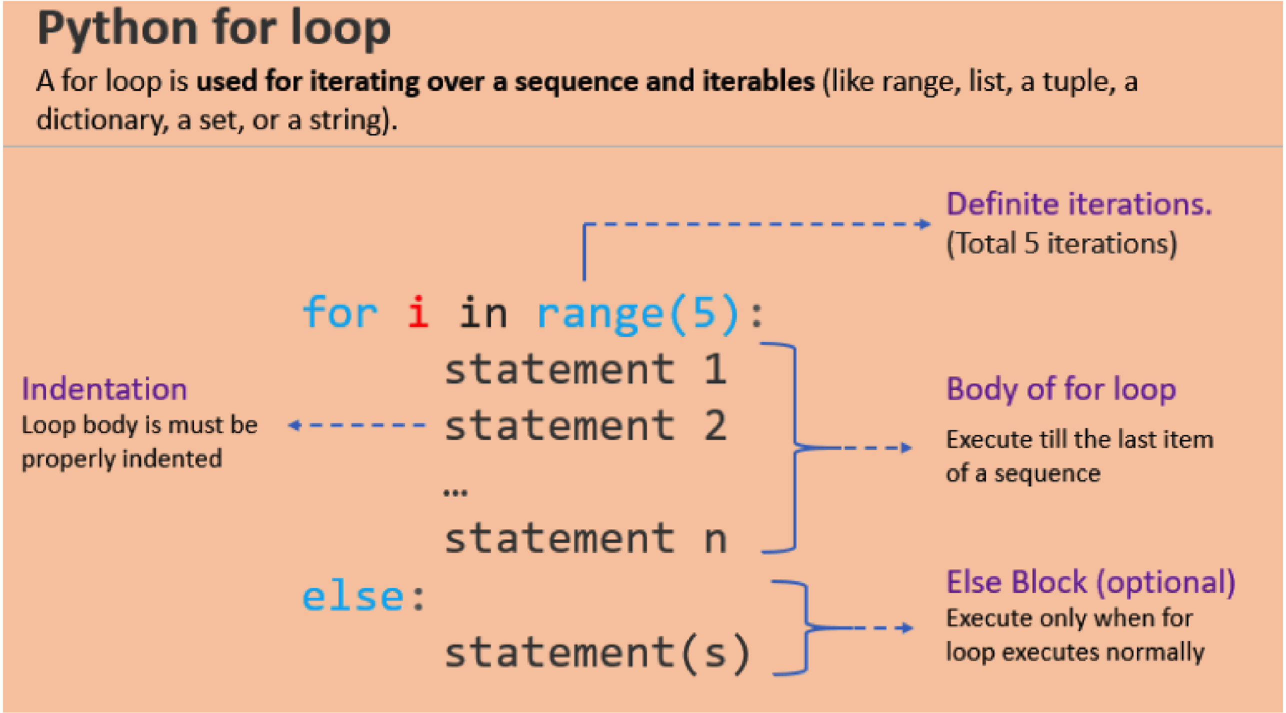
What is for loop in Python

In Python, the for loop is used to iterate over a sequence such as a list, string, tuple, other iterable objects such as range.

With the help of for loop, we can iterate over each item present in the sequence and executes the same set of operations for each item. Using a for loops in Python we can automate and repeat tasks in an efficient manner.

So the bottom line is using the for loop we can repeat the block of statements a fixed number of times. Let's understand this with an example.

for loops are executed a fixed number of times, you need to know how many times to repeat the code.



Python for loop

A for loop is **used for iterating over a sequence and iterables** (like range, list, a tuple, a dictionary, a set, or a string).

```
for i in range(5):
        statement 1
        statement 2
        ...
        statement n
else:
        statement(s)
```

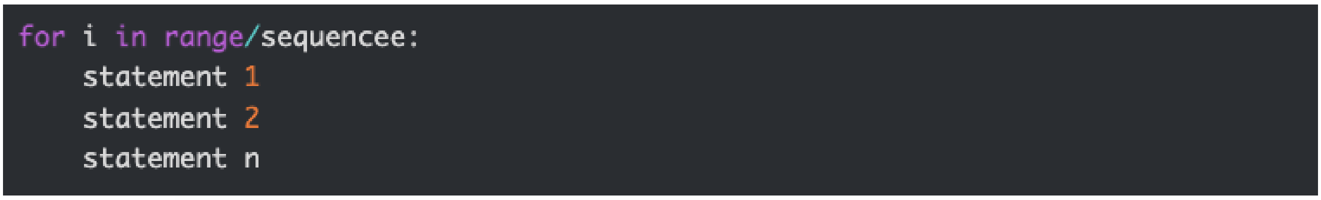
Definite iterations.
(Total 5 iterations)

