

---

# 다중 for() 활용

5주차\_03

한 동 대 학 교  
김 경 미 교수

# 다중 for문

- **for() 문 안에 for()이 사용되는 것**
  - 아래 예제와 같이 활용

```
# 리스트 처리
```

```
fruit = ["apple", "banana", "lemon", "tomato"]
```

```
for fr in fruit :
```

```
    count = 0
```

```
    for f in fr :
```

```
        print(f)
```

```
        count = count + 1
```

```
    print("아이템 수 = ", count)
```

```
>>>
===== RESTART :
==
a
p
p
l
e
아이템 수 = 5
b
a
n
a
n
a
a
아이템 수 = 6
|
e
m
o
n
아이템 수 = 5
t
o
m
a
t
o
아이템 수 = 6
>>> |
```

# 다중 for() 문 실행 과정

	변수 fr	변수 f	변수 fr	변수 f
1번 for	"apple"		1번 for	"lemon"
	2번 for	"a"	2번 for	"l"
		"p"		"e"
		"p"		"m"
		"l"		"o"
		"e"		"n"
1번 for	"banana"		1번 for	"tomato"
	2번 for	"b"	2번 for	"t"
		"a"		"o"
		"n"		"m"
		"a"		"a"
		"n"		"t"
		"a"		"o"

# 중첩 for문(nested for)

---

- for문 여러 개 나열

```
for i in range(3) :  
    for j in range(3) :  
        print ("  $")
```

```
>>>  
$  
$  
$  
$  
$  
$  
$  
$  
$  
>>> |
```

```
for i in range(2) :  
    print("#")  
    for j in range(5) :  
        print ("  $")
```

```
>>>  
#  
$  
$  
$  
$  
$  
$  
#  
$  
$  
$  
$  
$  
$  
>>> |
```

# 중첩 for문 예제

```
for i in range(2,7,2) :  
    print ("1st level = ", i)
```

```
    for j in range(1,5,1) :  
        print(" 2nd level =  
        print(" i * j = ", i*j)
```

```
>>>  
===== RESTART:  
=====  
1st level = 2  
 2nd level = 1  
   i * j = 2  
 2nd level = 2  
   i * j = 4  
 2nd level = 3  
   i * j = 6  
 2nd level = 4  
   i * j = 8  
1st level = 4  
 2nd level = 1  
   i * j = 4  
 2nd level = 2  
   i * j = 8  
 2nd level = 3  
   i * j = 12  
 2nd level = 4  
   i * j = 16  
1st level = 6  
 2nd level = 1  
   i * j = 6  
 2nd level = 2  
   i * j = 12  
 2nd level = 3  
   i * j = 18  
 2nd level = 4  
   i * j = 24  
>>>
```

```
# times table : 7, 8
```

```
for i in range(7, 9) :  
    print (i, "th table")
```

```
    for j in range(1,10,1) :  
        print( " ", i, " * ", j, " = ", i*j)
```

```
>>>  
===== RESTART:  
=====  
7 th table  
 7 * 1 = 7  
 7 * 2 = 14  
 7 * 3 = 21  
 7 * 4 = 28  
 7 * 5 = 35  
 7 * 6 = 42  
 7 * 7 = 49  
 7 * 8 = 56  
 7 * 9 = 63  
8 th table  
 8 * 1 = 8  
 8 * 2 = 16  
 8 * 3 = 24  
 8 * 4 = 32  
 8 * 5 = 40  
 8 * 6 = 48  
 8 * 7 = 56  
 8 * 8 = 64  
 8 * 9 = 72
```

# 무엇이 출력 될까?

---

```
# a의 값은 무엇인가?
```

```
a=0
```

```
for i in range(10) :
```

```
    a=a+1
```

```
for j in range(10) :
```

```
    a=a+1
```

```
print(a)
```

# 무엇이 출력 될까?, 답안

---

# a의 값은 무엇인가?

a=0

for i in range(10) :

    a=a+1

for j in range(10) :

    a=a+1

print(a)

```
>>>
===== |
====
20
>>> |
```

# 무엇이 출력 될까?

---

```
# a의 값은 무엇인가?  
a=0  
for i in range(10) :  
    a=a+1  
    for j in range(10) :  
        a=a+1  
  
print(a)
```



# 무엇이 출력 될까?, 답안

---

```
# a의 값은 무엇인가?  
a=0  
for i in range(10) :  
    a=a+1  
    for j in range(10) :  
        a=a+1  
  
print(a)
```

```
>>>  
===== RI  
=====  
110  
>>>
```

# 무엇이 출력 될까?

---

# 다음은 무엇이 출력되는가?

```
fruit = "banana"
count = 0
for char in fruit :
    if char == 'a' :
        count = count + 1

print(count)
```

# 무엇이 출력 될까?, 답안

---

# 다음은 무엇이 출력되는가?

```
fruit = "banana"
count = 0
for char in fruit :
    if char == 'a' :
        count = count + 1

print(count)
```

```
>>>
===== R
=====
3
>>>
```

# break문

---

- 반복문에서 벗어나도록 중단시킴
  - 실행 흐름이 루프 이후, 첫번째 명령문으로 넘어감

```
fruit = "banana"
count = 0

for char in fruit :
    if char == 'a':
        count = count + 1
    elif char == 'n':
        break
    else :
        print(char)

print(count)
```

```
>>>
===== |
b
1
>>>
```

