Lecture 06: More List, loop..

Khushnur Binte Jahangir, Lecturer

B.Sc. in Data Science
Department of Computer Science and Engineering
United International University

for loop

```
days = ["Saturday", "Sunday", "Monday"]
for x in days:
  print(x)
```

```
Output:
Saturday
Sunday
Monday
```

break , continue

```
days = ["Saturday", "Sunday", "Monday"]
for x in days:
  if(x=="Monday"):
    break
  print(x)
```

Output:

Saturday
Sunday

```
days = ["Saturday", "Sunday", "Monday"]
for x in days:
  if(x=="Sunday"):
    continue
  print(x)
```

Output:

Saturday
Monday

range() function

```
days = ["Saturday", "Sunday", "Monday"]
for x in range(len(days)):
  print(x)
  print(days[x])
```

Output:

```
0
Saturday
1
Sunday
2
Monday
```

Practice Problem

- Write a Python program to find the sum of all numbers from 1 to 5.

Solution:

```
sum = 0
for i in range(1, 6):
    sum += i
print("The sum of numbers from 1 to 5 is:", sum)
```

Practice Problem

- Write a Python program to print all the even numbers between 1 and 20.

Solution:

```
for i in range(2, 21, 2):
    print(i)
```

Practice Problem

- Write a Python program to print the following series: 1, 3, 5, 7, 9, ...

Solution:

```
n = int(input("Enter the number of terms: "))
for i in range(1, n*2, 2):
    print(i, end=' ')
```

Practice Problem

- Write a Python program to print the following series: 1, 4, 9, 16, 25, ...

```
Solution:

n = int(input("Enter the number of terms: "))
for i in range(1, n+1):
    print(i*i, end=' ')
```

Practice Problem

- Write a Python program to find the factorial of a given number.

```
Solution:

num = int(input("Enter a number: "))
factorial = 1
for i in range(1, num + 1):
    factorial *= i
print("The factorial of", num, "is:",factorial)
```

while loop

```
#Counting from 1 to 5
count = 1
while count <= 5:
    print(count)
    count += 1

#Output:
1
2
3
4
5
```

Iterating until a condition is met

```
while True:
    response = input("Enter 'quit' to exit: ")
    if response.lower() == 'quit':
        break
    print("You entered:", response)
```

Iterating until a condition is met

```
valid_input = False
while not valid_input:
    user_input = input("Enter a number between 1 and 10: ")
    if user_input.isdigit() and 1 <= int(user_input) <= 10:
        valid_input = True
```

Processing items in a list

```
my_list = [1, 2, 3, 4, 5]
index = 0
while index < len(my_list):
    print(my_list[index])
    index += 1

#Output
1
2
3
4
5
```

Slicing

```
my_list = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

# Sublist from index 2 to 5 (exclusive)
sublist1 = my_list[2:5]
print(sublist1)  # Output: [3, 4, 5]

# Sublist from index 0 to 7 (exclusive), with a step of 2
sublist2 = my_list[0:7:2]
print(sublist2)  # Output: [1, 3, 5, 7]

# Sublist from index 5 to the end of the list
sublist3 = my_list[5:]
print(sublist3)  # Output: [6, 7, 8, 9, 10]

sublist4 = my_list[:-1]
print(sublist4)
```

Nested Loop

```
Outer_loop Expression:
    Inner_loop Expression:
        Statement inside inner_loop
    Statement inside Outer_loop
```

```
for i in range(2):
    print("Printed by Outer loop: ")
    print(i)
    for j in range(50,52):
        print("Printed by inner loop: ")
        print(j)
```

```
Printed by Outer loop:
0
Printed by inner loop:
50
Printed by inner loop:
51
Printed by Outer loop:
1
Printed by inner loop:
50
Printed by inner loop:
51
```

Tracing Table

```
# outer loop
for i in range(1, 3):
    for j in range(1, 3):
        # print multiplication
        print(i * j, end=' ')
    print()
```

