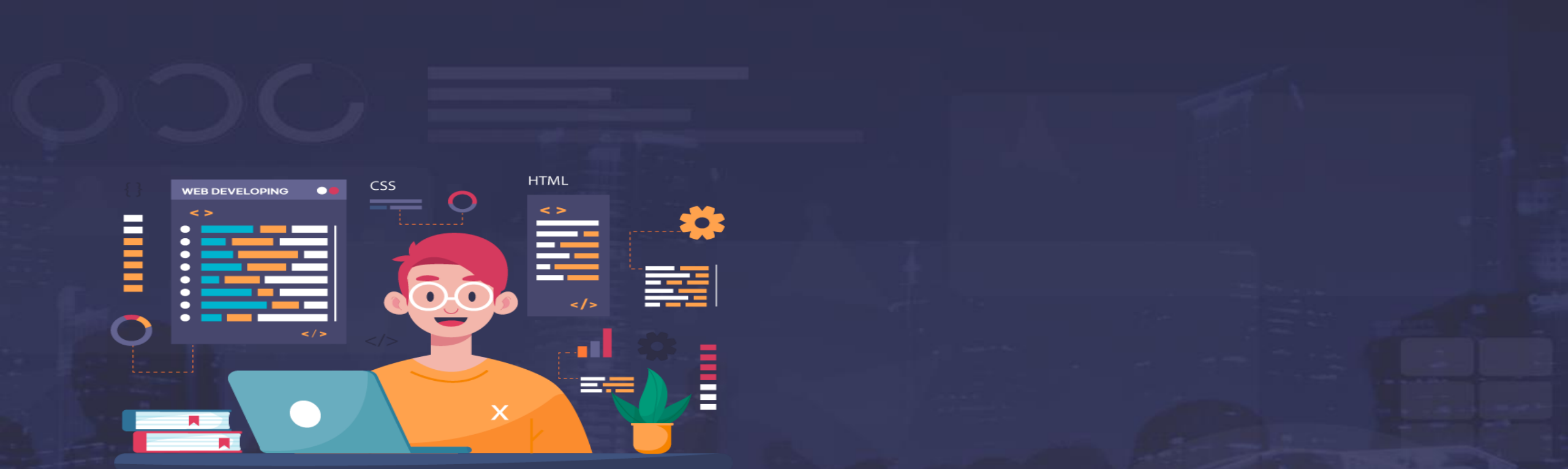


비교과 겨울 방학 특강

파이썬 3일 - 수업자료



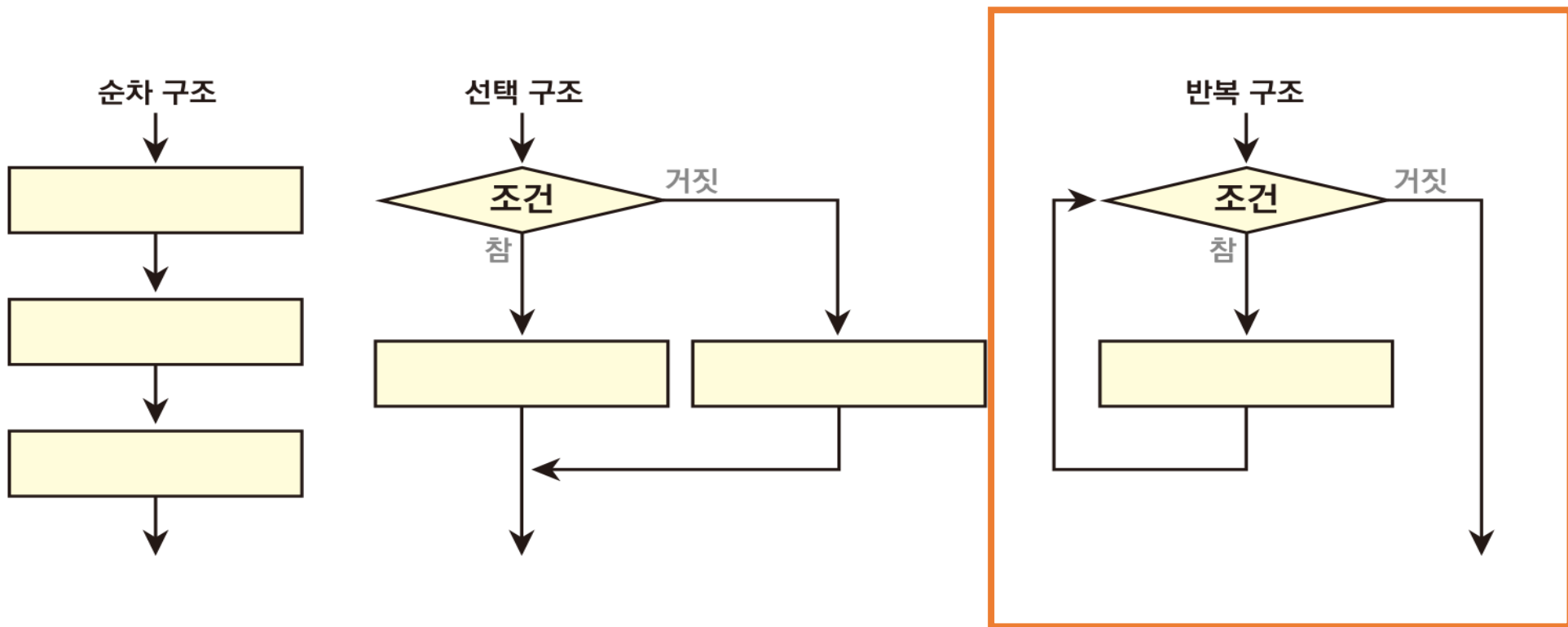
양 자영



반복문 - for

프로그램에서 명령어 기본 제어 구조

- 순차 구조(sequence) : 명령어들이 순차적으로 실행되는 구조
- 선택 구조(selection) : 둘 중의 하나의 명령을 선택하여 실행되는 구조
- 반복 구조(iteration) : 동일한 명령이 반복되면서 실행되는 구조



프로그램에서 반복구조의 필요성



- 동일한 작업을 여러 번 수행할 때 무엇이 더 간결할까?
 - 반복되는 패턴을 찾아 **반복구조로** 표현해준다면 훨씬 프로그램이 간결하고 명료해질 수 있다.

똑같은 문장 여러 번 쓰기 vs 반복구조

같은 작업을 여러 번 나열하여 작성

```
print("방문을 환영합니다!")  
print("방문을 환영합니다!")  
print("방문을 환영합니다!")  
print("방문을 환영합니다!")  
print("방문을 환영합니다!")
```

반복 구조를 이용하여 작성한 코드

```
for i in range(5):  
    print("방문을 환영합니다!")
```

반복문 - for 문 : 횟수제어 반복



- 반복을 수행하는 방법은 for, while 두가지 문법이 있다.
- 파이썬에서 **횟수** 제어 반복은 for 루프라고도 한다.

- for의 구조

반복되는 문장들여쓰기

```
for 변수 in 리스트(또는 튜플, 문자열):  
    수행할 문장1  
    수행할 문장2  
    ...
```

횟수 제어 반복(for ~)

- "안녕하세요"를 5번 출력하기

```
for i in range(5):  
    print("안녕하세요")
```

안녕하세요
안녕하세요
안녕하세요
안녕하세요
안녕하세요

for 변수 in range (종료 값) :

문장

0에서 (종료 값-1)까지의 숫자를 반환합니다.