Indentation
Loop body is must be properly indented

Body of for loop
Execute till the last item of a sequence

Else Block (optional)
Execute only when for loop executes normally

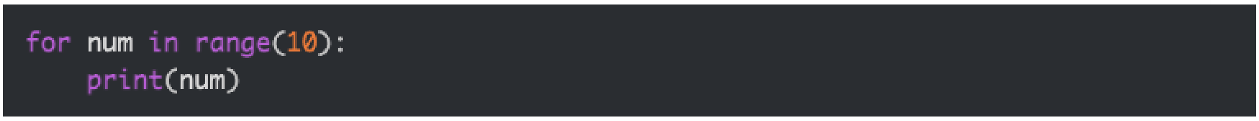
Syntax of `for` loop



```
for i in range/sequencee:
    statement 1
    statement 2
    statement n
```

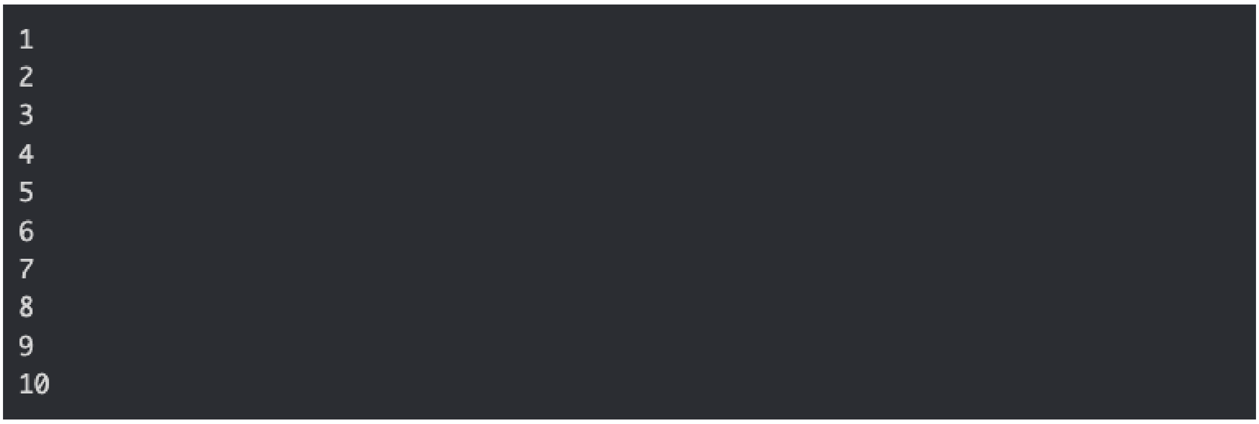
- in the syntax, `i` is the iterating variable, and the range specifies how many times the loop should run. For example, if a list contains 10 numbers then for loop will execute 10 times to print each number.
- In each iteration of the loop, the variable `i` get the current value.

Example: Print first 10 numbers using a for loop



```
for num in range(10):
    print(num)
```

