

Looping Statement - II

Session Objectives

- ✓ Understand what looping statements are
- ✓ Understand what a for loop is
- ✓ Shorthand for loop (List Comprehension)
- ✓ Understand is nested for/while loop
- ✓ Understand what a while loop is

Syntax : Shorthand for loop (List Comprehension)

```
new_list = [expression for item in iterable (optional condition)]
or
new_list = [expression1 if condition else expression2 for item in iterable]
```

```
# Looping with Enumerate() -> index[position of an element] , value = enumerate()
car_list = ['Taigun','Slavia','Verna','Thar','Innova','Defender','Lord Alto',
            'Safari','Harrier','Bolero','XUV700','Altroz']
for index , car_name in enumerate(car_list):
    print(index, car_name)
```

```
0 Taigun
1 Slavia
2 Verna
3 Thar
4 Innova
5 Defender
6 Lord Alto
7 Safari
8 Harrier
9 Bolero
10 XUV700
11 Altroz
```

```
car_list = ['Taigun','Slavia','Verna','Thar','Innova','Defender','Lord Alto',
            'Safari','Harrier','Bolero','XUV700','Altroz']
for index , car_name in enumerate(car_list): # (idx , value)
    print(car_list[index], end = " ")
```

```
Taigun Slavia Verna Thar Innova Defender Lord Alto Safari Harrier Bolero XUV700 Altroz
```

```
# enumerates provides both the index and the values of an iterable.
car_list = ['Taigun','Slavia','Verna','Thar','Innova','Defender','Lord Alto',
            'Safari','Harrier','Bolero','XUV700','Altroz']
for index , car in enumerate(car_list):
    print(car, end = " ")
```

```
Taigun Slavia Verna Thar Innova Defender Lord Alto Safari Harrier Bolero XUV700 Altroz
```

```

_dict = {
    'name' : 'Aman',
    'age' : 29,
    'gender' : 'M',
    'city' : 'Delhi',
    'Country' : 'India'
}
_dict.keys()
dict_keys(['name', 'age', 'gender', 'city', 'Country'])
_dict.values()
dict_values(['Aman', 29, 'M', 'Delhi', 'India'])
_dict.items() # (key,value)pair
dict_items([('name', 'Aman'), ('age', 29), ('gender', 'M'), ('city', 'Delhi'), ('Country', 'India')])

```

```

for key , value in _dict.items() :
    print((key,value))

('name', 'Aman')
('age', 29)
('gender', 'M')
('city', 'Delhi')
('Country', 'India')

# Else in For Loops :
for x in range(1,11):
    print(x , end = " ") # [1,2...10]
else:
    print()
    print("Loop Ended Successfully!")

1 2 3 4 5 6 7 8 9 10
Loop Ended Successfully!

```

```

# Break statement -> tries to break the loop abruptly
for x in range(1,11):
    print(x , end = " ") # [1,2...10]
    if x == 7 :
        break
else:
    print()
    print("Loop Ended Successfully!")

1 2 3 4 5 6 7

# Print all the factors of 'n':
n = int(input("Enter the value of 'n' to calculate its factors : "))
for i in range(2,n):
    if n % i == 0:
        print(i , end = " ")

Enter the value of 'n' to calculate its factors : 12
2 3 4 6

```

```
# Print all the factors of 'n':
n = int(input("Enter the value of 'n' to calculate its factors : "))
for i in range(2,n):
    if n % i == 0:
        print(i , end = " ")
```

```
Enter the value of 'n' to calculate its factors : 87
3 29
```

```
# Print all the factors of 'n':
n = int(input("Enter the value of 'n' to calculate its factors : "))
if(n < 0): # negative
    for i in range(n,0):
        if n % i == 0:
            print(i , end = " ")
else: # positive
    for i in range(2,n):
        if n % i == 0:
            print(i , end = " ")
```

```
Enter the value of 'n' to calculate its factors : -11
-11 -1
```

```
# Print all the factors of 'n':
n = int(input("Enter the value of 'n' to calculate its factors : "))
if(n < 0): # negative
    for i in range(n,0):
        if n % i == 0:
            print(i , end = " ")
else: # positive
    for i in range(2,n):
        if n % i == 0:
            print(i , end = " ")
```

```
Enter the value of 'n' to calculate its factors : -16
-16 -8 -4 -2 -1
```

```
new_list = [expression for item in iterable (optional condition)]
or
new_list = [expression1 if condition else expression2 for item in iterable]
```



```
_list = [55,99,121,91,999,-77,False,1221,77,29,100,1,5]
max_val = float('-inf')
for val in _list:
    if val > max_val:
        max_val = val

print(max_val) # 1221
```

