1, 증가값) 형식으로 사용한다.



```
Python 3.6.0 Shell
File Edit Shell Debug Options Window Help
===== RESTART: C:/CookPython/self05-02.py =====
*** 첫 번째 숫자를 입력하세요 : 1
*** 두 번째 숫자를 입력하세요 : 100
*** 더할 숫자를 입력하세요 : 3
1+4+...+100는 1717입니다.
>>> |
```

사용자가 입력한 값

Ln: 9 Col: 4



Thank You
