

---

# Lecture 3

반복문

---

## 과거의 미션...

- 길이가 다른 단어 5개를 f-string으로 균일하게 출력하는 미션
- 사용자의 입력값을 5번 받아서 출력하는 미션

## 반복문 – for문 기초

- 코드를 n번 반복하고 싶으면 for문을 사용하면 된다
- 여기서 주의할 점은 바로 **들여쓰기 (indentation)**!
  - 반복하는 줄들(for문 안에 넣을 코드)은 for문 만든 줄에 있는 들여쓰기 칸 수보다 많아야 되고, 같은 for문 아래에 있는 줄들의 들여쓰기 칸 수가 같아야 한다

```
1 for i in range(5):  
2     print("Hello world!")
```

  
들여쓰기

## 반복문 – for문 기초

- 근데 for문에 들어가는 i는 무엇이고 range()는 무엇이지?

```
1 for i in range(5):  
2     print("Hello world!")
```

## 자료형 – range

- range 자료형은 정수를 순서대로 반환할 수 있는 자료형이다
- range(n)으로 만든 객체는 0부터 n-1까지의 정수를 하나씩 반환하는 기능을 갖고 있다
- 객체를 하나씩 반환하는 기능을 갖고 있는 자료형을 iterable 자료형이라고 부른다

```
1 for i in range(5):  
2     print("Hello world!")
```

## 자료형 – range

- range()를 여러 방법으로 호출할 수 있다
  - range(start, end): start부터 end-1까지의 정수를 하나씩 반환한다
  - range(start, end, step): start부터 end-1까지의 정수를 step씩 증가시키면서 하나씩 반환한다
  - start가 end보다 크면 range 객체가 아무것도 반환하지 않아 아예 반복이 안 된다

## 자료형 – range

- range()를 여러 방법으로 호출할 수 있다
  - range(start, end, step): start부터 end-1까지의 정수를 step씩 증가시키면서 하나씩 반환한다
  - 음수인 step도 가능하다
  - step이 음수면 start가 end보다 커야 반복이 된다

## 반복문 – for문 기초

- i는 그냥 변수 이름이다; 다른 이름을 주어도 괜찮다
- for문이 돌아가면서 i라는 변수에 range 객체가 반환하는 요소를 대입시킨다

```
1 for i in range(5):  
2     print("Hello world!")
```



## 반복문 – for문 기초

- i는 그냥 변수 이름이다; 다른 이름을 주어도 괜찮다
- for문이 돌아가면서 i라는 변수가 range 객체가 반환하는 요소를 받는다

```
→ 1 for i in range(5):  
   2     print("Hello world!")
```

i → 0



## 반복문 – for문 기초

- i는 그냥 변수 이름이다; 다른 이름을 주어도 괜찮다
- for문이 돌아가면서 i라는 변수가 range 객체가 반환하는 요소를 받는다

```
1 for i in range(5):  
→ 2     print("Hello world!")
```

i → 0

```
Hello world!
```

## 반복문 – for문 기초

- i는 그냥 변수 이름이다; 다른 이름을 주어도 괜찮다
- for문이 돌아가면서 i라는 변수가 range 객체가 반환하는 요소를 받는다

```
→ 1 for i in range(5):  
   2     print("Hello world!")
```

i → 1

```
Hello world!
```

## 반복문 – for문 기초

- i는 그냥 변수 이름이다; 다른 이름을 주어도 괜찮다
- for문이 돌아가면서 i라는 변수가 range 객체가 반환하는 요소를 받는다

```
1 for i in range(5):  
→ 2     print("Hello world!")
```

i → 1

```
Hello world!  
Hello world!
```

## 반복문 – for문 기초

- i는 그냥 변수 이름이다; 다른 이름을 주어도 괜찮다
- for문이 돌아가면서 i라는 변수가 range 객체가 반환하는 요소를 받는다

```
→ 1 for i in range(5):  
   2     print("Hello world!")
```

i → 2

```
Hello world!  
Hello world!
```

## 반복문 – for문 기초

- i는 그냥 변수 이름이다; 다른 이름을 주어도 괜찮다
- for문이 돌아가면서 i라는 변수가 range 객체가 반환하는 요소를 받는다

```
1 for i in range(5):  
→ 2     print("Hello world!")
```

i → 2

```
Hello world!  
Hello world!  
Hello world!
```

## 반복문 – for문 기초

- i는 그냥 변수 이름이다; 다른 이름을 주어도 괜찮다
- for문이 돌아가면서 i라는 변수가 range 객체가 반환하는 요소를 받는다

```
→ 1 for i in range(5):  
   2     print("Hello world!")
```

i → 3

```
Hello world!  
Hello world!  
Hello world!
```

## 반복문 – for문 기초

- i는 그냥 변수 이름이다; 다른 이름을 주어도 괜찮다
- for문이 돌아가면서 i라는 변수가 range 객체가 반환하는 요소를 받는다

```
1 for i in range(5):  
→ 2     print("Hello world!")
```

i → 3

```
Hello world!  
Hello world!  
Hello world!  
Hello world!
```



## 반복문 – for문 기초

- i는 그냥 변수 이름이다; 다른 이름을 주어도 괜찮다
- for문이 돌아가면서 i라는 변수가 range 객체가 반환하는 요소를 받는다

```
→ 1 for i in range(5):  
   2     print("Hello world!")
```

i → 4

```
Hello world!  
Hello world!  
Hello world!  
Hello world!
```

## 반복문 – for문 기초

- i는 그냥 변수 이름이다; 다른 이름을 주어도 괜찮다
- for문이 돌아가면서 i라는 변수가 range 객체가 반환하는 요소를 받는다

```
1 for i in range(5):  
→ 2     print("Hello world!")
```

i → 4

```
Hello world!  
Hello world!  
Hello world!  
Hello world!  
Hello world!
```