반복되는 문장으로 들여쓰기 하여야 합니다.

```
for i in range(1000):  
    print("방문을 환영합니다!")
```

0에서 종료 값 -1 까지 숫자

실습 - 3의 배수 판별 5번 반복실행하기

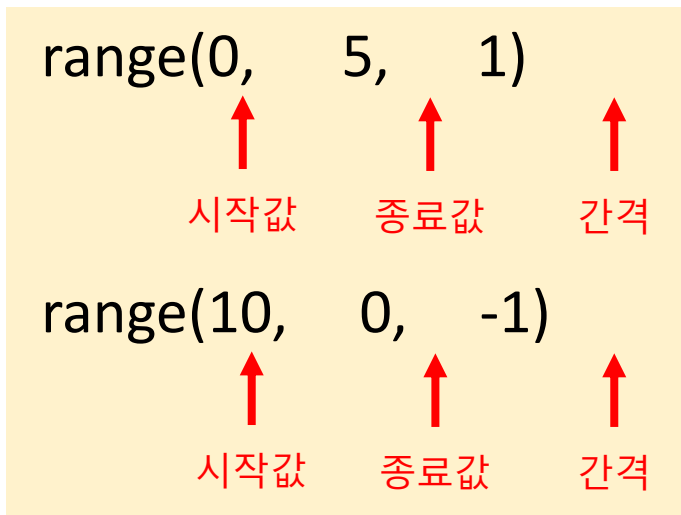


- 숫자를 입력받고, "3으로 나누어떨어진다", "3으로 나누어 떨어지지 않는다"를 출력하는 프로그램을 5번 반복하여 실행하시오.

```
숫자 : 15
3으로 나누어 떨어진다.
숫자 : 3
3으로 나누어 떨어진다.
숫자 : 5
3으로 나누어 떨어지지 않는다.
숫자 : 1
3으로 나누어 떨어지지 않는다.
숫자 : 2
3으로 나누어 떨어지지 않는다.
```

range() 함수

- 시작값부터 **종료값 이전까지** 일정한 간격으로 숫자를 순서대로 만들어 내는 함수
(종료값은 포함되지 않음)



```
for i in range(0, 5, 1):  
    print(i, end = " ")
```

0 1 2 3 4

```
for i in range(10, 0, -1):  
    print(i, end = " ")
```

10 9 8 7 6 5 4 3 2 1

용법	설명	포함된 숫자
range(5)	range(끝 숫자)	0, 1, 2, 3, 4
range(1, 5)	range(시작 숫자, 끝 숫자)	1, 2, 3, 4
range(1, 10, 2)	range(시작 숫자, 끝 숫자, 뛰어 세기)	1, 3, 5, 7, 9

- for와 range()를 사용하여 0부터 10까지 출력하시오.
- for와 range()를 사용하여 5부터 15까지 출력하시오.
- for와 range()를 사용하여 0~21까지의 짝수를 모두 출력하시오.
- for와 range()를 사용하여 -100~100까지의 4의 배수를 모두 출력하시오.

횟수 제어 반복 예제



- 1에서 10까지의 자연수의 합을 구하여 출력 프로그램을 작성하시오.

```
sum = 0
for i in range(1, 11, 1):
    sum += i
print(sum)
```

55

sum = 1+2+3+... + 10

```
sum = 0
sum = sum + 1
sum = sum + 2
sum = sum + 3
...
sum = sum + 10
```

- 0에서 10까지의 짝수의 합을 구하는 프로그램을 작성하시오.

```
sum = 0
for i in range(0, 11, 1):
    if i % 2 == 0:
        sum += i
print(sum)
```

```
sum = 0
for i in range(0, 11, 2):
    sum += i
print(sum)
```

30

break와 continue



```
for n in range(1, 10, 1) :
```

```
    if n % 5 == 0 :
```

```
        break
```

```
    print(n, end = " ")
```

1 2 3 4

반복문 밖으로 빠져나옴

```
for n in range(1, 10, 1):
```

```
    if n % 5 == 0 :
```

```
        continue
```

```
    print(n, end = " ")
```