Output

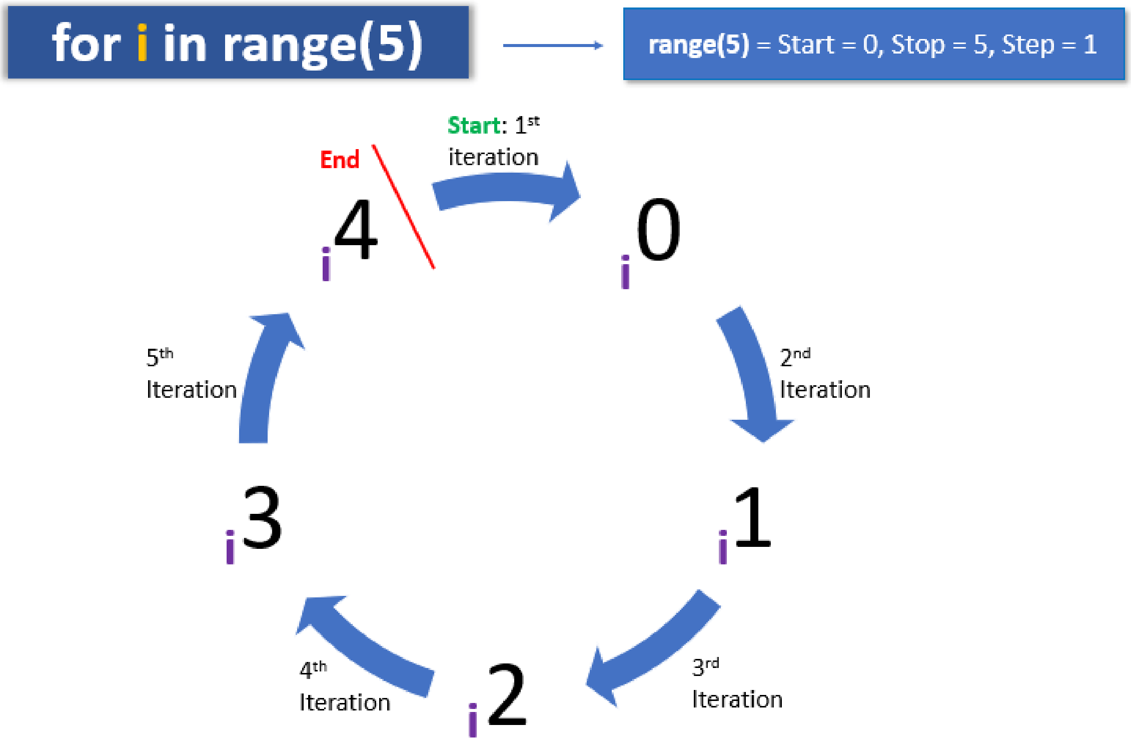


```
1
2
3
4
5
6
7
8
9
10
```

`for` loop with `range()`

The range() function returns a sequence of numbers starting from 0 (by default) if the initial limit is not specified and it increments by 1 (by default) until a final limit is reached.

The `range()` function is used with a loop to specify the range (how many times) the code block will be executed. Let us see with an example.



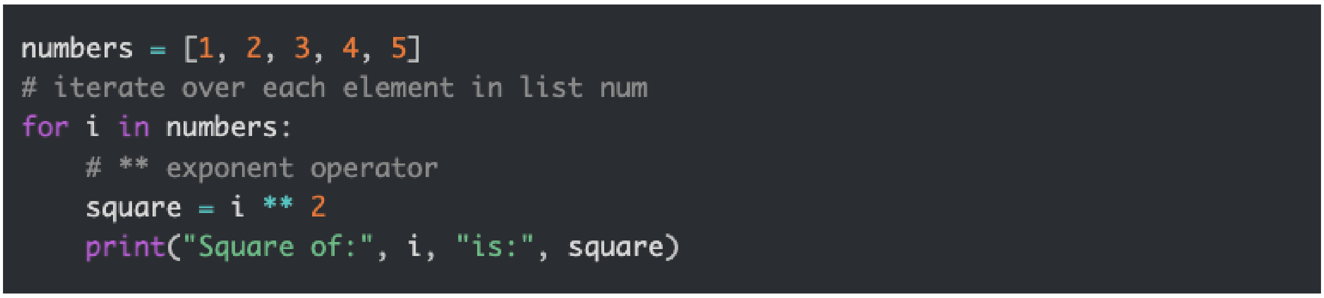
How for loop works

The for loop is the easiest way to perform the same actions repeatedly. For example, you want to calculate the square of each number present in the list.

Write for loop to iterate a list, In each iteration, it will get the next number from a list, and inside the body of a loop, you can write the code to calculate the square of the current number.

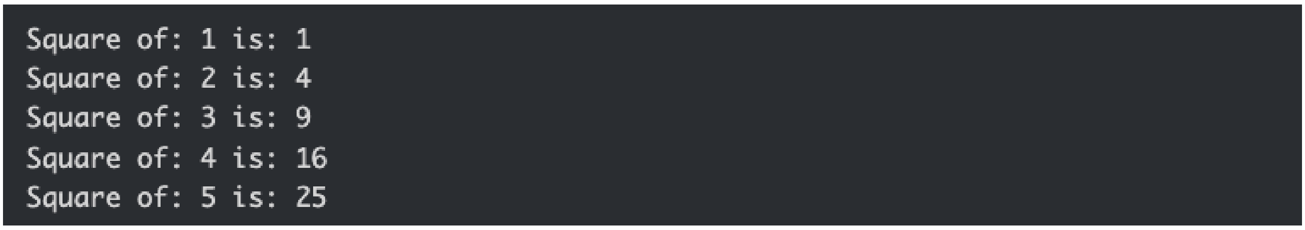
Example: Calculate the square of each number of list

Python list is an ordered sequence of items. Assume you have a list of 10 numbers. Let's see how to want to calculate the square of each number using for loop.



```
numbers = [1, 2, 3, 4, 5]
# iterate over each element in list num
for i in numbers:
    # ** exponent operator
    square = i ** 2
    print("Square of:", i, "is:", square)
```

