05.반복문

반복문 이란

- 로직을 반복하여 수행한다.
  . 1 부터 100 까지 숫자를 출력 하는 문제가 주어졌을경우
  > 방법 1 : 결과 를 도출 할때까지 모두 작성한다.

```
Chap01_intro > 05.Loop[반복문].py
1  print(1)
2  print(2)
3  print(3)
4  print(4)
5  print(5)
6  print(6)
7  # .......
8  print(100)
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Programs/P
1
2
3
4
5
6
100
```

> 방법 2 : 반복적으로 수행하는 로직을 찾아 반복문을 사용한다. (range() 함수 를 사용)
  * range(1,101) : 1 ~ 100
  * range(101) : 0 ~ 100

```
94  # # range(1,101) : 1 ~ 100
95  # # range(101) : 0 ~ 100
96  for number in range(1,101) :
97      print(number,',',end='')
98  print('\n----------------------------------------')
99  for number in range(101) :
100     print(number,',',end='')
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS                Python Debug Console

Microsoft Windows [Version 10.0.22621.2283]
(c) Microsoft Corporation. All rights reserved.

D:\MyDoc\11.EDUCATION\1. Master Source\15.Python\0.MyFirstPython\0.Source\Python> cmd /C "C:\Users\MasterD\AppData\Local\Programs\Python\Python311\python.exe c:\Users\MasterD\.v
scode\extensions\ms-python.python-2023.16.0\pythonFiles\lib\python\debugpy\adapter/../..\debugpy\launcher 53322 -- "D:\MyDoc\11.EDUCATION\1. Master Source\15.Python\0.MyFirstPyt
hon\0.Source\Python\05.Loop[반복문].py" "
6561
1 ,2 ,3 ,4 ,5 ,6 ,7 ,8 ,9 ,10 ,11 ,12 ,13 ,14 ,15 ,16 ,17 ,18 ,19 ,20 ,21 ,22 ,23 ,24 ,25 ,26 ,27 ,28 ,29 ,30 ,31 ,32 ,33 ,34 ,35 ,36 ,37 ,38 ,39 ,40 ,41 ,42 ,43 ,44 ,45 ,46 ,47
 ,48 ,49 ,50 ,51 ,52 ,53 ,54 ,55 ,56 ,57 ,58 ,59 ,60 ,61 ,62 ,63 ,64 ,65 ,66 ,67 ,68 ,69 ,70 ,71 ,72 ,73 ,74 ,75 ,76 ,77 ,78 ,79 ,80 ,81 ,82 ,83 ,84 ,85 ,86 ,87 ,88 ,89 ,90 ,91
,92 ,93 ,94 ,95 ,96 ,97 ,98 ,99 ,100 ,
----------------------------------------
0 ,1 ,2 ,3 ,4 ,5 ,6 ,7 ,8 ,9 ,10 ,11 ,12 ,13 ,14 ,15 ,16 ,17 ,18 ,19 ,20 ,21 ,22 ,23 ,24 ,25 ,26 ,27 ,28 ,29 ,30 ,31 ,32 ,33 ,34 ,35 ,36 ,37 ,38 ,39 ,40 ,41 ,42 ,43 ,44 ,45 ,46
,47 ,48 ,49 ,50 ,51 ,52 ,53 ,54 ,55 ,56 ,57 ,58 ,59 ,60 ,61 ,62 ,63 ,64 ,65 ,66 ,67 ,68 ,69 ,70 ,71 ,72 ,73 ,74 ,75 ,76 ,77 ,78 ,79 ,80 ,81 ,82 ,83 ,84 ,85 ,86 ,87 ,88 ,89 ,90 ,
91 ,92 ,93 ,94 ,95 ,96 ,97 ,98 ,99 ,100 ,
```

반복문 FOR

- C#,JAVA 의 foreach 와 같은 동작원리.
- 반복 범위의 설정

```
for number in range(1,100) :
```

> for [하나씩 뽑아올 변수] in range([시작], [끝])
  * range 는 이상 , 미만 을 나태므로 1부터 100까지 숫자는 rang(1,101) 이 되어야 한다.

```
Chap01_intro > 05.Loop[반복문].py > ...
1  for number in range(1,101) :
2      print(number,',',end='')
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL                Python

Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/Python311/python.exe d:/Python/Chap01_intro/05.Loop[반복문].py
1 ,2 ,3 ,4 ,5 ,6 ,7 ,8 ,9 ,10 ,11 ,12 ,13 ,14 ,15 ,16 ,17 ,18 ,19 ,20 ,21 ,22 ,23 ,24 ,25 ,26 ,27 ,28 ,29 ,30 ,31 ,32 ,33 ,34 ,35 ,36 ,37 ,38 ,39
,40 ,41 ,42 ,43 ,44 ,45 ,46 ,47 ,48 ,49 ,50 ,51 ,52 ,53 ,54 ,55 ,56 ,57 ,58 ,59 ,60 ,61 ,62 ,63 ,64 ,65 ,66 ,67 ,68 ,69 ,70 ,71 ,72 ,73 ,74 ,75 ,7
6 ,77 ,78 ,79 ,80 ,81 ,82 ,83 ,84 ,85 ,86 ,87 ,88 ,89 ,90 ,91 ,92 ,93 ,94 ,95 ,96 ,97 ,98 ,99 ,100 ,
```

