

assertion

6주차_02_03

한 동 대 학 교
김경미 교수

학습목표

2

- ▶ assertion에 대해 알아보기
- ▶ assertion 사용하기

Assertion

- ▶ 원하는 범위의 값을 입력 받고 싶을 때, `assert()` 문장을 넣으면 그 범위의 값이 들어오지 않았을 때 에러메시지를 띄우고 프로그램이 중단 됨
- ▶ 에러 처리를 해서 진행하지 않고, 에러 처리가 힘들거나 중요한 값이라서 중단해야 할 때 사용 함

Assert 예제 1, 1~12 정수만 받는 경우(월)

4

```
month = int(input("월을 입력하시오: "))  
  
assert 1<=month<=12, "잘못된 월 입니다."
```

```
===== RESTART: /Users/hyerm2/Desktop/hi.py  
월을 입력하시오: 13  
Traceback (most recent call last):  
  File "/Users/hyerm2/Desktop/hi.py", line 2, in <module>  
    assert 1<=month<=12, "잘못된 월 입니다."  
AssertionError: 잘못된 월 입니다.
```

Assert 예제 2, 요일만 받는 경우

5

```
day = input("요일을 입력하시오: ")
```

```
assert day in ['월', '화', '수', '목', '금', '토', '일'], "잘못된 요일"
```

```
요일을 입력하시오: 월요일
Traceback (most recent call last):
  File "main.py", line 3, in <module>
    assert day in ['월', '화', '수', '목', '금', '토', '일'], "잘못된 요일"
AssertionError: 잘못된 요일
> █
```

```
요일을 입력하시오: 수
> █
```

Assertion 예제 3

6

```
#assert
```

```
def KToFahrenheit(Temperature):  
    assert (Temperature >= 0), print("Colder than absolute zero!")  
    return ((Temperature-273)*1.8)+32
```

```
print(KToFahrenheit(273))  
print(KToFahrenheit(505.78))  
print(KToFahrenheit(-5))
```

```
>>>  
32.0  
451.00399999999996  
Colder than absolute zero!  
Traceback (most recent call last):  
  File "C:/Users/user/Desktop/eh.py", line 7, in <module>  
    print(KelvinToFahrenheit(-5))  
  File "C:/Users/user/Desktop/eh.py", line 2, in KelvinToFahrenheit  
    assert (Temperature >= 0), print("Colder than absolute zero!")  
AssertionError: None  
>>> |
```

Assertion 예제 4

7

```
#assert
```

```
num = int(input('Enter a positive number: '))  
assert(num > 0), print('Only positive numbers are allowed!')  
  
print("number is ", num )
```

```
>>>  
Enter a positive number: 7  
number is 7  
>>> ===== RESTART =====  
>>>  
Enter a positive number: -3  
Only positive numbers are allowed!  
Traceback (most recent call last):  
  File "C:/Users/user/Desktop/eh.py", line 12, in <module>  
    assert(num > 0), print('Only positive numbers are allowed!')  
AssertionError: None  
>>> |
```

연습문제 I

- ▶ 영어로 요일을 입력 받을 때 아닌 값이 들어오면 확인하는 `assert` 구문을 쓰시오

연습문제 1, 코드

```
import sys
```

```
day = input("영문소문자로 요일을 입력하시오: ")
```

```
assert day=="monday"or day=="tuesday"or day=="wendesday"or  
day=="thursday"or day=="friday"or day=="saturday"or day=="sunday", "error!!!"
```

```
print("It's available")
```

```
영문소문자로 요일을 입력하시오: Monday
```

```
Traceback (most recent call last):
```

```
File "C:/Users/user/Desktop/b.py", line 5, in <module>
```

```
    assert day=="monday"or day=="tuesday"or day=="wendesday"or day=="thursday"or  
    day=="friday"or day=="saturday"or day=="sunday", "error!!!"
```

```
AssertionError: error!!!
```

```
===== RESTART: C:/Users/user/Desktop/b.py =====
```

```
영문소문자로 요일을 입력하시오: monday
```

```
It's available
```

연습문제 2

10

- ▶ 음의 정수를 입력 받을 때 아닌 값이 들어오면 확인하는 `assert` 구문을 쓰시오

연습문제 2, 코드

11

```
num = int(input('Enter a negative number: '))
assert (num < 0) , print('Only negative numbers are allowed!')

print("number is ", num )
```

```
Enter a negative number: -2
number is  -2
```

```
Enter a negative number: 5
Only negative numbers are allowed!
Traceback (most recent call last):
  File "main.py", line 4, in <module>
    assert(num < 0), print('Only negative numbers are allowed!')
AssertionError: None
```

연습문제 3

12

- ▶ 나이를 입력 받는다
- ▶ 입력 받은 나이가 1보다 작고, 100보다 크면 오류라고 알려준다
- ▶ assert문을 사용한다

연습문제 3, 코드와 결과

13

```
#assert
```

```
age = int(input('Enter your age: '))
```

```
assert (age > 1 and age <= 100), print("나이는 1세에서 100세까지 만 입력 가능!")
```

```
print("age is ", age )
```

```
>>>
===== RESTART: E:/1_Works/2017Work/KMooC강의/Exercise Code/turtle_test.
py =====
Enter your age: 5
age is 5
>>>
===== RESTART: E:/1_Works/2017Work/KMooC강의/Exercise Code/turtle_test.
py =====
Enter your age: 102
나이는 1세에서 100세까지만 입력 가능!
Traceback (most recent call last):
  File "E:/1_Works/2017Work/KMooC강의/Exercise Code/turtle_test.py", li
ne 2, in <module>
    assert(age > 1 and age <= 100), print("나이는 1세에서 100세까지만
입력 가능!")
AssertionError: None
>>> |
```

강의 요약

14

- ▶ 입력하는 값의 범위를 지정하여 처리하기
 - ▶ `assert`(원하는 범위 지정)

목표 달성 질문

15

- ▶ 음수가 아닌 값만 입력 받는 `assert` 문장을 쓰시오

감사합니다