

문자열

들어가기

- 데이터는 다양한 타입으로 저장된다. 특히 많이 사용하게 될 문자열은 빼놓을 수 없다.
- 파이썬의 문자열 특성과 루프와의 활용될 접점을 찾아보자.

학습 목표

- 문자열의 특성을 이해하고 루프와 함께 문자(Character) 타입을 출력할 수 있다.

핵심 키워드

- 문자열

강의 내용

문자열

- 문자열 타입과 관련해서는 **타입 변환, 인덱싱, len() 함수, for 루프 활용** 을 이해하고 사용할 수 있으면 된다.

1. 문자열 읽기, 타입 변환

- 문자열을 사용한 데이터를 읽어 오게 되면, 우리는 에러나 사용자 입력에 대해 많은 대처를 할 수 있게 된다.
- 또한 사용자 입력으로 들어오는 값은 문자열 타입으로 입력되므로, 입력된 값으로 다른 무엇인가를 하기 원한다면 → 적절한 타입 변환을 해줘야 한다.

```
name = input('Enter: ')
print(type(name))
print(name)

# > Enter: 123 으로 입력합니다.
# 인풋값 123의 타입은 <class 'str'>과 같습니다.
# 123으로 출력됩니다.
```

2. 문자열의 내부 들여다 보기

- 우리는 특정 문자열을 구성하고 있는 개별 문자 값에 인덱스를 활용해서 접근할 수 있다.
- 여기서 유의해야 할 것은 첫 번째 오는 문자에 대한 인덱스는 0 부터 시작한다는 점이다.
- 만약 해당 문자열이 가지고 있는 인덱스를 넘어서는 값을 호출하게 되면 인덱스 오류가 발생하게 된다.

```
fruit = 'banana'
letter = fruit[0]
print(letter)
letter = fruit[1]
print(letter)
letter = fruit[2]
print(letter)

# b로 출력
# a로 출력
# n로 출력
```

3. len() 함수

- 문자열에 대해서 우리는 len() 내장 함수를 활용해서 문자열의 길이를 알 수 있다.
- 예를 들어, len(banana) 라고 한다면 banana가 몇 개의 문자로 구성되어 있는지를 알 수 있게 되는 것이다.

```
fruit = 'banana'
print(len(fruit))

# 6
```

4. 문자열의 길이만큼 루프 실행

- 우리는 len() 함수를 활용하면 문자열의 길이 만큼 루프를 실행할 수 있다.

```
fruit = 'banana'
index = 0

# while 루프

while index < len(fruit):
    letter = fruit[index]
    print(index, letter)
```

```
index += 1
```

```
# 0 b
```

```
# 1 a
```

```
# 2 n
```

```
# 3 a
```

```
# 4 n
```

```
# 5 a
```