

문제해결을 위한 코딩 첫걸음

⋮ 7장 반복문에 대해 알아봅시다(1)_for 반복문



학습 목표

- 반복문의 필요성에 대해 이해합니다.
- for문과 range문의 사용에 대해 이해합니다.
- while문 사용에 대해 이해합니다.
- break문, continue문과 다중 반복에 대해서 이해합니다.



7.1 반복문이란?

- 반복문은 왜 필요할까?
 - 반복되는 작업의 명령문을 모두 작성할 경우
 - 프로그램의 길이가 길어지고 오류 수정이 어려워짐
 - for ~ range() 또는 while 문과 같은 반복문 이용 단순화
 - 한 선의 길이가 90픽셀인 사각형 그리기

[소스코드] 7-1.py

```
import turtle
```

1번

```
turtle.forward(90)
```

```
turtle.right(90) # 1 times
```

2번

```
turtle.forward(90)
```

```
turtle.right(90) # 2 times
```

3번

```
turtle.forward(90)
```

```
turtle.right(90) # 3 times
```

4번

```
turtle.forward(90)
```

```
turtle.right(90) # 4 times
```

[소스코드] 7-2.py

```
import turtle
```

```
for x in range(4) :
```

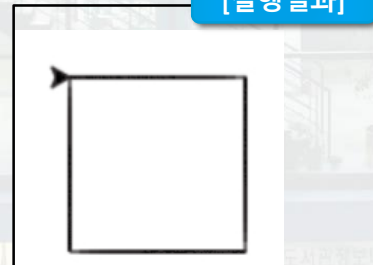
0~3까지 4번
처리문을 반복

1번

```
turtle.forward(90)
```

```
turtle.right(90)
```

[실행 결과]



7.1 반복문이란? – for문과 range문

- for문과 range문 사용하기
 - range(4)는 리스트 [0,1,2,3]과 동일

변수명
↓
for **x** in range(4) :
 print(x)

← x 변수를 이용하여 x=0부터 x=3까지 4번 처리문을 반복

동일
↓
for **x** in [0,1,2,3] :
 print(x)

[실행결과]

0
1
2
3

- x와 같은 명명된 변수 대신 익명변수 '_' 사용 가능

익명 변수
↓
for **_** in range(3):
 print("Hello Python")

[실행결과]

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




7.1 따라 해보기 - 합계 구하기

- 리스트 안에 저장된 4개의 숫자 [3, 5, 8, 10]의 합계 구하기

합계를 저장하기 위한 변수

```
sum = 0
```

합계를 구하기 위해 초기값을 0으로 설정

```
for n in [3, 5, 8, 10]:
```

```
    sum = sum + n
```

sum 변수 값과 n변수 값을 더하여 sum 변수에 다시 저장

```
print("The sum is",sum)
```

[실행결과]

```
The sum is 26
```



7.2 for문

- 파이썬에서 사용되는 반복문 종류
 - for

```
for 변수 in 시퀀스 :
```

시퀀스의 값이 처음부터 하나씩 변수에 대입되어 실행됨

```
    실행문장1
```

```
    ...
```

- while

```
while condition :
```

condition의 결과가 참인 경우 문장을 반복하여 수행

```
    실행문장1
```

```
    ...
```



7.2 for문 - 따라 해보기

- 리스트에 저장된 이름을 차례로 출력하는 프로그램

[소스코드] 7-5.py

```
for name in ['John', 'Smith', 'Kim'] :  
    print(name)
```

[실행결과]

```
John  
Smith  
Kim
```

- 문자열에 저장된 문자를 하나씩 차례로 출력하는 프로그램

문자열도 시퀀스

[소스코드] 7-6.py

```
for ch in "abcde" :  
    print(ch, end=" ")
```

[실행결과]

```
a b c d e
```

종결문자 지정
줄 바꿈대신 공백문자(" ") 출력



7.2 for문 - range문

- range()문
 - 사용자가 원하는 범위의 정수를 발생
 - 형식

```
range([start,] stop [,step])
```

- start: 시작 정수, 기본 값(default value)은 0
- stop: stop 미만의 값 발생
- step: 발생하는 정수의 간격, 기본 값은 1



