1. For loop

In [1]:
```
nums = [1, 2, 3, 4, 5]

for num in nums:
    print(num)
```

```
1
2
3
4
5
```

1.1 break

The break statement terminates the loop containing it. Control of the program flows to the statement immediately after the

In [2]:
```
# Break for loop under condition
for num in nums:
    if num == 3:
        print('Found!')
        break
    print(num)
```

```
1
2
Found!
```

1.2 continue

The continue statement is used to skip the rest of the code inside a loop for the current interation only. Loop does not terminate but continues on with the next iteration.

In [3]:
```
for num in nums:
    if num == 3:
        print('Found!')
        continue
    print(num)
```

```
1
2
Found!
4
5
```

1.3 Nested loop

A nested loop is a loop within a loop, an inner loop within the body of an outer one. It gives a combination of two elements

In [4]:
```
for num in nums:
    for letter in 'abc':
        print(num, letter)
```

```
1 a
1 b
1 c
2 a
2 b
2 c
3 a
3 b
3 c
4 a
4 b
4 c
5 a
5 b
5 c
```

1.4 range function in loop

We can iterate through the integers using range function with loop.

In [5]:
```
for i in range(1, 11):
    print(i)
```

```
1
2
3
4
5
6
7
8
9
10
```

2. While loop

While loop iterates until certain conditions are evaluated as False.

In [6]:
```
x = 0

while x < 10:
    print(x)
    x += 1
```

```
0
1
2
3
4
5
6
7
8
9
```

In [ ]:
```

```