1221

```
# walrus operator = ":" [Assigning value]
# [expression1 if condition else expression2 for item in iterable]
num_list = [55,99,121,91,999,-77,False,1221,77,29,100,1,5]
max_number = num_list[0]
[max_number := x if x > max_number else max_number for x in num_list]
max_number
```

1221

```
# new_list = [expression for item in iterable (optional condition)]
num_list = [55,99,121,91,999,-77,False,1221,77,29,100,1,5]
squared = [x ** 2 for x in num_list]
squared
```

[3025, 9801, 14641, 8281, 998001, 5929, 0, 1490841, 5929, 841, 10000, 1, 25]

```
num_list = [55,99,121,91,999,-77,False,1221,77,29,100,1,5]
add_10 = [x + 10 for x in num_list]
add_10
```

[65, 109, 131, 101, 1009, -67, 10, 1231, 87, 39, 110, 11, 15]

```
# new_list = [expression for item in iterable (optional condition)]
num_list = [55,99,121,91,999,-77,False,1221,77,29,100,1,5]
even_list = [x for x in num_list if (x % 2 == 0)]
even_list
```

[False, 100]

```
# new_list = [expression for item in iterable (optional condition)]
num_list = [55,99,121,91,999,-77,False,1221,77,29,100,1,5]
odd_list = [odd for odd in num_list if (odd % 2 == 1)]
odd_list
```

[55, 99, 121, 91, 999, -77, 1221, 77, 29, 1, 5]

```
# Replaces the elements conditionally
# [expression1 if condition else expression2 for item in iterable]
car_list = ['Taigun','Slavia','Verna','Thar','Innova','Defender','Lord Alto',
            'Safari','Harrier','Bolero','XUV700','Altroz']
new_car_list = [car if car != 'Harrier' else 'Curvv' for car in car_list]
print(new_car_list, end = " ")
```

['Taigun', 'Slavia', 'Verna', 'Thar', 'Innova', 'Defender', 'Lord Alto', 'Safari', 'Curvv', 'Bolero', 'XUV700', 'Altroz']

```

car_list = ['Taigun', 'Slavia', 'Verna', 'Thar', 'Innova', 'Defender', 'Lord Alto',
            'Safari', 'Harrier', 'Bolero', 'XUV700', 'Altroz']
new_car_list = []
for car in car_list:
    if car != "Harrier":
        new_car_list.append(car)
    else :
        new_car_list.append("Curvv")
print(new_car_list, end = " ")

['Taigun', 'Slavia', 'Verna', 'Thar', 'Innova', 'Defender', 'Lord Alto', 'Safari', 'Curvv', 'Bolero', 'XUV700', 'Altroz']

```

```

# Multiple List in Comprehension:
# Syntax : [exp1 for item1 in itr1 for item2 in itr2.... if conditon]
[(i,j) for i in range(4) for j in range(4)]

[(0, 0),
 (0, 1),
 (0, 2),
 (0, 3),
 (1, 0),
 (1, 1),
 (1, 2),
 (1, 3),
 (2, 0),
 (2, 1),
 (2, 2),
 (2, 3),
 (3, 0),
 (3, 1),
 (3, 2),
 (3, 3)]

```

```

[(i,j) for i in range(4) for j in range(4) if i == j]

[(0, 0), (1, 1), (2, 2), (3, 3)]

nested_list = [[i*j for i in range(4)] for j in range(4)]
nested_list

[[0, 0, 0, 0], [0, 1, 2, 3], [0, 2, 4, 6], [0, 3, 6, 9]]

# Generator Expression -> () instead []
squared_generator = (x ** 2 for x in range(1,6)) # [1,4,9,16,25]
print(squared_generator)

<generator object <genexpr> at 0x00000161B3AED7D0>

tuple(squared_generator)

(1, 4, 9, 16, 25)

list(squared_generator)

[]

```

```

add_10_generator = (x + 10 for x in range(1,6)) # [11,12,13,14,15]
print(add_10_generator)

<generator object <genexpr> at 0x00000161B3AEC380>

list(add_10_generator)

[11, 12, 13, 14, 15]

tuple(add_10_generator)

()

add_10_generator = ((x,x + 10) for x in range(1,6)) # [11,12,13,14,15]
print(add_10_generator)

<generator object <genexpr> at 0x00000161B3AEEE90>

dict(add_10_generator)

{1: 11, 2: 12, 3: 13, 4: 14, 5: 15}

