```
Python 3.14.0 (tags/v3.14.0:ebf955d, Oct  7 2025, 10:15:03) [MSC v.1944 64 bit
(AMD64)] on win32
Enter "help" below or click "Help" above for more information.
#-----ADVANCED LIST COMPREHENSION-----
#-----CONCEPT OF CARTESIAN PRODUCT----
C = '''
Simple example:
S1 = {1.1, 2.2, 3.3}
S2 = {10, 20}
We can take a product of sets S1 and S2.
What is the meaning of product in the context of sets?
One must paid every element in the first set with every other element
in the second set. Each pair is stored in the tuple of two elements.
Set of all such tuples is nothing but the product of two sets.

Applying above concept to S1 and S2
1.1 -> 10
1.1 -> 20
2.2 -> 10
2.2 -> 20
3.3 -> 10
3.3 -> 20
Store every pair in a tuple
(1.1, 10) (1.1, 20) (2.2, 10) (2.2, 20) (3.3, 10) (3.3, 20)

Cartesian_Product_Of_S1_and_S2 = {
(1.1, 10), (1.1, 20), (2.2, 10), (2.2, 20), (3.3, 10), (3.3, 20)
}
'''
L1 = [1.1, 2.2, 3.3]
L2 = [10, 20]
cartp = []
for x in L1:
    for y in L2:
        T = (x, y)
        cartp.append(T)


cartp
[(1.1, 10), (1.1, 20), (2.2, 10), (2.2, 20), (3.3, 10), (3.3, 20)]
# In basic list comprehension, there was only one variable
# Most advanced version looked like this:
# L = [f(v) for v in iterable]
# L = [f(v) for v in iterable if cond(v)]
# In advanced comprehension, we are going to make use of more than one variable
# Let us consider a concrete example before general syntax
L = [x + y   for x in [10, 20, 30]
             for y in [1.1, 2.2] ]
L
[11.1, 12.2, 21.1, 22.2, 31.1, 32.2]
```

```
# CPA : List comprehension version 5
# Example:
L = [(x, y) for x in [10, 20, 30] for y in [1.1, 2.2]]
print(L)
[(10, 1.1), (10, 2.2), (20, 1.1), (20, 2.2), (30, 1.1), (30, 2.2)]
# General form of version 5:
# L = [(x, y) for x in iterable_1 for y in iterable_2]
# List Comprehension version 6
L = [(x, y)  for x in range(1, 10) if x % 2 == 0
             for y in range(1, 10) if y % 2 == 1]
L
[(2, 1), (2, 3), (2, 5), (2, 7), (2, 9), (4, 1), (4, 3), (4, 5), (4, 7), (4, 9),
(6, 1), (6, 3), (6, 5), (6, 7), (6, 9), (8, 1), (8, 3), (8, 5), (8, 7), (8, 9)]
L = [x+y  for x in range(1, 10) if x % 2 == 0
             for y in range(1, 10) if y % 2 == 1]
L
[3, 5, 7, 9, 11, 5, 7, 9, 11, 13, 7, 9, 11, 13, 15, 9, 11, 13, 15, 17]
# General form of version 6
4
# L = [(v1, v2) for v1 in iterable_1 if cond_1(v1) for v2 in iterable_2 if
cond_2(v2)]

# General form of version 7
L = [(x**2, y**3) for x in range(1, 6) for y in range(9, 12)]
L
[(1, 729), (1, 1000), (1, 1331), (4, 729), (4, 1000), (4, 1331), (9, 729), (9,
1000), (9, 1331), (16, 729), (16, 1000), (16, 1331), (25, 729), (25, 1000), (25,
1331)]
# General form of version 7
>>> # L = [(f(v1), f(v2))   for v1 in it1  for v2 in it2]
>>> # L = [(f1(v1), f2(v2))   for v1 in it1  for v2 in it2]
>>> # Version 8 Example
>>> L = [(x**2, y**3) for x in range(1, 10) if x % 2 == 0 for y in range(1, 10) if
y % 2 == 1]
>>> L
[(4, 1), (4, 27), (4, 125), (4, 343), (4, 729), (16, 1), (16, 27), (16, 125), (16,
343), (16, 729), (36, 1), (36, 27), (36, 125), (36, 343), (36, 729), (64, 1), (64,
27), (64, 125), (64, 343), (64, 729)]
>>> # General format of version 8
>>> # L = [(f1(v1), f2(v2)) for v1 in it1 if cond_1(v1) for v2 in it2 if
cond_2(v2)]
>>> # Version 9
>>> L = [(x**2 + y**3) for x in range(1, 10) if x % 2 == 0 for y in range(1, 10) if
y % 2 == 1]
>>> L
[5, 31, 129, 347, 733, 17, 43, 141, 359, 745, 37, 63, 161, 379, 765, 65, 91, 189,
407, 793]
>>> L = [(x**2 + y**3) for x in range(1, 10) if x % 2 == 0 for y in range(1, 10) if
y % 2 == 1
...      if (x**2 + y**3) > 300]
```

```
>>> L
[347, 733, 359, 745, 379, 765, 407, 793]
>>> # General form of version 9
>>> L = [h(f(x), g(y)) for x in it1 if C1(x) for y in it2 if C2(y)]
Traceback (most recent call last):
  File "<pyshell#79>", line 1, in <module>
    L = [h(f(x), g(y)) for x in it1 if C1(x) for y in it2 if C2(y)]
NameError: name 'it1' is not defined
>>> # L = [h(f(x), g(y)) for x in it1 if C1(x) for y in it2 if C2(y)]'
>>> # MasterCondition
>>> L = [(x**2 + y**3) for x in range(1, 10) if x % 2 == 0 for y in range(1, 10) if
y % 2 == 1
...      if (x**2 + y**3) > 300]
>>> # General form
>>> # L = [h(f(x), g(y)) for x in it1 if C1(x) for y in it2 if C2(y) if
Cond(h(f(x), g(y)))]
```