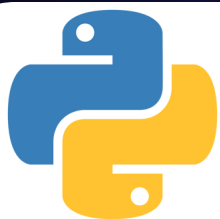
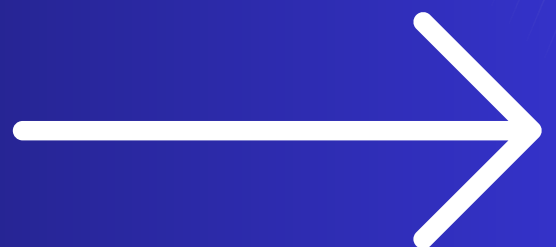


USE PYTHON LIST COMPREHENSIONS FOR CLEANER CODE



EREN HAN





Square Numbers

```
squares = [i**2 for i in range(10)]
```

Result: [0, 1, 4, 9, 16, 25, 36, 49, 64, 81]



Square Numbers

```
squares = []  
for i in range(10):  
    squares.append(i**2)
```

Result: [0, 1, 4, 9, 16, 25, 36, 49, 64, 81]





Filter Even Numbers

```
evens = [i for i in range(20) if i % 2 == 0]
```

Result: [0, 2, 4, 6, 8, 10, 12, 14, 16, 18]



Filter Even Numbers

```
evens = []  
for i in range(20):  
    if i % 2 == 0:  
        evens.append(i)
```

Result: [0, 2, 4, 6, 8, 10, 12, 14, 16, 18]



●●● Convert Strings to Uppercase

```
names = ['Nelly', 'Richard', 'Sam', 'Adam']  
upper_names = [name.upper() for name in names]
```

Result: ['NELLY', 'RICHARD', 'SAM', 'ADAM']



●●● Convert Strings to Uppercase

```
names = ['Nelly', 'Richard', 'Sam', 'Adam']  
upper_names = []  
for name in names:  
    upper_names.append(name.upper())
```

Result: ['NELLY', 'RICHARD', 'SAM', 'ADAM']



●●● Create a List of Tuple Pairs

```
pairs = [(x, y) for x in range(2) for y in range(2)]
```

Result: *[(0, 0), (0, 1), (1, 0), (1, 1)]*



●●● Create a List of Tuple Pairs

```
pairs = []  
for x in range(3):  
    for y in range(3):  
        pairs.append((x, y))
```

Result: *[(0, 0), (0, 1), (1, 0), (1, 1)]*



●●● Filter Strings by Length

```
names = ['Nelly', 'Richard', 'Sam', 'Adam']  
short_names = [name for name in names if len(name) < 5]
```

Result: ['Sam', 'Adam']



●●● Filter Strings by Length

```
names = ['Nelly', 'Richard', 'Sam', 'Adam']  
short_names = []  
for name in names:  
    if len(name) < 5:  
        short_names.append(name)
```

Result: ['Sam', 'Adam']



●●● Multiply Elements in a List

```
nums = [3, 5, 6, 4]  
doubled = [i * 2 for i in nums]
```

Result: [6, 10, 12, 8]



●●● Multiply Elements in a List

```
nums = [3, 5, 6, 4]  
doubled = []  
for i in nums:  
    doubled.append(i * 2)
```

Result: [6, 10, 12, 8]



●●● Create a List of Boolean Values

```
is_even = [True if i % 2 == 0 else False for i in range(5)]
```

Result: [True, False, True, False, True]



●●● Create a List of Boolean Values

```
is_even = []  
for i in range(5):  
    if i % 2 == 0:  
        is_even.append(True)  
    else:  
        is_even.append(False)
```

Result: [True, False, True, False, True]



●●● Flatten a 2D List

```
matrix = [  
    [1, 2, 3],  
    [4, 5, 6],  
    [7, 8, 9]  
]  
flat_list = [item for sublist in matrix for item in sublist]
```

Result: [1, 2, 3, 4, 5, 6, 7, 8, 9]



●●● Flatten a 2D List

```
matrix = [  
    [1, 2, 3],  
    [4, 5, 6],  
    [7, 8, 9]  
]  
flat_list = []  
for sublist in matrix:  
    for item in sublist:  
        flat_list.append(item)
```

Result: [1, 2, 3, 4, 5, 6, 7, 8, 9]



●●● List of Lengths of Strings

```
names = ['Nelly', 'Richard', 'Sam', 'Adam']  
lengths = [len(name) for name in names]
```

Result: [5, 7, 3, 4]



●●● List of Lengths of Strings

```
names = ['Nelly', 'Richard', 'Sam', 'Adam']  
lengths = []  
for name in names:  
    lengths.append(len(name))
```

Result: [5, 7, 3, 4]



●●● Filter and Modify

```
squares_of_evens = [i**2 for i in range(10) if i % 2 == 0]
```

Result: [0, 4, 16, 36, 64]



●●● Filter and Modify

```
squares_of_evens = []  
for i in range(10):  
    if i % 2 == 0:  
        squares_of_evens.append(i**2)
```

Result: [0, 4, 16, 36, 64]



●●● Comprehensions with Dictionaries

```
names = ['Nelly', 'Richard', 'Sam', 'Adam']  
name_lengths = {name: len(name) for name in names}
```

Result: {'Nelly': 5, 'Richard': 7, 'Sam': 3, 'Adam': 4}



●●● Comprehensions with Dictionaries

```
names = ['Nelly', 'Richard', 'Sam', 'Adam']  
name_lengths = {}  
for name in names:  
    name_lengths[name] = len(name)
```

Result: {'Nelly': 5, 'Richard': 7, 'Sam': 3, 'Adam': 4}



WAS IT HELPFUL?

follow for more!



EREN HAN

<https://www.linkedin.com/in/eren-han/>