

List comprehensions

It is a common programming task to produce a list whose elements are the result of a function applied to another list. For example:

```
def square(x):  
    return x*x  
  
my_list = [1,2,3,4,5,6,7,8]  
  
new_list = []  
for x in my_list:  
    new_list.append(square(x))  
  
print(new_list)
```

This is so common that Python has special syntax to achieve the same thing:

```
list_comp = [x*x for x in my_list]  
print(list_comp)
```

This is called a list comprehension. It creates a new list by applying an operation to each item of another list.

It is also possible to filter out elements of list in a comprehension:

```
my_list = [1,2,3,4,5,6,7,8,9,10,11]  
odds = [x for x in my_list if x % 2 != 0]  
odds
```

Python also has set and dictionary comprehensions.

References

- List comprehensions in Python Tutorial