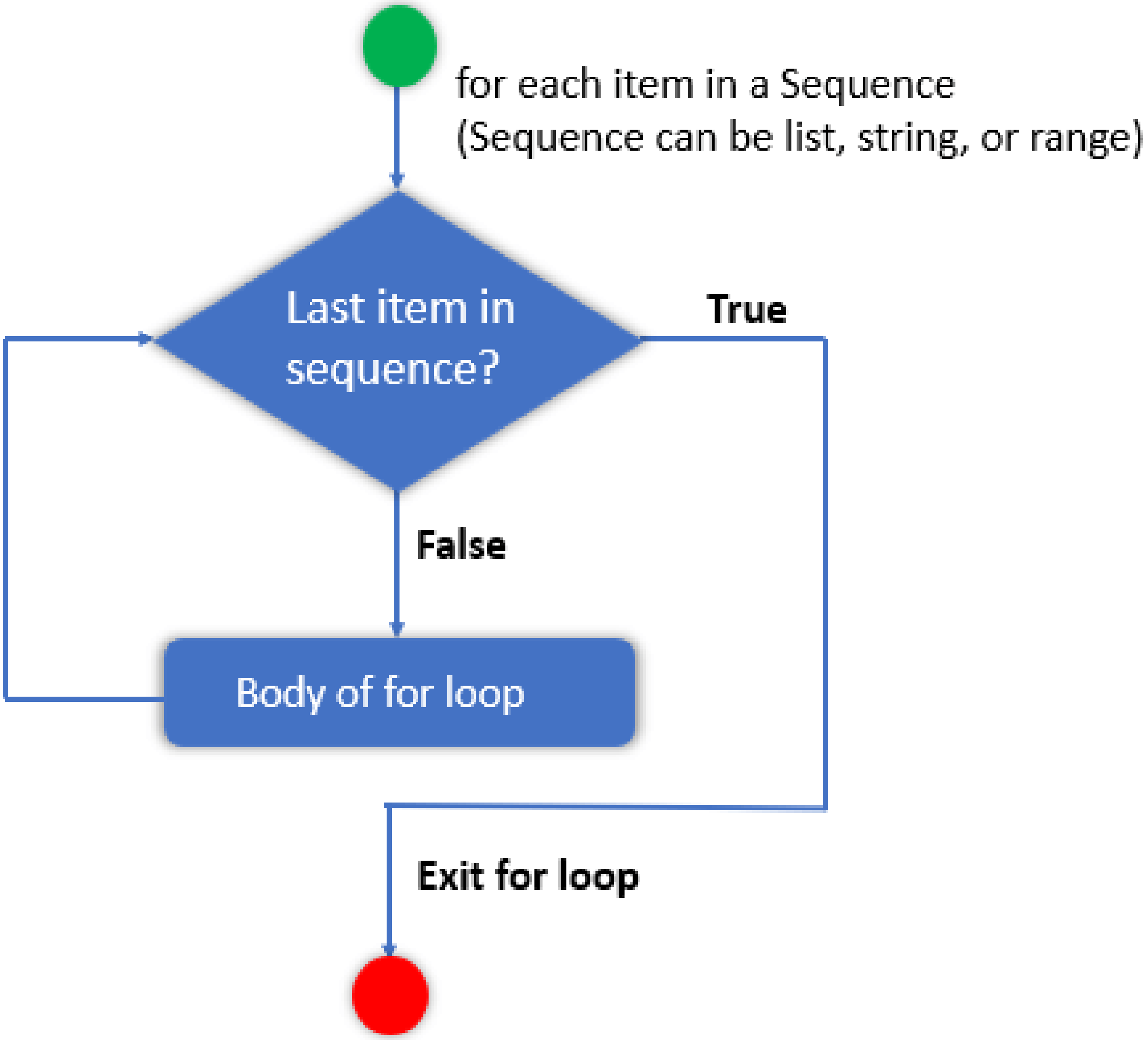
Output



```
Square of: 1 is: 1
Square of: 2 is: 4
Square of: 3 is: 9
Square of: 4 is: 16
Square of: 5 is: 25
```

Note:

The loop runs till it reaches the last element in the sequence. If it reaches the last element in the sequence, it exits the loop. otherwise, it keeps on executing the statements present under the loop's body


for each item in a Sequence
(Sequence can be list, string, or range)
Last item in sequence?
True
False
Body of for loop
Exit for loop

Why use for loop?

Let's see the use for loop in Python.

- **Definite Iteration**: When we know how many times we wanted to run a loop, then we use count-controlled loops such as for loops. It is also known as definite iteration. For example, Calculate the percentage of 50 students. here we know we need to iterate a loop 50 times (1 iteration for each student).
- **Reduces the code's complexity**: Loop repeats a specific block of code a fixed number of times. It reduces the repetition of lines of code, thus reducing the complexity of the code. Using for loops and while loops we can automate and repeat tasks in an efficient manner.
- **Loop through sequences**: used for iterating over lists, strings, tuples, dictionaries, etc., and perform various operations on it, based on the conditions specified by the user.