

LAB 05

2020-01 소프트웨어의 이해 01분반 / 조교 이경민

LAB 05

- 반복문
 - while
 - for
- break/continue
- 중첩 루프
- 예제
- 실습 과제

반복문 - while 문

- while 문
 - 조건을 만족하는 동안 (= 조건식이 참인 동안) 문장 1을 반복하여 실행
 - 1. 조건을 만족하는지 확인 → 맞는 경우, 문장 1 실행
 - 2. 다시 조건식 확인 → 맞는 경우, 문장 1 실행
 - ...
 - n. 조건을 만족하지 않는 경우, while문 빠져 나옴
- 만약 처음 조건 확인 시, 거짓일 경우, while문 내 문장이 한 번도 실행되지 않을 수 있음!

while 조건식 :
문장 1

예제 1) 화씨 - 섭씨 변환 표 출력

- 화씨 온도 0도부터 100도까지 매 10도마다 화씨 온도를 섭씨 온도로 변환하여 출력
- 변환 공식

$$^{\circ}\text{C} = (^{\circ}\text{F} - 32) \times 5 / 9$$

$$^{\circ}\text{F} = (^{\circ}\text{C} \times 9 / 5) + 32$$

```
f = 0

while f <= 100 :
    c = (f-32) * 5 / 9
    print(f, "F = %.1f" % c, "C")
    f = f + 10
```

→ 소수점 이하
한 자리

```
=====
===
0 F = -17.8 C
10 F = -12.2 C
20 F = -6.7 C
30 F = -1.1 C
40 F = 4.4 C
50 F = 10.0 C
60 F = 15.6 C
70 F = 21.1 C
80 F = 26.7 C
90 F = 32.2 C
100 F = 37.8 C
```

예제 2) 온도 변환 메뉴 프로그램

- 온도 변환 메뉴를 보여주고, 메뉴 선택에 따라
 - 1. 화씨-섭씨 변환 2. 섭씨-화씨 온도 변환을 수행하여 결과를 출력

온도 변환 메뉴에 따른 화씨 온도-섭씨 온도 변환 프로그램

```
print("* 온도 변환 메뉴 *")
print("1. 화씨-섭씨 변환")
print("2. 섭씨-화씨 변환")
print("3. 종료")
```

```
menu = 0
```

```
while menu != 3:
```

```
    menu = int(input("메뉴를 선택하세요 : "))
```

```
    if menu == 1:
```

```
        fahr = float(input("화씨 온도 입력: "))
```

```
        celsius = 5 * (fahr - 32) / 9
```

```
        print("섭씨 온도: %.1f" % celsius)
```

```
    elif menu == 2:
```

```
        celsius = float(input("섭씨 온도 입력: "))
```

```
        fahr = celsius * 9/5 + 32
```

```
        print("화씨 온도: %.1f" % fahr)
```

```
    elif menu == 3:
```

```
        print("종료합니다")
```

```
    else:
```

```
        print("잘못된 메뉴입니다")
```

메뉴이름
패턴

```
=====
```

```
=====
```

```
* 온도 변환 메뉴 *
```

```
1. 화씨-섭씨 변환
```

```
2. 섭씨-화씨 변환
```

```
3. 종료
```

```
메뉴를 선택하세요 : 1
```

```
화씨 온도 입력: 95
```

```
섭씨 온도: 35.0
```

```
메뉴를 선택하세요 : 2
```

```
섭씨 온도 입력: 12
```

```
화씨 온도: 53.6
```

```
메뉴를 선택하세요 : 3
```

```
종료합니다
```

```
>>> |
```

반복문 - for 문 *하나씩 꺼내면서 반복할 때 많이 사용*

- 순서형 자료의 이용
 - 순서형 자료 : 문자열, 리스트, 튜플, 사전, 파일 등 - 서열이나 순위를 나타내는 자료
- 각 원소에 대해 순차적으로 문장 1을 반복
- 루프 제어 변수
 - 순서형 자료에 있는 각 원소가 하나씩 대입
 - 각 원소에 대해 순차적으로 문장 1을 반복


for 변수 in 순서형 자료 :
문장 1

반복문 - for 문

- 문자열

- 변수 x : 문자열에 있는 각 문자가 대입됨.

```
>>> for x in "Hello" :  
        print(x)  
  
H  
e  
l  
l  
o  
>>> |
```




- 리스트

- 변수 x : 리스트의 원소인 각 정수가 대입됨.

list element를
하나씩 꺼내면서.

```
>>> for x in [1, 2, 3, 4, 5] :  
        print(x)  
  
1  
2  
3  
4  
5  
>>> |
```



반복문 - for 문

파일

- 여러 줄로 이루어진 일종의 순서형 자료
- for문을 이용하여 각 줄에 대해서 반복

각 줄마다
읽어야 함
처리를 해야 함

```
total = 0
count = 0
file = open("scores.txt")
for line in file :
    score = int(line)
    total += score
    count += 1
    print(score)
print("총점 : ", total)
print("평균 : %.2f" % (total/count))
```

input
output의
증거와 같은 것
문자로 이루어져 있다.