```

```

key_generator = (x for x in range (11,100,11))
value_generator = (x ** 2 for x in range(11,100,11))
dict(zip(key_generator , value_generator))

```

```

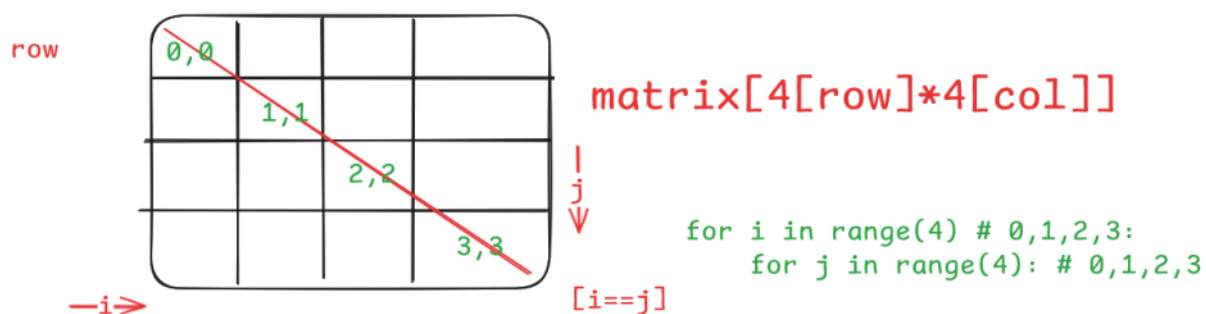
{11: 121,
 22: 484,
 33: 1089,
 44: 1936,
 55: 3025,
 66: 4356,
 77: 5929,
 88: 7744,
 99: 9801}

```

```

for outer_item in outer_sequence:
    # Outer loop Body
    for inner_item in inner_sequence:
        # Inner loop Body

```



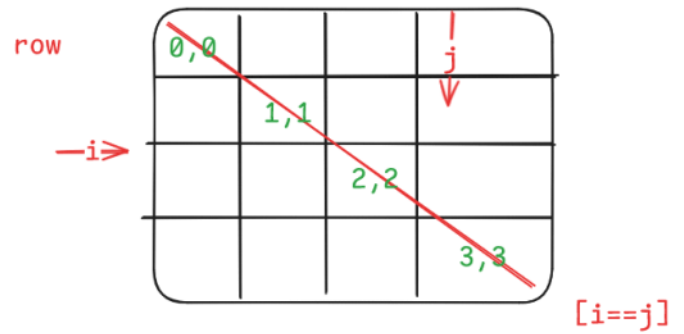
```

i=0 , j=0,1,2,3
i=1 , j=0,1,2,3
i=2 , j=0,1,2,3
i=3 , j=0,1,2,3
i=4 outer loops break

```

```
[(i,j) for i in range(4) for j in range(4)]
```

```
[(0, 0),
(0, 1),
(0, 2),
(0, 3),
(1, 0),
(1, 1),
(1, 2),
(1, 3),
(2, 0),
(2, 1),
(2, 2),
(2, 3),
(3, 0),
(3, 1),
(3, 2),
(3, 3)]
```



```
nested_list = [[i*j for i in range(4)] for j in range(4)]
nested_list
```

```
[[0, 0, 0, 0], [0, 1, 2, 3], [0, 2, 4, 6], [0, 3, 6, 9]]
```

		j			
		j			
		0	1	2	3
i	0	0	0	0	0
1	0	1	2	3	
2	0	2	4	6	
3	0	3	6	9	

$i = [0,1,2,3]$

$j = [0,1,2,3]$

```
for i in range(11,12): # i = [11]
    for j in range (1,11): # j = [1,2,3....10]
        print(f"{i} X {j} = {i*j}")
```

```
11 X 1 = 11
11 X 2 = 22
11 X 3 = 33
11 X 4 = 44
11 X 5 = 55
11 X 6 = 66
11 X 7 = 77
11 X 8 = 88
11 X 9 = 99
11 X 10 = 110
```



```

for i in range(11,21): # i = [11,...20]
    # Outer Loop [i]
    print(f"Mutliplication Table for value: {i} -> ")
    for j in range (1,11): # j = [1,2,3....10]
        print(f"{i} X {j} = {i*j}" , end = "\n")
    print()

