

파이썬을 활용한 실전 웹크롤링

1주차 - 보충자료

2018.07.21

신준호

jjjjoonno@gmail.com

목차 - 실습

1. 파이썬의 변수, 자료형, 객체

변수, 자료형 객체 이해하기의 목적

파이썬 기본 개념을 학습하여 추후 코드실습에 활용합니다.

변수, 자료형, 객체 이해하기

변수

글로 생각한다면 대명사에 해당하는 개념!

이것 = 사과라고 한다면 우리는 이러한 선언 뒤에서 ‘이것’을 사용한다면 사과가 이것에 대입되어 사용된다는 것을 알수 있습니다.

컴퓨터 언어에서도 $a = 3$ 이라고 선언한 후에 뒤에서 a 를 입력한다면 컴퓨터는 자동적으로 3으로 인식하게 됩니다.

또한 $a = 3$ 후에 다시 $a = 5$ 라고 선언한다면 가장 최근의 선언인 5로 인식하게 됩니다.

변수, 자료형, 객체 이해하기

```
a = 3
print(a)
# 결과값 = 3
a = 5
print(a)
# 결과값 = 5
b = 7
print(a+b)
# 결과값 = 12
```

변수, 자료형, 객체 이해하기

자료형

자료형이란 변수의 성격을 나타내는 것입니다.

파이썬에서는 여러가지 자료형이 있습니다.

이 강의에서는 기본적인 int(정수형) / float(실수형) / string(문자형)을 주로 사용할 것입니다.

정수형 : 소수점이 없는 숫자

Ex) 5

실수형 : 소수점을 포함한 숫자

Ex) 5.7

문자형 : 말 그대로 문자를 표현하는 자료형

Ex) '안녕하세요' - 문자형은 따옴표와 쌍따옴표를 사용하여 구분합니다.

변수, 자료형, 객체 이해하기

```
variable_int = 3
```

print(x)는 x를 출력하는 함수입니다.

```
print(variable_int)
```

```
variable_float = 3.6
```

```
print(variable_float)
```

str(x) / int(x) / float(x)는 괄호안의 x의 자료형을 변경해줍니다.

그러나 x가 문자열인 경우 int나 float로 변환은 불가능합니다.

```
variable_string = 'hello'
```

```
print(variable_string)
```

```
print(type(variable_int))
```

```
print(type(variable_string))
```

아래와 같이 " 혹은 ""로 감싼 대상은 문자열로 취급합니다.

```
variable_int_to_string = str(variable_int)
```

```
print(variable_int_to_string)
```

변수, 자료형, 객체 이해하기

기본 배열

리스트

['a','b',5]와 같은 형태로, 대괄호로 둘러싸여 있고, 내부에 문자열, 숫자 등의 다양한 자료형이 공존 가능합니다. 쉼표(,)로 각각의 item을 구분합니다.

딕셔너리

{'a': 5, 'b': 7}와 같은 형태로 중괄호로 둘러싸여 있고, 내부에 a, b와 같은 KEY와 5, 7과 같은 VALUE로 나누어져 구성됩니다. 마찬가지로 하나의 짝은 쉼표로 구분되며 KEY와 VALUE는 :으로 구분합니다.

변수, 자료형, 객체 이해하기

파이썬의 기본 배열중 리스트와 딕셔너리.

리스트 []를 사용하여 선언.

```
list_ = ['a', 3, 5]
```

list는 index를 사용하여 내용에 접근.

파이썬은 0부터 시작하는 index를 가지고 있습니다.

```
print(list_[0])
```

리스트에 새로운 변수를 삽입하는 함수 append

```
list_.append(7)
```

```
print(list_[3])
```

변수, 자료형, 객체 이해하기

딕셔너리

딕셔너리는 {key : value}로 이루어집니다.

key는 중복이 불가능하지만 value는 중복이 가능.

```
dic = {'a' : 1, 'b' : 2}
```

딕셔너리는 key값으로 내용에 접근이 가능합니다.

```
print(dic['a'])
```

딕셔너리에 새로운 변수를 삽입하는 함수 update

```
dic.update({'c' : 3})
```

```
print(dic['c'])
```

감사합니다.

