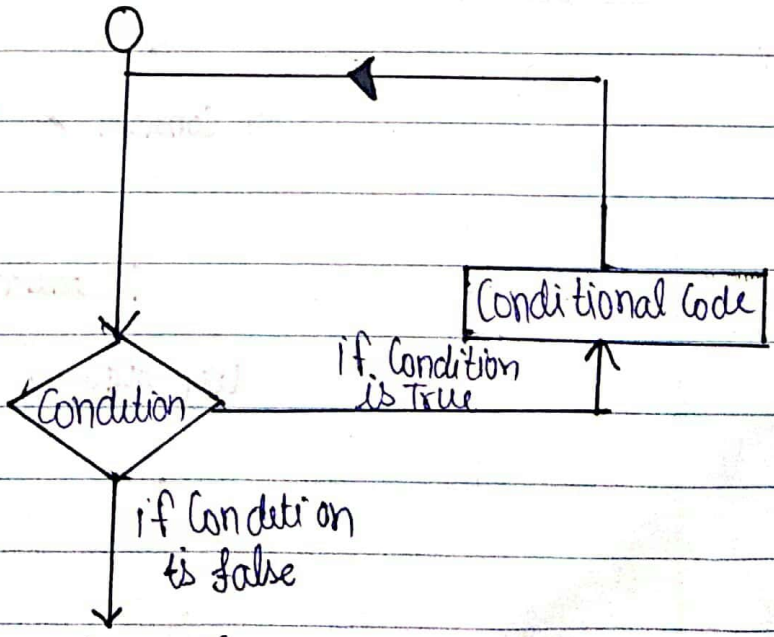
# Python Loops

- A loop statement allows us to execute a statement or group of statement multiple times.

- Python Programming Language Provide following types of loops to handle Programming language.

- There are three ways for executing the loops.

1. while loop
2. for loop
3. nested loop




(Diagram to illustrates a loop statement).

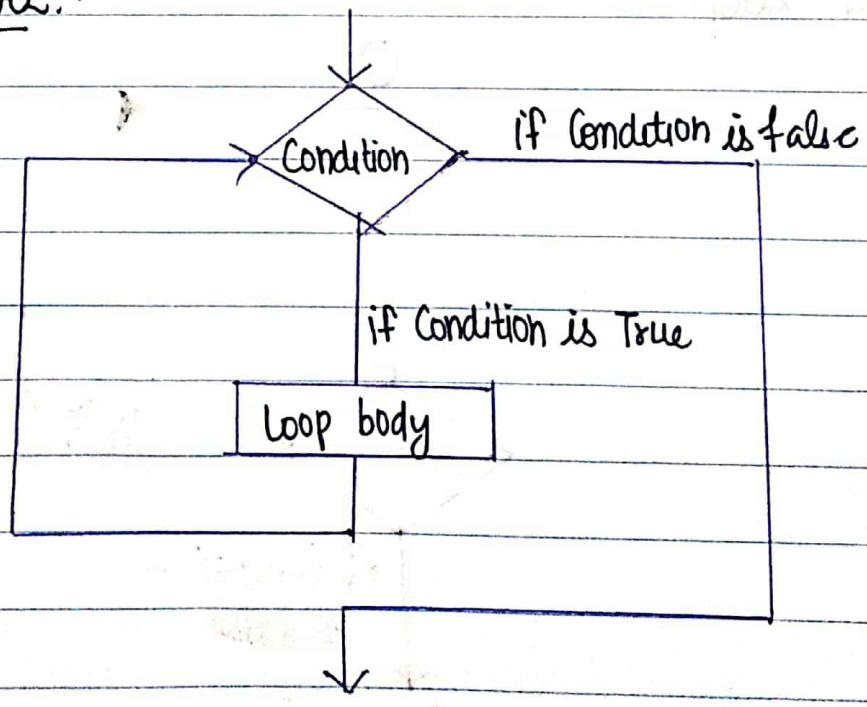
# 1. Python While loop

- The Python while loop allows a part of code to be executed until the given condition returns false.
- It is also known as pre-tested loop.
- It can be viewed as repeating if statement.

## Syntax

```
while expression:
    statement
```

## Flowchart:-



example

```
count = 0
while (count < 3):
    count = count + 1
    print(count)
```