해당 차례를 건너뛰고 새롭게 다음 차례의 반복 실행

1 2 3 4 6 7 8 9

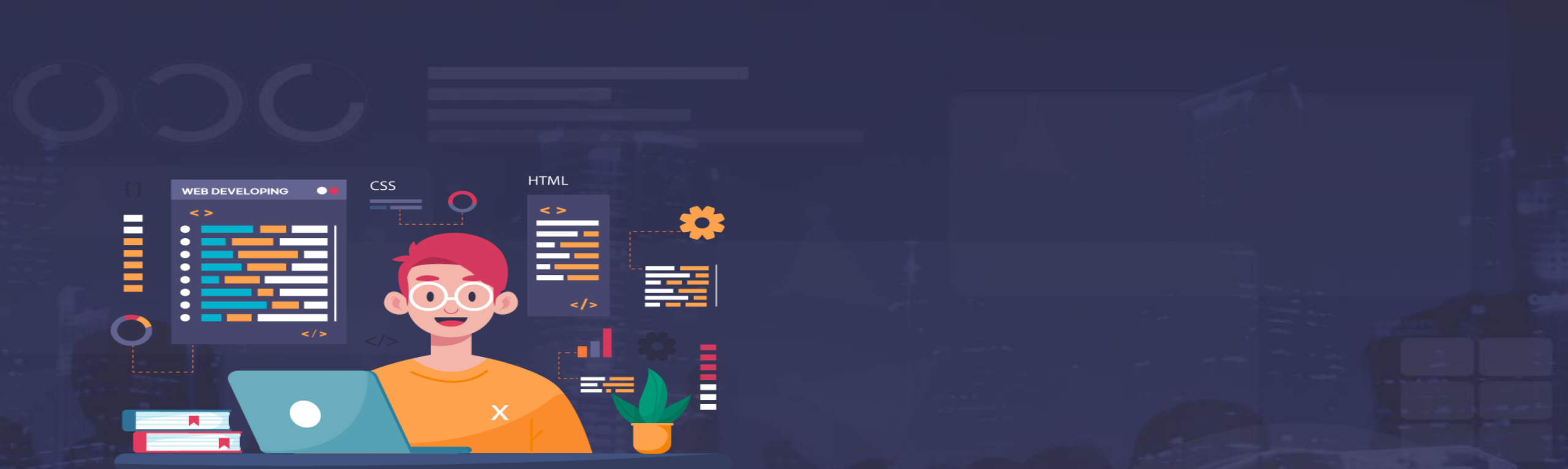
실습 - 소수 판별하기

- 양의 정수를 입력받고, 소수인지 판별하는 프로그램을 작성하시오.

1. 숫자를 입력받는다
2. isPrime 변수를 True로 둔다.
3. k는 2부터 수-1(또는 수를 2로 나눈 정수)까지 반복한다.
4. 입력받은 수를 반복문의 변수로 나누어 떨어지면
5. isPrime 변수를 False로 둔다.
6. 반복문을 멈춘다.
7. isPrime이 True 이면
8. 소수를 출력한다.
9. 아니면
10. 복합수 를 출력한다.

