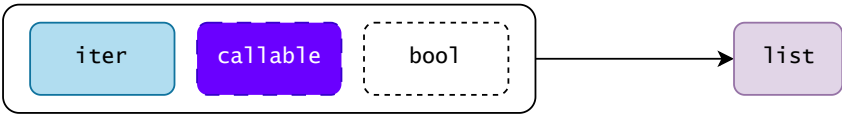


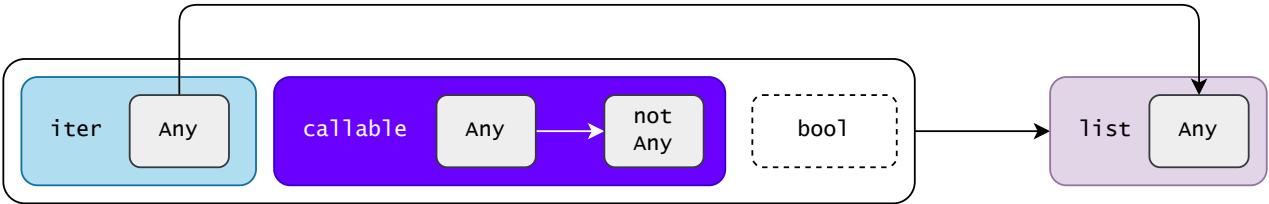
# SORTED

`sorted(iterable, key=None, reverse=False)`



Первый аргумент это итерируемый тип, второй - вызываемый, третий bool  
Обязательным является только первый аргумент  
Возвращает список элементов итерируемого объекта

По умолчанию (`reverse=False`) сортирует по возрастанию



Вызываемый тип должен принимать на вход только 1 аргумент.  
Но стоит обратить внимание что на вход `callable` будут подаваться элементы из `iter`.  
`callable` должен вернуть тип данных, который поддерживает сравнение `<` `>`.

## Примеры:

```
list1 = [5, 2, 3, 1, 4, 6]
print(sorted(list1)) # [1, 2, 3, 4, 5, 6]
```

```
list1 = [5, 2, 3, 1, 4, 6]
print(sorted(list1, reverse=True)) # [6, 5, 4, 3, 2, 1]
```

```
list1 = [5, 2, 3, 1, 4, 6, 1, 3, 4, 2, 2, 2]
print(sorted(list1)) # [1, 1, 2, 2, 2, 2, 3, 3, 4, 4, 5, 6]
```

## Сортировка отзывов по возрастанию оценок

```
reviews = [
    {
        "rating": 5,
        "comment": "Amazing sound quality and battery life!"
    },
    {
        "rating": 2,
        "comment": "Bad sound quality and too bulky."
    },
    {
        "rating": 4,
        "comment": "Great headphones but a bit pricey."
    }
]

def get_rating(review):
    return review["rating"]

print(sorted(reviews, key=get_rating))
```

```
reviews = [
    {
        "rating": 5,
        "comment": "Amazing sound quality and battery life!"
    },
    {
        "rating": 2,
        "comment": "Bad sound quality and too bulky."
    },
    {
        "rating": 4,
        "comment": "Great headphones but a bit pricey."
    }
]

print(sorted(reviews, key=lambda x: x["rating"]))
```

```
reviews = [
    {
        "rating": 5,
        "comment": "Amazing sound quality and battery life!"
    },
    {
        "rating": 2,
        "comment": "Bad sound quality and too bulky."
    },
    {
        "rating": 4,
        "comment": "Great headphones but a bit pricey."
    }
]

def invalid_func(review):
    return None

print(sorted(reviews, key=invalid_func))
```

Ошибка!

Функция возвращает None, а он не поддерживает сравнение на больше или меньше