#Output
1 2
2 4

Iteration	i	j	Output
initial	-	-	-
1	1	1	1
	1	2	2
2	2	1	2
	2	2	4

Nested Loop Example

```
# outer loop
for i in range(1, 11):
    # nested loop
    # to iterate from 1 to 10
    for j in range(1, 11):
        # print multiplication
        print(i * j, end=' ')
    print()
```

```
1 2 3 4 5 6 7 8 9 10
2 4 6 8 10 12 14 16 18 20
3 6 9 12 15 18 21 24 27 30
4 8 12 16 20 24 28 32 36 40
5 10 15 20 25 30 35 40 45 50
6 12 18 24 30 36 42 48 54 60
7 14 21 28 35 42 49 56 63 70
8 16 24 32 40 48 56 64 72 80
9 18 27 36 45 54 63 72 81 90
10 20 30 40 50 60 70 80 90 100
```

Nested Loop

- How to print the following pattern?

```
# Define the number of rows
rows = 5
# Outer loop for rows
for i in range(1, rows + 1):
    # Inner loop for columns
    for j in range(1, i + 1):
        print(j, end=" ")
    print()
```

```
#Output:
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
```

Nested Loop

- What will be output of the following code?

```
for i in range(1, 7):
    for j in range(1, i + 1):
        print(j, end=" ")
    print()

for i in range(5, 0, -1):
    for j in range(1, i + 1):
        print(j, end=" ")
    print()
```

```
#Output:

1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
1 2 3 4 5 6
1 2 3 4 5
1 2 3 4
1 2 3
1 2
1
```

Nested Loop

- What will be the output of the following code?

```
x = [1, 2]
y = [3, 4]

for i in x:
  for j in y:
    print(i, j)
```

```
#Output:
1 3
1 4
2 3
2 4
```

Iterating 2D list

```
matrix = [[1, 2, 3],[4, 5, 6],[7, 8, 9]]

for row in matrix:
    for element in row:
        print(element,end=" ")
    print()
```

```
#Output:
1 2 3
4 5 6
7 8 9
```

Accessing 2D list Elements By Indices

```
matrix = [[1, 2, 3],[4, 5, 6],[7, 8, 9]]

for i in range(len(matrix)):
    for j in range(len(matrix[i])):
        print(matrix[i][j],end=" ")
    print()
```

```
#Output:
1 2 3
4 5 6
7 8 9
```

Break and Continue statement in Nested Loops

```
for i in range(2, 4):
    for j in range(1, 11):
      if i==j:
        break
      print(i, "*", j, "=", i*j)
```

```
#Output:

2 * 1 = 2
3 * 1 = 3
3 * 2 = 6
```

```
for i in range(2, 3):
    for j in range(1, 11):
      if i==j:
        continue
      print(i, "*", j, "=", i*j)
```

```
#Output
2 * 1 = 2
2 * 3 = 6
2 * 4 = 8
2 * 5 = 10
2 * 6 = 12
2 * 7 = 14
2 * 8 = 16
2 * 9 = 18
2 * 10 = 20
```

Nested Loop Example

```
my_strings = ["hello", "world", "python"]

for string in my_strings:
    for char in string:
        print(char)
```

Nested Loop Example

```
my_strings = ["hello", "world", "python"]
vowels = "aeiou"

for string in my_strings:
    vowel_count = 0
    for char in string:
        if char in vowels:
            vowel_count += 1
    print(f"Number of vowels in '{string}': {vowel_count}")
```

```
#Output
Number of vowels in 'hello': 2
Number of vowels in 'world': 1
Number of vowels in 'python': 1
```

Nested Loop Example

```
sum = 10
for i in range(1, 4):
    sum = sum + 10
    for j in range(1, 4):
        sum = sum + 20
        for k in range(1, 4):
            sum = sum + 30
print("Total sum:" , sum)
```

```
Total sum:
1030
```