Output

1
2
3



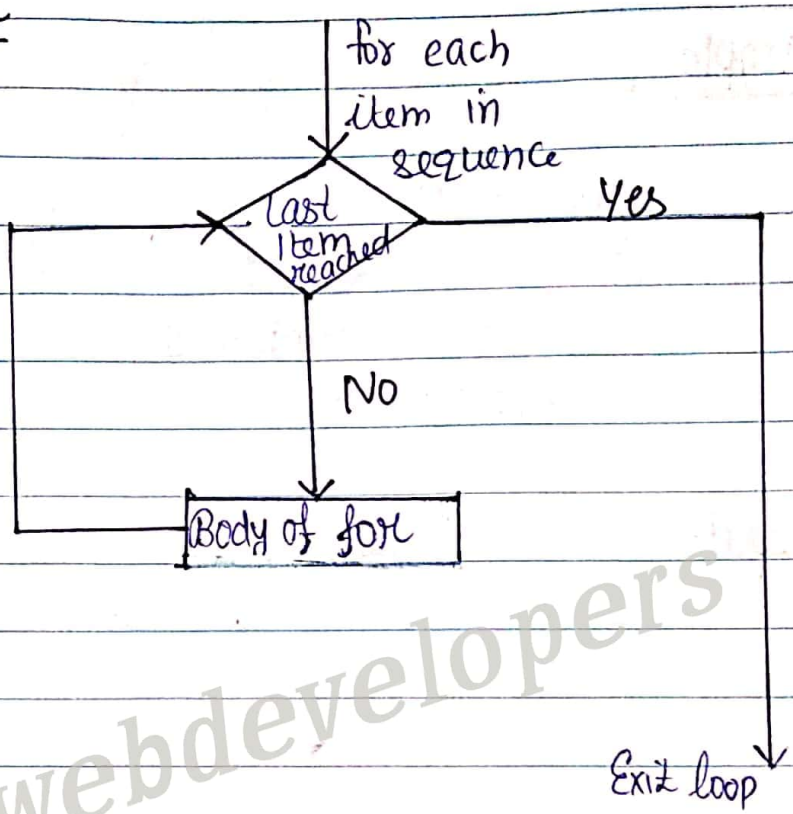
## 2 Python for Loop :-

- A `for` loop is used for iterating over a sequence (that is either a list, a tuple, a dictionary, a set or a string).

Syntax :-

```
for var in sequence :
    statement
```

## Flowchart






## Example

```
st = "jpwebdevelopers"
for ch in st:
    print(ch)
```

output :::

```
j
p
w
e
b
d
e
v
e
l
o
p
e
r
s
```

Example II (for loop in Range)

```
a = range(5)
for i in a:
    print(i)
```

Output:-

0
1
2
3
4



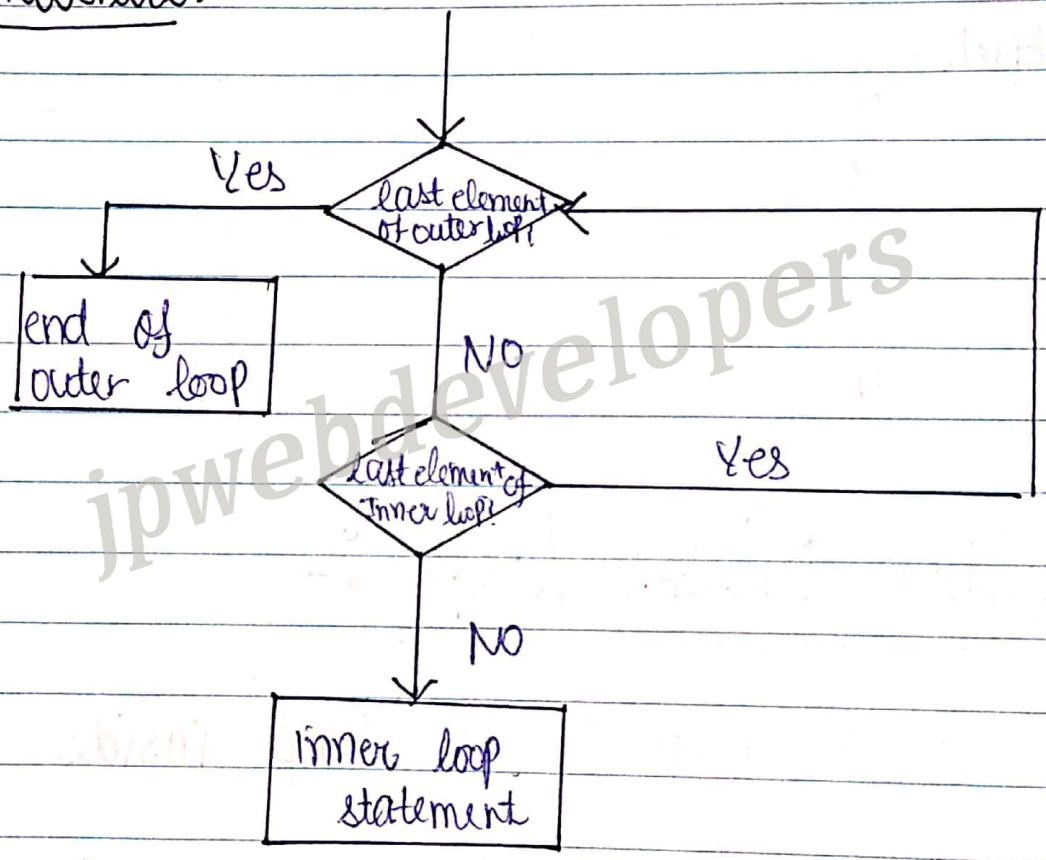
## 3. Python Nested Loops :-

- A nested loop is a "loop inside a loop."
- The "inner loop" will be executed one time for each iteration of the "outer loop":
- Python allows to use one loop inside another loop.

Syntax:-

for iterator_var in sequence:
    for iterator_var in sequence:
        statement(s)
        statement(s)

flowchart:-






Example1

```
adj = ["red", "big", "tasty"]
fruits = ["mango", "Apple", "cherry"]

for x in adj:
    for y in fruits:
        print(x,y)
```

Output:-

red mango
red apple
red cherry
big mango
big apple
big cherry
tasty mango
tasty apply
tasty chery