```

Mutliplication Table for value: 11 ->

```

11 X 1 = 11
11 X 2 = 22
11 X 3 = 33
11 X 4 = 44
11 X 5 = 55
11 X 6 = 66
11 X 7 = 77
11 X 8 = 88
11 X 9 = 99
11 X 10 = 110

```

Mutliplication Table for value: 12 ->

```

12 X 1 = 12
12 X 2 = 24
12 X 3 = 36
12 X 4 = 48

```

```
[(i,j) for i in range(4) for j in range(4)]
```

```

[(0, 0),
 (0, 1),
 (0, 2),
 (0, 3),
 (1, 0),
 (1, 1),
 (1, 2),
 (1, 3),
 (2, 0),
 (2, 1),
 (2, 2),
 (2, 3),
 (3, 0),
 (3, 1),
 (3, 2),
 (3, 3)]

```

```

for i in range(4):
    for j in range(4):
        print((i,j) , end = "")

```

```
(0, 0)(0, 1)(0, 2)(0, 3)(1, 0)(1, 1)(1, 2)(1, 3)(2, 0)(2, 1)(2, 2)(2, 3)(3, 0)(3, 1)(3, 2)(3, 3)
```

```

car_list = [
    ['Taigun','Slavia','Verna'],
    ['Thar','Innova','Defender'],
    ['Lord Alto','Safari','Harrier'],
    ['Bolero','XUV700','Altroz']
]
for row in car_list :
    print(row)

```

```

['Taigun', 'Slavia', 'Verna']
['Thar', 'Innova', 'Defender']
['Lord Alto', 'Safari', 'Harrier']
['Bolero', 'XUV700', 'Altroz']

```



```

car_list = [
    ['Taigun', 'Slavia', 'Verna'],
    ['Thar', 'Innova', 'Defender'],
    ['Lord Alto', 'Safari', 'Harrier'],
    ['Bolero', 'XUV700', 'Altroz']
]
for row in car_list :
    for car in row:
        print(car , end = " ")
    print()

```

```

Taigun Slavia Verna
Thar Innova Defender
Lord Alto Safari Harrier
Bolero XUV700 Altroz

```

```

car_list = [
    ['Taigun', 'Slavia', 'Verna'],
    ['Thar', 'Innova', 'Defender'],
    ['Lord Alto', 'Safari', 'Harrier'],
    ['Bolero', 'XUV700', 'Altroz']
]
flatten_car_list = []
for row in car_list :
    for car in row:
        flatten_car_list.append(car)
flatten_car_list

```

```

['Taigun',
'Slavia',
'Verna',
'Thar',
'Innova',
'Defender',
'Lord Alto',
'Safari',
'Harrier',
'Bolero',
'XUV700',
'Altroz']

```

```

car_list = [
    ['Taigun', 'Slavia', 'Verna'],
    ['Thar', 'Innova', 'Defender'],
    ['Lord Alto', 'Safari', 'Harrier'],
    ['Bolero', 'XUV700', 'Altroz']
]
new_car_list = []
for row in car_list :
    reversed_row = row[::-1]
    new_car_list.append(reversed_row)

new_car_list

```

```

[['Verna', 'Slavia', 'Taigun'],
 ['Defender', 'Innova', 'Thar'],
 ['Harrier', 'Safari', 'Lord Alto'],
 ['Altroz', 'XUV700', 'Bolero']]

```

```

car_list = [
    ['Taigun', 'Slavia', 'Verna'],
    ['Thar', 'Innova', 'Defender'],
    ['Lord Alto', 'Safari', 'Harrier'],
    ['Bolero', 'XUV700', 'Altroz']
]
new_car_list = []
for row in car_list :
    new_car_list.append(row)

new_car_list

```

```

[['Taigun', 'Slavia', 'Verna'],
 ['Thar', 'Innova', 'Defender'],
 ['Lord Alto', 'Safari', 'Harrier'],
 ['Bolero', 'XUV700', 'Altroz']]

```

```

car_list = [
    ['Taigun', 'Slavia', 'Verna'],
    ['Thar', 'Innova', 'Defender'],
    ['Lord Alto', 'Safari', 'Harrier'],
    ['Bolero', 'XUV700', 'Altroz']
]
flatten_car_list = []
for row in car_list :
    for car in row:
        flatten_car_list.append(car)
print(flatten_car_list[::-1])

['Altroz', 'XUV700', 'Bolero', 'Harrier', 'Safari', 'Lord Alto', 'Defender', 'Innova', 'Thar', 'Verna', 'Slavia', 'Taigun']

```