7.2 for문 - range문 예시

- range()문의 출력 예시

소스 코드	실행 결과
<code>print list(range(5))</code>	<code>[0, 1, 2, 3, 4]</code>
<code>print list(range(0, 5, 1))</code>	<code>[0, 1, 2, 3, 4]</code>
<code>print list(range(1, 5, 1))</code>	<code>[1, 2, 3, 4]</code>
<code>print list(range(0, 5, 2))</code>	<code>[0, 2, 4]</code>
<code>print list(range(5, 0, -1))</code>	<code>[5, 4, 3, 2, 1]</code>
<code>print list(range(10, 0, -2))</code>	<code>[10, 8, 6, 4, 2]</code>
<code>print list(range(10, 0, -3))</code>	<code>[10, 7, 4, 1]</code>



7.2 for문 - 따라 해보기

- 1부터 10까지 정수의 합 구하기
 - $1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 + 10$ 의 합 구하기

[소스코드] 7-7.py

```
sum = 0
for n in range(11):
    sum = sum+n
print("The sum=" + str(sum))
```

[실행결과]

The sum=55

- 1부터 10까지 정수 중 홀수만의 합 구하기
 - $1 + 3 + 5 + 7 + 9$ 의 합 구하기

[소스코드] 7-8.py

```
sum = 0
for n in range(1,11,2):
    sum = sum + n
print("The sum=" + str(sum))
```

[실행결과]

The sum=25



7.2 따라 해보기 - 정 다각형 그리기

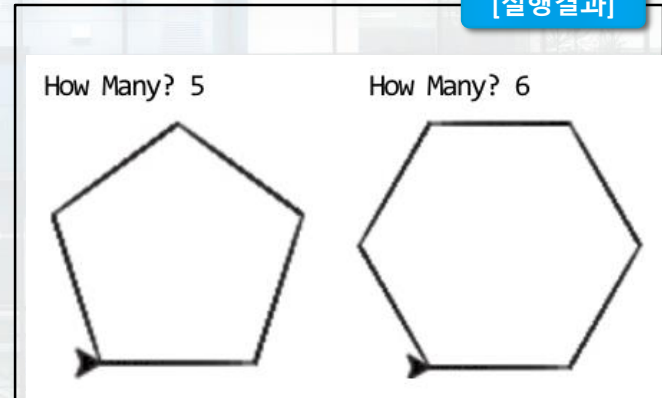
- 터틀 그래픽스와 for문을 사용하여 정다각형 그리는 프로그램
- 사용자로부터 값을 입력 받아서 그리기

[소스코드] 7-9.py

```
import turtle
t=turtle.Turtle()
n = int(input("How Many? "))
for m in range(n) :
    t.forward(60)
    t.left(360/n)
```

회전 각도

[실행결과]





따라 해보기 - 추가 실습

- 입력 : 번호 순서대로 10명의 학생 점수 입력
- 조건 : 점수가 70점 이상이면 합격/70점 미만이면 불합격
- 출력 : 학생번호와 점수,합격 여부 출력

```
score_list = [70, 55, 67, 45, 80, 88, 90, 100, 70, 70]
```

```
number = 0
```

```
for score in score_list:
```

```
    number = number + 1
```

```
    if score >= 70:
```

```
        print("%d번 학생은 %d점으로 합격입니다." % (number,score))
```

```
    else:
```

```
        print("%d번 학생은 %d점으로 불합격입니다." % (number,score))
```

1번 학생은 70점으로 합격입니다.
2번 학생은 55점으로 불합격입니다.
3번 학생은 67점으로 불합격입니다.
4번 학생은 45점으로 불합격입니다.
5번 학생은 80점으로 합격입니다.
6번 학생은 88점으로 합격입니다.
7번 학생은 90점으로 합격입니다.
8번 학생은 100점으로 합격입니다.
9번 학생은 70점으로 합격입니다.
10번 학생은 70점으로 합격입니다.



마무리

- 이번 수업에서는 반복문에 대한 이해와 반복문 중 for문에 대해 알아보았습니다.