**Python List Comprehension**

List comprehension offers a concise way to create a new list based on the values of an existing list.

Suppose we have a list of numbers and we desire to create a new list containing the double value of each element in the list.

numbers = [1, 2, 3, 4]

# list comprehension to create new list

doubled\_numbers = [num \* 2 for num in numbers]

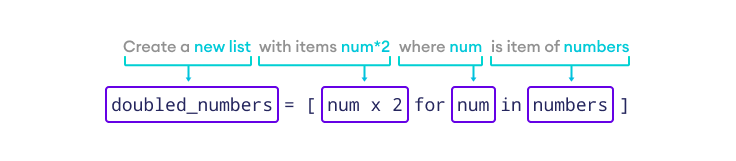
print(doubled\_numbers)

[Run Code](https://www.programiz.com/python-programming/online-compiler)

**Output**

[2, 4, 6, 8]

Here is how the list comprehension works:

Python List Comprehension

**Syntax of List Comprehension**

[expression for item in list if condition == True]

Here,

for every item in list, execute the expression if the condition is True.

**Note:** The if statement in list comprehension is optional.

**for Loop vs. List Comprehension**

List comprehension makes the code cleaner and more concise than for loop.

Let's write a program to print the square of each list element using both for loop and list comprehension.

**for Loop**

numbers = [1, 2, 3, 4, 5]

square\_numbers = []

# for loop to square each elements

for num in numbers:

square\_numbers.append(num \* num)

print(square\_numbers)

# Output: [1, 4, 9, 16, 25]

[Run Code](https://www.programiz.com/python-programming/online-compiler)

**List Comprehension**

numbers = [1, 2, 3, 4, 5]

# create a new list using list comprehension

square\_numbers = [num \* num for num in numbers]

print(square\_numbers)

# Output: [1, 4, 9, 16, 25]

[Run Code](https://www.programiz.com/python-programming/online-compiler)

It's much easier to understand list comprehension once you know [Python for loop()](https://www.programiz.com/python-programming/for-loop).

**Conditionals in List Comprehension**

List comprehensions can utilize conditional statements like [if…else](https://www.programiz.com/python-programming/if-elif-else) to filter existing lists.

Let's see an example of an if statement with list comprehension.

# filtering even numbers from a list

even\_numbers = [num for num in range(1, 10) if num % 2 == 0 ]

print(even\_numbers)

# Output: [2, 4, 6, 8]

[Run Code](https://www.programiz.com/python-programming/online-compiler)

Here, list comprehension checks if the number from range(1, 10) is even or odd. If even, it appends the number in the list.

**Note**: The range() function generates a sequence of numbers. To learn more, visit [Python range()](https://www.programiz.com/python-programming/methods/built-in/range).

if...else With List Comprehension

Nested if With List Comprehension

**Example: List Comprehension with String**

We can also use list comprehension with iterables other than lists.

word = "Python"

vowels = "aeiou"

# find vowel in the string "Python"

result = [char for char in word if char in vowels]

print(result)

# Output: ['o']

[Run Code](https://www.programiz.com/python-programming/online-compiler)