## 반복문 – for문 기초

- 반복문 안에 i의 값을 사용할 수 있다

```
1 for i in range(5):  
2     print(i)
```

## 반복문 – for문 기초

- for문에서 꼭 range() 함수를 쓸 필요가 없다
  - in 오른쪽에 있는 객체는 아무 iterable이면 된다
  - 문자열도 iterable 자료형이다!

## 반복문 – while문 기초

- n번 돌리는 것보다 어떤 조건이 계속 충족되고 있을 때 코드를 반복하고 싶을 수도 있다
- while condition: 으로 while문을 활용할 수 있다
- while문을 처음 들어갈 때, 그리고 반복할 코드의 실행이 끝날 때 condition의 진리값이 확인된다
  - condition의 진리값이 True면 반복할 코드를 실행한다
  - False면 멈추고 while문에서 탈출한다
- while문도 들여쓰기 신경써야 된다

```
1 i = 5
2 while i > 0:
3     print("Hello!")
4     i -= 1
```

## 반복문 – while문 기초

```
→ 1 i = 5  
   2 while i > 0:  
   3     print("Hello!")  
   4     i -= 1
```

i → 5

## 반복문 – while문 기초

```
1 i = 5  
→ 2 while i > 0:   → True  
3     print("Hello!")  
4     i -= 1
```

i → 5

## 반복문 – while문 기초

```
1 i = 5  
2 while i > 0:  
→ 3     print("Hello!")  
4     i -= 1
```

Hello!





## 반복문 – while문 기초

```
1 i = 5
2 while i > 0:
3     print("Hello!")
→ 4     i -= 1
```

Hello!



## 반복문 – while문 기초

```
1 i = 5  
→ 2 while i > 0:   → True  
3     print("Hello!")  
4     i -= 1
```

i → 4

Hello!

## 반복문 – while문 기초

```
1 i = 5
2 while i > 0:
→ 3     print("Hello!")
4     i -= 1
```

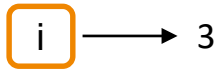
i → 4

```
Hello!
Hello!
```

## 반복문 – while문 기초

```
1 i = 5
2 while i > 0:
3     print("Hello!")
→ 4     i -= 1
```

```
Hello!
Hello!
```



## 반복문 – while문 기초

```
1 i = 5  
→ 2 while i > 0:   → True  
3     print("Hello!")  
4     i -= 1
```

i → 3

```
Hello!  
Hello!
```

## 반복문 – while문 기초

```
1 i = 5  
2 while i > 0:  
→ 3     print("Hello!")  
4     i -= 1
```

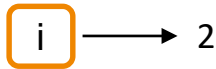
i → 3

```
Hello!  
Hello!  
Hello!
```

## 반복문 – while문 기초

```
1 i = 5  
2 while i > 0:  
3     print("Hello!")  
→ 4     i -= 1
```

```
Hello!  
Hello!  
Hello!
```



## 반복문 – while문 기초

```
1 i = 5  
→ 2 while i > 0:   → True  
3     print("Hello!")  
4     i -= 1
```

i → 2

```
Hello!  
Hello!  
Hello!
```



## 반복문 – while문 기초

```
1 i = 5  
2 while i > 0:  
→ 3     print("Hello!")  
4     i -= 1
```

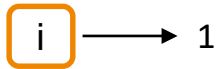
i → 2

```
Hello!  
Hello!  
Hello!  
Hello!
```

## 반복문 – while문 기초

```
1 i = 5
2 while i > 0:
3     print("Hello!")
→ 4     i -= 1
```

```
Hello!
Hello!
Hello!
Hello!
```



## 반복문 – while문 기초

```
1 i = 5  
→ 2 while i > 0:   → True  
3     print("Hello!")  
4     i -= 1
```

i → 1

```
Hello!  
Hello!  
Hello!  
Hello!
```

## 반복문 – while문 기초

```
1 i = 5  
2 while i > 0:  
→ 3     print("Hello!")  
4     i -= 1
```

i → 2

```
Hello!  
Hello!  
Hello!  
Hello!  
Hello!
```

## 반복문 – while문 기초

```
1 i = 5
2 while i > 0:
3     print("Hello!")
→ 4     i -= 1
```

i → 0

```
Hello!
Hello!
Hello!
Hello!
Hello!
```

## 반복문 – while문 기초

```
1 i = 5  
→ 2 while i > 0:   → False  
3     print("Hello!")  
4     i -= 1
```

i → 0

```
Hello!  
Hello!  
Hello!  
Hello!  
Hello!
```

## 반복문 – while문 기초

```
1 i = 5
2 while i > 0:
3     print("Hello!")
4     i -= 1
```



```
Hello!
Hello!
Hello!
Hello!
Hello!
```