. 범위 의 설정 2

> for [하나씩 뽑아올 변수] in range([**시작**], [**끝**],[**증가값**])

* 1부터 2단계 씩 띄워서 표현한다. (홀수 만 표현한다)

1, 2, 3, 4, 5, 6, 7, 8, 9, ... 100

```
for number in range(1,101,2) :
    print(number,',',end='')
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/Python311/python.exe d:/Python/Chap01_intro/05.Loop[반복문].py
1 ,3 ,5 ,7 ,9 ,11 ,13 ,15 ,17 ,19 ,21 ,23 ,25 ,27 ,29 ,31 ,33 ,35 ,37 ,39 ,41 ,43 ,45 ,47 ,49 ,51 ,53 ,55 ,57 ,59 ,61 ,63 ,65 ,67 ,69 ,71 ,73 ,75
,77 ,79 ,81 ,83 ,85 ,87 ,89 ,91 ,93 ,95 ,97 ,99 ,
```

- 리스트 의 데이터 를 추출하기

. 리스트 의 데이터 수 만큼 순차적으로 추출하여 로직을 반복한다.

. for [하나씩 뽑아올 변수] in [1,2,3,4,5]

> 범위

```
for number in [1,2,3,4,5] :
    print(number,',',end='')
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Programs/Pytho
1 ,2 ,3 ,4 ,5 ,
```

* 범위 로 표현한 데이터의 정렬 순서는 관계 없다.

```
for number in [1,3,7,9,2] :
    print(number,',',end='')
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Programs/Pytho
1 ,3 ,7 ,9 ,2 ,
```

실습

1. 1 부터 100 사이의 수 중 짝수 만 표현하는 로직을 구현해 보세요

```
D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/
2 ,4 ,6 ,8 ,10 ,12 ,14 ,16 ,18 ,20 ,22 ,24 ,26 ,28 ,30 ,3
,78 ,80 ,82 ,84 ,86 ,88 ,90 ,92 ,94 ,96 ,98 ,100 ,
```

2. 1부터 100 까지의 수 중 3 의 배수 인 숫자의 모든 합을 구하세요

* 외부 변수 의 응용

```
Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Program
1683
```

- 반복문 강제 종료

. **Break**

> 특정 조건일 경우 실행하고 있는 반복문을 강제로 종료 할 수 있다.

* 현재 처리할 수 가 77 인경우 반복문을 종료한다.
* 그동안 누적하였던 sum 의 결과는 표현된다.

```
sum = 0
for number in range(0,101) :
    if number % 3 == 0 :
        sum += number
    elif number == 77 :
        break
print(sum)
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/
975
```

< 1,2,3,4,5, ... ,77 break >

- 다음 순번의 내용으로 반복문 돌리기

. **Continue**

> 조건을 만족 시켰을 경우 반복문을 종료하지 않고 다음 순차 로 로직을 처리 하도록 한다.

* 현재 루프의 처리 번호 가 30 과 40 사이 인경우 다음 차순으로 처리 되도록 처리

```
Chap01_intro > 05.Loop[반복문].py > ...
1   sum = 0
2   for number in range(0,101) :
3       if 30 < number < 40 :
4           continue
5       elif number % 3 == 0 :
6           sum += number
7       elif number == 77 :
8           break
9   print(sum)
10
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL
Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/Python
867
```

< 1,2,3,4,5...30, (Continue ...) ,40, 41, 42, 43 .... 77 break >

이중 루프

- 반복 문 안에 반복문을 구현

. 구구단 만들기

> 단수 를 2단부터 9단 까지 돌리는 dan 루프 문 과
> 각 단수 별 1 부터 9 까지의 곱을 나타내는 hang 의 이중 루프를 통하여 구구단을 표현한다.

```
for dan in range(2,10):
    print('\n',dan , ' 단 입니다.')
    for hang in range(1,10) :
        print(dan, ' * ', hang , ' = ', dan*hang)
```

```
 2 단 입니다.
2 * 1 = 2
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 단 입니다.
3 * 1 = 3
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 단 입니다.
4 * 1 = 4
4 * 2 = 8
4 * 3 = 12
4 * 4 = 16
4 * 5 = 20
4 * 6 = 24
4 * 7 = 28
4 * 8 = 32
4 * 9 = 36
```

< 실행 결과 >

. triangle 표현하기

> 이중 루프 를 이용한 * 삼각형 표현하기

```
for y in range(1,10) :
    for x in range(y):
        print('*',end='')
    print()
```

```
*
**
***
****
*****
******
*******
********
*********
```

< 실행 결과 >

* 행의 순번을 가지고있는 y 에 1~9 행을 처리 하도록 하고 행 별로 * 을 반복적으로 표현하기 위하여 이중 루프 로직 사용
이때 행 의 수와 * 의 개수 가 같으므로 `for x in range(y):` 로직이 가능해 진다.

```
Chap01_intro > 05.Loop[반복문].py > ...
1   for y in range(1,9):
2       print('*' * y)
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Program
*
**
***
****
*****
******
*******
********
```

>2행 : 문자열 곱하기 문법을 응용 하여 간단하게 표현 할 수 있다.