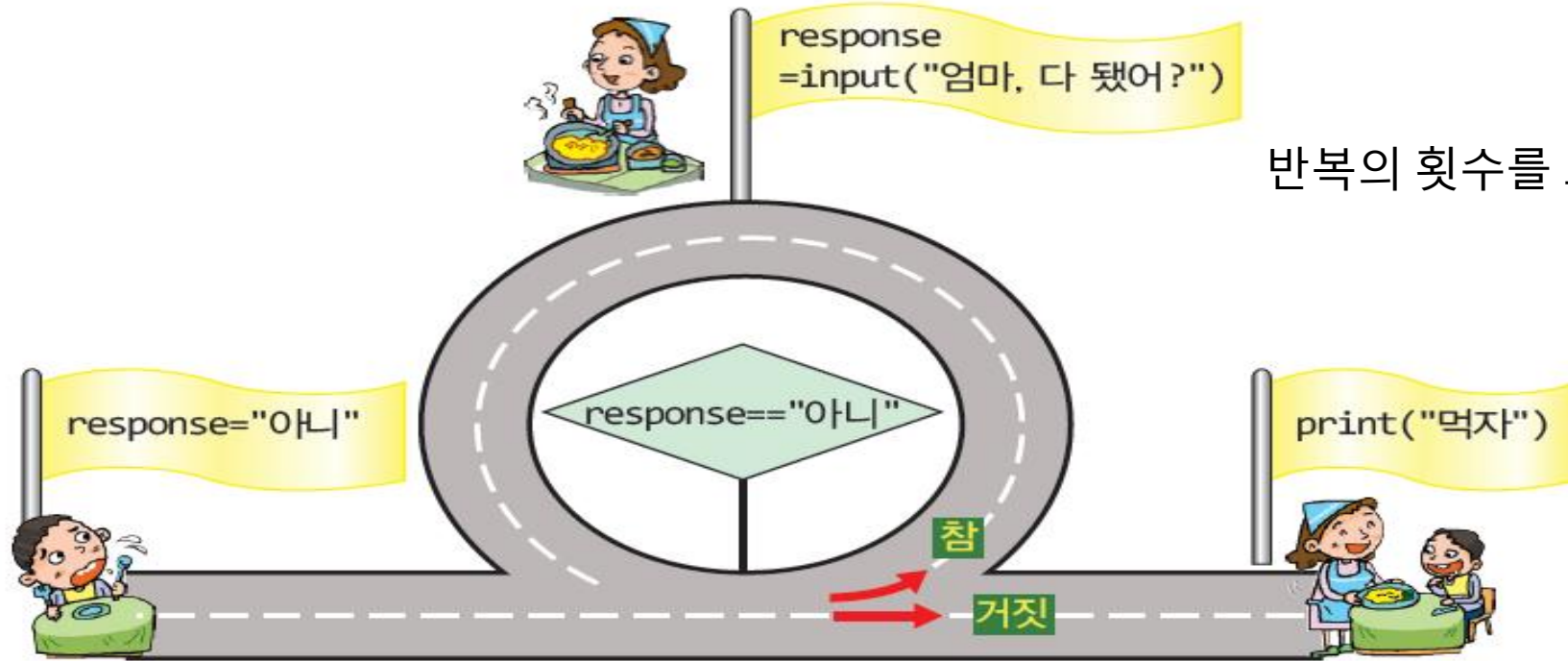
숫자 : 5
5 은 소수입니다

숫자 : 15
15 은 복합수입니다



반복문 - while

while 문



```
response = "아니"
while response == "아니":
    response = input("엄마, 다 됐어?");
print("먹자")
```

반복문 – while 문



- 반복을 수행하는 방법은 for, while 두 가지 문법이 있다.
- while의 구조 : 조건을 만족할 때까지 반복

```
while 조건문:      콜론  
    수행할 문장1  
    수행할 문장2  
    수행할 문장3  
    ...
```

조건 제어 반복(while ~)



- 1에서 10까지 자연수를 출력하는 프로그램을 작성하시오.

```
i = 1
while i <= 10 :
    print(i, end = " ")
    i += 1
```

1 2 3 4 5 6 7 8 9 10

vs

```
for i in range(1, 11, 1) :
    print(i, end = " ")
```

- 1에서 10까지의 자연수의 합을 구하여 출력하는 프로그램을 작성하시오.

```
i = 1
sum = 0
while i <= 10 :
    sum += i
    i += 1
print(sum)
```

55

vs

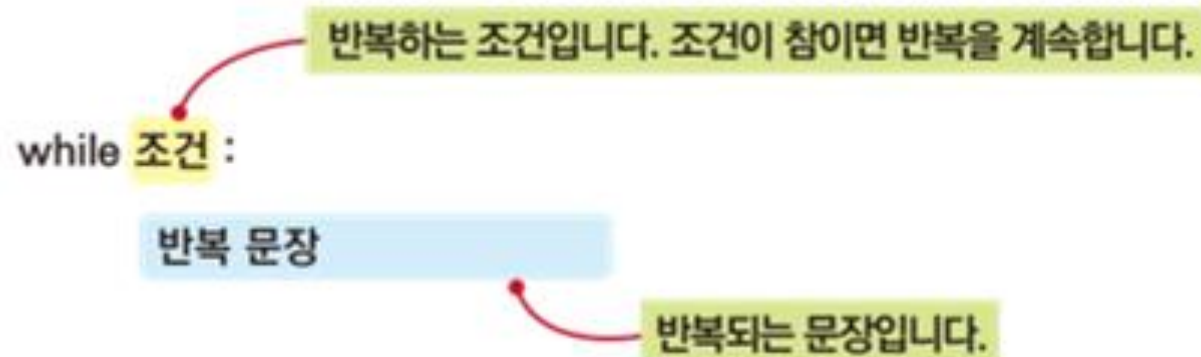
```
sum = 0
for i in range(1, 11, 1) :
    sum += i
print(sum)
```