# break문 예제

- Nested for문에서 사용

#2개 수를 더한 결과가 5의 배수 이면 반복문 종료

```
count = 0
```

```
for i in range(5):  
    for j in range(3):  
        if (i + j) % 5 != 0:  
            count = count + 1  
            print("i + j =", i+j, "and count =", count)  
        else:  
            break
```

```
print("i + j = ", i+j, "last count: ", count)
```

```
>>>  
===== RESTART: E:/1_Work  
i + j = 1 and count = 1  
i + j = 2 and count = 2  
i + j = 3 and count = 3  
i + j = 2 and count = 4  
i + j = 3 and count = 5  
i + j = 4 and count = 6  
i + j = 3 and count = 7  
i + j = 4 and count = 8  
i + j = 4 and count = 9  
i + j = 5 last count: 9  
>>>
```

# Prime Number 확인 과정

---

- 소수의 정의

- 양의 정수 중 1과 자기자신으로만 나누어 지는 수

- 확인 과정

- 2부터 시작해서 입력 받은 수를 나누기 시작하여,
- (입력 받은 수 -1) 까지 나누는 것을 반복한다
- 그 과정에서 나누어지는 경우가 발생하면, 소수가 아니고
- 끝까지 나누어지지 않고 반복문이 끝나면 소수이다
- 나누어지면, 소수를 확인하는 flag 변수의 값을 False로 처리한다

# Prime Number 확인

---

```
# prime number 여부 확인

num = int(input("input a positive integer : "))

prime_yes = True

for i in range(2, num) :
    if num % i == 0:
        prime_yes = False
        break

if prime_yes == True :
    print(num, "is a prime number")
else :
    print(num, "is not a prime number")
```

```
>>>
===== RESTART: E:/1_Works/20
input a positive integer : 12
12 is not a prime number
>>>
===== RESTART: E:/1_Works/20
input a positive integer : 11
11 is a prime number
>>> |
```

# 연습문제 1

---

- 영어단어를 입력한다
- 위의 단어에 들어가는 글자 중 하나를 입력한다
- 몇 번째 위치에 그 글자가 있는지 찾는다
- 존재하지 않는 글자인 경우, 없다고 출력한다



# 연습문제 1, 코드와 결과

```
word = input("단어를 입력하세요: ")
letter = input("위치를 찾을 한 글자를 입력하세요: ")

count = 0

for char in word :
    count = count + 1
    if char == letter:
        break

if count == len(word) :
    print( letter, "는", word, "내에 없습니다")
else :
    print(letter, "는", word, "의", count, "번째 위치합니다")
```

```
>>>
===== RESTART: E:/1_Works/2017Work/t
단어를 입력하세요: JavaScript
위치를 찾을 한 글자를 입력하세요: S
S 는 JavaScript 의 5 번째 위치합니다
>>>
===== RESTART: E:/1_Works/2017Work/t
단어를 입력하세요: JavaScript
위치를 찾을 한 글자를 입력하세요: s
s 는 JavaScript 내에 없습니다
>>>
```

## 연습문제 2

---

- 사용자에게 메시지를 입력 받는다
- 메시지 내에서 'a' 를 개수를 확인하여 출력한다

# 연습문제 2, 코드와 결과

---

- 문자열에서 한 글자씩 읽어서 그 글자가 'a' 인지를 확인하여 센다

```
s = input("Input message : ")
count = 0

for c in s :
    if c == "a" :
        count += 1

print("count of 'a' : ", count)
```

```
>>> =====
>>>
Input message : apple and banana
count of 'a' : 5
>>>
```

## 연습문제 3

---

- 구구단 출력
- 2단부터 시작
- 몇 단까지 출력할 지는 사용자에게 입력 받는다

# 연습문제 3, 코드와 결과

```
num = int(input("구구단, 몇 단까지 입력 하실래요? : "))

for i in range(2, num+1, 1):
    print("="*20)
    for j in range(2, 10, 1):
        print(i, "*", j, " = ", i * j)
```

```
>>>
===== RESTART: E:/1_Works/2017Wo
구구단, 몇 단까지 입력하실래요? : 4
=====
2 * 2 = 4
2 * 3 = 6
2 * 4 = 8
2 * 5 = 10
2 * 6 = 12
2 * 7 = 14
2 * 8 = 16
2 * 9 = 18
=====
3 * 2 = 6
3 * 3 = 9
3 * 4 = 12
3 * 5 = 15
3 * 6 = 18
3 * 7 = 21
3 * 8 = 24
3 * 9 = 27
=====
4 * 2 = 8
4 * 3 = 12
4 * 4 = 16
4 * 5 = 20
4 * 6 = 24
4 * 7 = 28
4 * 8 = 32
4 * 9 = 36
>>>
```

# 숙제

---

- 연습문제 2, 3 코드와
- 실행결과 캡처 한 사진을 게시판에 올려주세요!

# 요약

---

- 다중 for()문 이해하기
- break 이해하기
- 다중 for()문 활용 연습하기

---

# 감사합니다

5주차\_03 다중 for() 활용