While

- 반복문 While

. 특정 조건을 만족 시킬 때 로직을 수행한다.

> 증가 한 번호 가 5 이하 일 경우 만 로직 을 실행한다.

```
Chap01_intro > 05.Loop[반복문].py > ...
1  student = 1
2  while student <= 5 :
3      print(student, ' 번 학생입니다.')
4      student += 1
5
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/Python311/pyt
1  번 학생입니다.
2  번 학생입니다.
3  번 학생입니다.
4  번 학생입니다.
5  번 학생입니다.
```

>1행 : student 에 초기값 할당(초기화)
>2행 : student 의 값이 5 이하이면 True
>3행 : 문자열 출력
>4행 : student 의 값 1 증가.

> While 에도 Break 와 Continue 문법이 적용 된다.

```
Chap01_intro > 05.Loop[반복문].py > ...
1  student = 1
2  while student <= 20 :
3      print(student, ' 번 학생입니다.')
4      student += 1
5      if  10 < student < 15:
6          continue
7      elif student == 18 :
8          break
9
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/Python311/pyth
1  번 학생입니다.
2  번 학생입니다.
3  번 학생입니다.
4  번 학생입니다.
5  번 학생입니다.
6  번 학생입니다.
7  번 학생입니다.
8  번 학생입니다.
9  번 학생입니다.
10  번 학생입니다.
11  번 학생입니다.
12  번 학생입니다.
13  번 학생입니다.
14  번 학생입니다.
15  번 학생입니다.
16  번 학생입니다.
17  번 학생입니다.
```

>1행 : student 에 초기값 할당(초기화)
>2행 : student 의 값이 5 이하이면 True
>3행 : 문자열 출력
>4행 : student 의 값 1 증가.
>5행 : student 값이 11 ~ 14 일 경우 다음 반복문 실행
>6행 : student 값이 18 일 경우 로직 종료

실습

11 번 부터 14 번 학생은 출력이 되지 않도록 하였으나 출력이 된 이유를 확인하고 원하는 결과 를 출력 할 수 있도록 수정 해 보세요

반복문 While

- 특정한 조건 없이는 끝나지 않는 반복문
  . 프로그램이 구동되고 있는동안 현재 시간을 지속하여 표현 하는 로직의 예제

```
Chap01_intro > 05.Loop[반복문].py
1  from datetime import datetime
2  import time
3
4  while True :
5      print(datetime.now())
6      time.sleep(1)
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL
Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/Python3
2023-09-08 00:08:15.056557
2023-09-08 00:08:16.057774
2023-09-08 00:08:17.058075
2023-09-08 00:08:18.059014
2023-09-08 00:08:19.059376
2023-09-08 00:08:20.060039
2023-09-08 00:08:21.060974
```

>4행 : While True : 로 시작 되는 로직이므로 반드시 참 을 반환하여
While 블록의 코드를 무한히 반복하게 된다.

* 현재 일시 를 멈추지 않고 1초 마다 나타낸다. 이같이 특정 조건을 만족할 경우 끊임없이 계속 반복되는 로직을 무한 루프 라고 한다.
* 단 무한루프 도 continue 와 break 등을 통하여 반복문 종료를 제어 할 수 있다.

. 정답을 맞출때까지 끝나지 않는 로직

```
Chap01_intro > 05.Loop[반복문].py > ...
1  print('3 + 7 = ?')
2  while True:
3      result = int(input('정답은 ? '))
4      if (result == 7) : break
5  print('참 잘했습니다')
6
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL
Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/Python311
3 + 7 = ?
정답은 ? 4
정답은 ? 5
정답은 ? 6
정답은 ? 7
참 잘했습니다
```

실습

1. While 문을 사용하여 1 에서 200 까지 모든 3의 배수 를 모두 표현하고 누적 합계를 출력 하세요

```
3,6,9,12,15,18,21,24,27,30,33,36,39,42,45,48,51,54,57,60,63,66,69,72,75,78,81,84,87,90,93,96,99,
--------------------------------------------------
 101
```

2. 아래와 같은 별 표 삼각형을 그리는 로직을 구현 해 보세요

3. 아래와 같은 별 표 삼각형을 그리는 로직을 구현 해 보세요

4. 0 부터 9999 까지 의 수 중 6 이 들어가지 않은 수 의 개수를 구하세요

```
6561
```