scores - 메모장

파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)

40
93
28
50
67
90

EOF (end of file)
까지 반복

```
=====
==
40
93
28
50
67
90
총점 : 368
평균 : 61.33
>>>
```


반복문 - for 문

- 정수 범위의 이용

- range 함수

- `range(n)` : 0 부터 $n-1$ 까지의 정수 범위 (n번의반복)
- `range(n, m)` : n 부터 $m-1$ 까지의 정수 범위
- `range(n, m, k)` : n 부터 시작해서 k 만큼씩 증가하여 $m-1$ 까지의 정수들로 이루어진 범위
- `range` 함수를 사용한 결과를 확인하는 법 → `list()` 함수 사용

```
>>> list(range(10))
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
>>> list(range(1,10))
[1, 2, 3, 4, 5, 6, 7, 8, 9]
>>> list(range(1,10,2))
[1, 3, 5, 7, 9]
>>> list(range(10,0,-1))
[10, 9, 8, 7, 6, 5, 4, 3, 2, 1]
>>>
```

↑ 결과값은
리스트로 온다

→ 증가분이 0 인 경우
countdown

예제 3) 적금 프로그램

- 2개월마다 5만원씩 36개월 적금하는 프로그램

2개월마다 5만원씩 36개월 적금하는 프로그램

```
balance = 0
```

```
for month in range(1,36,2):
```

```
    balance = balance + 50000
```

```
    print(str(month)+"개월 입금, 현재 금액:"+str(balance)+"원")
```

```
print("총 적금액:", balance)
```

→ 1~35까지
2달마다 5만원씩

점진적으로~

for문 → while문
바꿀 줄 알아야 함

1개월 입금, 현재 금액:50000원
3개월 입금, 현재 금액:100000원
5개월 입금, 현재 금액:150000원
7개월 입금, 현재 금액:200000원
9개월 입금, 현재 금액:250000원
11개월 입금, 현재 금액:300000원
13개월 입금, 현재 금액:350000원
15개월 입금, 현재 금액:400000원
17개월 입금, 현재 금액:450000원
19개월 입금, 현재 금액:500000원
21개월 입금, 현재 금액:550000원
23개월 입금, 현재 금액:600000원
25개월 입금, 현재 금액:650000원
27개월 입금, 현재 금액:700000원
29개월 입금, 현재 금액:750000원
31개월 입금, 현재 금액:800000원
33개월 입금, 현재 금액:850000원
35개월 입금, 현재 금액:900000원
총 적금액: 900000

예제 3) 적금 프로그램

- 2개월마다 5만원씩 36개월 적금하는 프로그램

```
balance = 0
month = 1

while month < 36 :
    balance = balance + 50000
    print(str(month)+"개월 입금, 현재 금액: "+str(balance))
    month += 2

print("총 적금액:", balance)
```

✓ loop 바디 내에서
변수값이 변하지 않도록
하는 loop 구조

break/continue 문

- break

완전히 loop body에서 빠져나옴

- for나 while 루프 내에서 어떤 조건이 만족되면 중간에 루프에서 빠져 나오는데 사용

- continue

continue문 뒤를 찍어내고 반복의 시작부터.

- for나 while 루프 내에서 현재 반복을 중단하고 다음 반복으로 건너뛰는데 사용

```
for i in range(1, 21) :  
    if i%7 == 0 :  
        continue  
    elif i == 16 :  
        break  
    print(i)
```

1
2
3
4
5
6
8
9
10
11
12
13
15

중첩 루프

- 구구단 출력 프로그램

```
for i in range(2, 10):  
    print("==%d 단==" % i)  
    for j in range(1, 10):  
        print("%d X %d = %d" % (i, j, i*j))  
    print()
```

```
i=2  
i=2  
  
while i <= 9:  
    print("==%d 단==" % i)  
    while j <= 9:  
        print("%d X %d = %d" % (i, j, i*j))  
        j += 1  
    print()  
    i += 1  
i+=1
```

==8 단==

8 X 1 = 8
8 X 2 = 16
8 X 3 = 24
8 X 4 = 32
8 X 5 = 40
8 X 6 = 48
8 X 7 = 56
8 X 8 = 64
8 X 9 = 72

==9 단==

9 X 1 = 9
9 X 2 = 18
9 X 3 = 27
9 X 4 = 36
9 X 5 = 45
9 X 6 = 54
9 X 7 = 63
9 X 8 = 72
9 X 9 = 81

예제 4) 학생 별 점수 합 계산

- 학생 별 국영수 점수 합을 계산하는 프로그램

학생별 국영수 점수 합을 계산하는 프로그램

```
scores = [[80,90,85], [70,80,80], [90,90,95], [60,70,65], [90,80,75]]
stud = []

for lst in scores:
    sum = 0
    for score in lst:
        sum += score
    stud.append(sum)
print(stud)
```

