**For Loop**

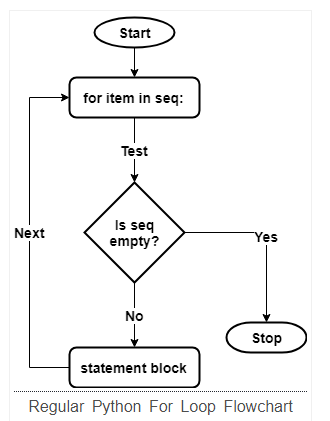
* Control flow statement
* Preferred when we know the total number of iteration.
* Required at least two variable to work
* Iterable object such as a list, tuple or a string.
* Variable to store the successive values from the sequence in the loop

**Syntax**:

for iter in sequence:

statements(iter)

Workflow:



**Example:**

vowels="AEIOU"

for iter in vowels:

print("char:", iter)

**Find the Average Of N Numbers:**

int\_list = [1, 2, 3, 4, 5, 6]

sum = 0

for iter in int\_list:

sum += iter

print("Sum =", sum)

print("Avg =", sum/len(int\_list))

**Range with For loop:**

* **range()** function can produce an integer sequence at runtime.
* >>> type(range(0, 10))
* <class 'range'>
* >>> range(0, 10)[0]
* 0
* >>> range(0, 10)[1]
* 1
* >>> range(0, 10)[9]
* 9
* >>> len(range(0, 10))
* 10
* >>>for iter in range(0, 3):
* print("iter: %d" % (iter))

**Else Clause with for loop**

The code under the else clause executes after the completion of the **“for”** loop. However, if the loop stops due to a **“break”** call, then it’ll skip the **“else”** clause.

# Foe-Else Syntax

for item in seq:

statement 1

statement 2

if <cond>:

break

else:

statements

Example:

birds = ['Belle', 'Coco', 'Juniper', 'Lilly', 'Snow']

ignoreElse = False

for theBird in birds:

print(theBird )

if ignoreElse and theBird is 'Snow':

break

else:

print("No birds left.")

