

《 Python: 내장 타입과 메서드 》

이 문서는 유료용으로 판매되는 강좌의 자료입니다. 무단 복제 및 배포를 금지합니다.

출처: www.youtube.com/@weekendcode // <https://inf.run/zSrvA>

빌트인 타입의 실제 값

```
values = [  
    42,                                # int  
    3.14,                             # float  
    1 + 2j,                           # complex  
    True,                             # bool  
    "Hello, World!",                  # str  
    [1, 2, 3],                        # list  
    (1, 2, 3),                        # tuple  
    {1, 2, 3},                        # set  
    {"a": 1, "b": 2, "c": 3},         # dict  
    b"Hello, World!",                 # bytes  
    bytearray(b"Hello, World!"),      # bytearray  
    memoryview(b"Hello, World!"),     # memoryview  
    range(10),                        # range  
    None                              # NoneType  
]
```

번호를 매겨서 실제 값과 타입을 출력하는 코드

```
for index, value in enumerate(values, start=1):  
    print(f"{index}. {value} - {type(value)}")
```

출력값:

1. 42 - <class 'int'>
2. 3.14 - <class 'float'>
3. (1+2j) - <class 'complex'>
4. True - <class 'bool'>
5. Hello, World! - <class 'str'>
6. [1, 2, 3] - <class 'list'>
7. (1, 2, 3) - <class 'tuple'>
8. {1, 2, 3} - <class 'set'>
9. {'a': 1, 'b': 2, 'c': 3} - <class 'dict'>
10. b'Hello, World!' - <class 'bytes'>
11. bytearray(b'Hello, World!') - <class 'bytearray'>
12. <memory at 0x7b62c4fc3040> - <class 'memoryview'>
13. range(0, 10) - <class 'range'>
14. None - <class 'NoneType'>

```
42,          # int
3.14,        # float
True,        # bool
"Hello, World!", # str
[1, 2, 3],    # list
(1, 2, 3),    # tuple
{1, 2, 3},    # set
{"a": 1, "b": 2}, # dict
```

숫자형

정수형 예제

```
n = 42
print(n.bit_length()) # 6
print(n.to_bytes(2, byteorder='big')) # b'\x00*'
print(int.from_bytes(b'\x00*', byteorder='big')) # 42
```

소수형 예제

```
f = 3.14
print(f.as_integer_ratio())
print(f.is_integer()) # False
print(f.hex()) # '0x1.91eb851eb851fp+1'
print(float.fromhex('0x1.91eb851eb851fp+1')) # 3.14
```

참/거짓형

```
b = True
c = False
print(int(b)) # 1
print(int(c)) # 0
print(b == 1) # True
print(b + 1) # 2
```

문자형태

```
s = "Hello, World!"
print(s.lower()) # 'hello, world!'
print(s.upper()) # 'HELLO, WORLD!'
print(s.split(',')) # ['Hello', ' World!']
print(', '.join(['Hello', 'World'])) # 'Hello, World'
print(s.replace('World', 'Python')) # 'Hello, Python!'
print(s.find('World')) # 7
print(s.index(',')) # 5
```

```
s = "Hello, World!"
print(s.capitalize()) # 'Hello, world!'
print(s.casefold()) # 'hello, world!'
print(s.count('o')) # 2
print(s.endswith('!')) # True
print("123".isdigit()) # True
print(" Hello".lstrip()) # 'Hello'
print(s.partition(',')) # ('Hello', ',', ' World!')
print(s.swapcase()) # 'hELLO, wORLD!'
s_lines = "Hello\nWorld"
print(s_lines.splitlines()) # ['Hello', 'World']
print(s.startswith('Hello')) # True
print(s.title()) # 'Hello, World!'
print("42".zfill(5)) # '00042'
```

리스트형

```
lst = [1, 2, 3]
lst.append(4)
print(lst)  # [1, 2, 3, 4]

lst.extend([5, 6])
print(lst)  # [1, 2, 3, 4, 5, 6]

lst.insert(2, 7)
print(lst)  # [1, 2, 7, 3, 4, 5, 6]

lst.remove(7)
print(lst)  # [1, 2, 3, 4, 5, 6]

print(lst.pop())  # 6
print(lst)  # [1, 2, 3, 4, 5]

lst.clear()
print(lst)  # []

lst = [1, 2, 3, 2]
print(lst.index(2))  # 1

print(lst.count(2))  # 2

lst.sort()
print(lst)  # [1, 2, 2, 3]

lst.reverse()
print(lst)  # [3, 2, 2, 1]

lst_copy = lst.copy()
print(lst_copy)  # [3, 2, 2, 1]
```

이 문서는 유료용으로 판매되는 강좌의 자료입니다. 무단 복제 및 배포를 금지합니다.

출처: www.youtube.com/@weekendcode // <https://inf.run/zSrvA>