다인코퍼하기

```
=====
==
[255, 230, 275, 195, 245]
```

LAB05 실습과제 1

- 사용자로부터 숫자를 입력 받아 그 수 사이에서 2의 배수이면서 3의 배수인 수를 제외하고 출력하는 프로그램
- for문과 while문 각각을 사용해서 총 2개의 파일로 제출
- for문을 사용한 프로그램 : **multiplication_for.py**
- while문을 사용한 프로그램 : **multiplication_while.py**

Loop
(
 조건: 2의 배수(%.2==0) and 3의 배수(%.3==0)
 print("오류")
 print(i)

```
===== RESTA
숫자를 입력해주세요 : 10
1
2
3
4
5
7
8
9
10
>>> |
```

```
숫자를 입력해주세요 : 20
1
2
3
4
5
7
8
9
10
11
13
14
15
16
17
19
20
>>> |
```

LAB05 실습과제 2 (coffee.py)

- 메뉴를 보여주고, 메뉴에 따라 주문을 입력 받고,
주문한 메뉴리스트와 지불할 총액을 출력하는 프로그램 작성

- [조건] *입력받아서*

✓ 계속 주문할 경우 입력하는 Y와 N이 대문자, 소문자 모두 가능하도록 설정

- 잘못된 메뉴를 선택 시, 안내 메시지를 출력하고 메뉴를 다시 입력 받을 것

✓ 주문 완료 후, 주문내역에 주문한 메뉴와 수량 출력

✓ 단, 주문하지 않은 메뉴는 주문 내역에 프린트 되면 안됨

✓ 메뉴판을 반복해서 출력할 필요는 없음

- 다음 페이지 실행화면 참고

*null list에 추가하는
방법이 있음.*

LAB05 실습과제 2 (coffee.py)

- 실행화면

```
=====Sookmyung Cafe=====
1. 아메리카노 1800원
2. 카페라떼 2200원
3. 카페모카 2800원
=====

메뉴를 선택해주세요 >>> |
```

```
=====Sookmyung Cafe=====
1. 아메리카노 1800원
2. 카페라떼 2200원
3. 카페모카 2800원
=====

메뉴를 선택해주세요 >>> 3
몇 잔을 주문하시겠습니까? >>> 2
주문을 계속하시겠습니까? (Y/N) >>> Y

메뉴를 선택해주세요 >>> 2
몇 잔을 주문하시겠습니까? >>> 1
주문을 계속하시겠습니까? (Y/N) >>> y

메뉴를 선택해주세요 >>> 1
몇 잔을 주문하시겠습니까? >>> 5
주문을 계속하시겠습니까? (Y/N) >>> N

=====주 문 내 역=====
카페모카 2 잔
카페라떼 1 잔
아메리카노 5 잔

==총 8 잔, 16800 원==

이용해주셔서 감사합니다
>>> |
```

LAB05 실습과제 2 (coffee.py)

- 실행화면

```
=====Sookmyung Cafe=====
      1. 아메리카노 1800원
      2. 카페라떼 2200원
      3. 카페모카 2800원
=====

메뉴를 선택해주세요 >>> 5
메뉴를 다시 선택해주세요
메뉴를 선택해주세요 >>> 2
몇 잔을 주문하시겠습니까? >>> 3
주문을 계속하시겠습니까? (Y/N) >>> n

=====주 문 내 역=====
      카페라떼 3 잔

==총 3 잔, 6600 원==

이용해주셔서 감사합니다
>>> |
```

과제 채점 기준·기한

- 과제 제출 기한

- 실습 다음주 화요일 오후 11시까지
- 5월 5일 화요일 오후 11시까지 제출

- 제출 장소

- 스노우보드 해당 주차 과제 제출 페이지에 업로드

- 추가 제출

- 제출기한 이후 24시간 이내 업로드 : 만점에서 20% 감점
- 그 이후는 받지 않음

- 표절 X

과제 제출 형식 & 질문 메일

- **제출물** : 소스파일(.py)과 과제보고서(.docx) 합친 **압축파일 제출**
 - **소스파일 이름** : 매 실습 과제 마다 ppt에 제시 예정
 - ex) test.py
 - **과제 보고서 양식** : 스노우보드에서 다운로드
- **(소스파일 + 과제보고서) 압축 파일 이름 : Lab05_본인학번_이름**
 - ex) Lab05_1911123_이경민.zip
- **이메일** : newkml22@gmail.com
- **질문 시 주의사항**
 - **과목, 분반, 이름, 학번** 알려주세요.
 - 몇 번 과제에서, 어떤 부분이 막혔는지, 어떤 과정인지 **설명**과 함께 보내주세요.
 - 답장까지 시간이 걸릴 수도 있으니 제출 과제 질문은 미리 해주세요!