```
max 2 3    # => 3
```

```
max  0 4 1    # => 4
```

```
min 'apple' 'banana' 'pear' key=len   # => 'pear'
```

```
sum 3 5 7    # => 15
```

```
pow 3 5          # => 243 (= 3 ** 5)
```

```
    3  5  10   # => 3 (= (3 ** 5) % 10, efficiently)
```

`divmod 10 6`

```
# quotient, remainder => (1, 4)
```

```python
# Flatten a list of lists (slower than itertools.chain)
```

```
3 5 , 1 7    4    []    # => [3, 5, 1, 7, 4]
```

Common One-Liners

```python
max(2, 3)  # => 3
max([0, 4, 1])  # => 4
min(['apple', 'banana', 'pear'], key=len)  # => 'pear'

sum([3, 5, 7])  # => 15

pow(3, 5)        # => 243 (= 3 ** 5)
pow(3, 5, 10)    # => 3 (= (3 ** 5) % 10, efficiently)

quotient, remainder = divmod(10, 6)
# quotient, remainder => (1, 4)

# Flatten a list of lists (slower than itertools.chain)
sum([[3, 5], [1, 7], [4]], [])  # => [3, 5, 1, 7, 4]
```

Other Modules

Modules that you should know exist