(iii) Nested loop

A nested loop is a loop within a loop.

The inner loop will iterate through its entire cycle for each iteration of the outer loop.

- For loop inside for loop
- For loop inside while loop.
- while loop inside for loop
- while loop inside while loop.

* For loop inside for loop

Syntax: for variable (1) in range (): # outer loop.
for variable 2 in range (): Inner Loop.
Statement of inner loop.
Statement of outer loop.

Eg: multiplication table from 1 to 10.

```
for i in range (1, 11, 1):
    for j in range (1, 11, 1):
        print (i*j, end=" ")
    print ()
```

* while loop inside while loop

Syntax:
Initialization of outer loop.
while condition: outer loop.
Initialization of inner loop.
while condition: inner loop.
Statements of inner Loop.
inc/dec of inner Loop
Statements of outer loop.
inc/dec of outer loop.

Example. multiplication table from 1 to 10.

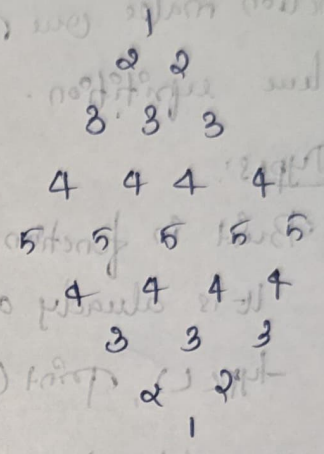
```
i=1
while i<=10:
    j=1
    while j<=10:
        print (i*j, end=" ")
        j=j+1
    print ()
    i=i+1
```

* ~~While~~ For inside a ~~while~~ loop.

```
For variable in range(): outer loop
    initialization.
    while (condition): inner loop
        Statement of inner loop
        inc/dec of while loop
    Statement of outer loop
```

Example

```
for i in range(1,6,1):
    j=1
    while (j<=5):
        print(" ", end=" ")
        j=j+1
    print(i, end=" ")
k=i
y=1
while y<=i:
    print(k, end=" ")
    y=y+1
print()
```

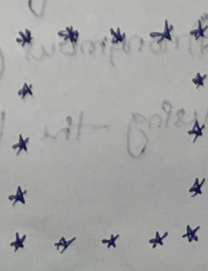


* For ~~while~~ inside a ~~for~~ while loop

```
Initialization of while loop.
while (condition): outer loop.
    for variable in range():
        Statement of inner loop.
    Statement of outer loop.
    inc/dec of while loop.
```

Example

```
i=1
while (i<=5):
    for j in range(1,6,1):
```



print("*", end=" ")

else:
print(" ", end=" ")

print()
p.pi+1

* FUNCTIONS

A function is a block of code that performs a specific task.

Function make our program more organized, readable, and reduce repitition.

→ Types:

① Built in function:
It is already available in python.
type(), print(), input(), len().

② user defined function:
Functions are created by user using def(define) keyword.

def function_name(parameter): (input)
Statement
return value. [Returns instead of print].

(i) Function without input or without return.
(ii) Function with input or without return.
(iii) Function without input and with return.
(iv) Function with input or with return.

③ Lambda function
Anonymous (Nameless) functions written in single line using the lambda keyword.

1. Function without input and without return:

```
def fun_name():
    Statement.

Call the function.
function_name.
```

Variable inside a function is called local variable.
It can't be used outside the function.

Example: Addition of 2 numbers.

```
def add11():
x = int(input("enter a value"))
y = int(input("enter a value"))
s = x+y
print = (f" sum of {x} and {y} is {s}")

Call the function.
    add11()
```

Output: Give a value = 3
Give a value = 4
Sum of 3 & 4 is 7.

2. Function with input and without return.

```
def fun-name (P1, P2, ... Pn):
    Statement.

(Call the function)
    fun-name.
```

Example (Addition)

```
def add2(x,y):
    s = x+y
    Print(f"the sum of {x} and {y} is {s}")

add12(4,5)
```

Output: Sum of 4 and 5 is 9

3. Function without input and with return:

```
def fun_name():
    Statement
    return value
```

example:

```
def add3():
    x = int(input("enter x value"))
    y = int(input("enter y value"))
    s = x+y
    return x,y,s
add3()
enter x value = 3
enter y value = 4.
Sum = add3()
```

4: Function with input and with return.

```
def fun-name(P1, P2 ... Pn):
    Statements
    return value.
```

example:

```
def add4(x,y):
    s = x+y
    return x,y,s.
add4(2,3)
```

Output:

```
(2,3,5)
x,y,s = add4(2,3).
Print("Sum of {x} and {y} is {s}")
Sum of 2 and 3 is 5.
```

Example

Create a function to check the given number is prime or not using with input and without given method.

```
def prime(n):
for i in range(2,n,1):
    if (n % i == 0):
        print(n," its not prime number")
        break.
    else:
        print(n," its prime number")
prime(14)
```

Output:

14 its prime number.