조건 제어 반복(while ~)

- 사용자가 맞는 비밀번호를 넣을 때까지 계속 입력을 받고, 맞는 비밀번호를 입력하면 "로그인 성공"을 출력하는 프로그램을 작성하시오.

```
pw = ""  
while pw != "python" :  
    pw = input("비밀번호를 넣으세요 : ")  
print("로그인 성공")
```

비밀번호를 넣으세요 : abcde
비밀번호를 넣으세요 : kkkkk
비밀번호를 넣으세요 : 123456
비밀번호를 넣으세요 : python
로그인 성공



- 다음과 같은 모양을 출력하는 프로그램을 작성하시오.

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

```
for i in range(5) :
    for j in range(10) :
        print ("*", end = " ")
    print("")
```

- for 문을 이용하여 다음을 출력하는 프로그램을 작성하라.

```
5 4 3 2 1
4 3 2 1
3 2 1
2 1
1
```

- 반복이 무한히 발생하는 것. 따라서, 특정 조건에서 무한 반복이 멈추도록 해주어야 함.

```
pw = ""  
while pw != "python" :  
    pw = input("비밀번호를 넣으세요 : ")  
print("로그인 성공")
```

```
notPW = True
```

```
while notPW :
```

```
    pw = input("비밀번호를 넣으세요 : ")
```

```
    if pw == "python" :
```

```
        notPW = False
```

```
print("로그인 성공")
```

```
while True :
```

```
    pw = input("비밀번호를 넣으세요 : ")
```

```
    if pw == "python" :
```

```
        break
```

```
print("로그인 성공")
```

실습 - 총점 구하기

- 5명의 점수를 하나씩 입력받고, 그 총합을 출력한다.

숫자 : 90
숫자 : 98
숫자 : 100
숫자 : 50
숫자 : 60
총합은 398 입니다.

1. total변수를 0으로 정한다.
2. 5번 반복한다.
 - ① 숫자를 입력받아 num에 저장한다
 - ② total변수에 num을 더한다
3. total을 출력한다.

- 여러 개의 정수를 하나씩 입력받고, 그 총합을 출력한다.
단, -999를 입력하면 입력을 중단한다.

숫자를 입력하시오.(종료: -999)13
숫자를 입력하시오.(종료: -999)45
숫자를 입력하시오.(종료: -999)-8
숫자를 입력하시오.(종료: -999)22
숫자를 입력하시오.(종료: -999)-999
합은 72입니다.

1. total변수를 0으로 정한다.
2. 무한 반복한다
 - ① 숫자를 입력받아 num에 저장한다
 - ② num이 -999이면 반복문을 빠져 나간다.
 - ③ total변수에 num을 더한다
3. total을 출력한다.

실습 - up & down



```
===== RESTART: C:/Python/chapter4_3_a_1.py =====
예상 숫자를 입력하세요 : 45
UP
예상 숫자를 입력하세요 : 52
DOWN
예상 숫자를 입력하세요 : 50
정답
>>>
```

Ln: 12 Col: 0

숫자(1~50)를 묻고 맞추는 프로그램을 작성하시오.

변수 : 입력한 숫자, 맞추려는 숫자

입력한 숫자와 맞추려는 숫자가 같지 않을 동안 반복한다.

만약에 입력한 숫자가 맞추려는 숫자보다 크면 Up 을 출력한다.

만약에 입력한 숫자가 맞추려는 숫자보다 작으면 Down 을 출력한다

정답이라고 출력한다.

```
import random
n = random.randint(1,50)
print(n) #랜덤으로 1에서 50 출력
```

randint(start, stop) : start와 stop 사이의 랜덤 정수를 리턴 (stop도 포함)
randrange(start, stop, step) : start, stop 사이의 정수 중 step 간격의 랜덤 정수 리턴