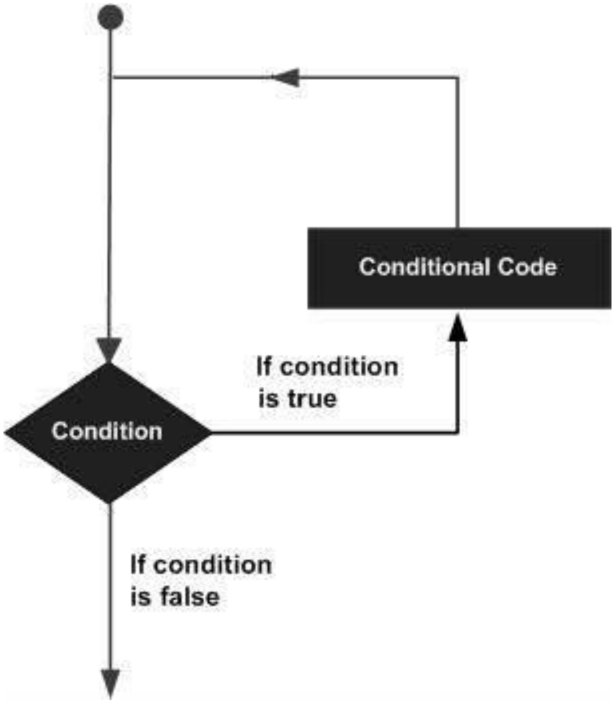
LOOPS:

Program to print hello five times.

Print (“Hello”)
Print (“Hello”)
Print (“Hello”)
Print (“Hello”)
Print (“Hello”)

What if you have to print it for 100 or 1000 times? Loops helps us in doing this.



Loops → repeat some section of code.

Loop will start with the condition, and will come back to the start point, let’s say for 5 times till the condition is fulfilled. It reduces the length of the code.

Two Types of Code:

1. While Loop
2. For Loop

Loops are constructs in programing language, which helps you to repeat some section of code.

For Loop

Range:

Range can be written in 3 ways.

1. range (n) → this means number starting from 0 to n – 1. [0,1,2,3…. n-1]
    a. ex: range (5) – [0, 1, 2, 3, 4]
    b. ex: range (6) – [0, 1, 2, 3, 4, 5]
2. range (start, n) → in this, it will print the value from [start, start + 1, start +2…. n – 1]
    a. ex: range (2, 5) – [2, 3, 4]
    b. ex: range (1, 6) – [1, 2, 3, 4, 5]

3. range (start, n, step) → it will print, [start, start +step, start + 2*step, start + 3*step…. n – 1]
    a. ex: range (2, 10, 2) – [2, 4, 6, 8]
    b. range (1, 12, 4) – [1, 5, 9]
    c. range (3, 9, 2) – [3, 5, 7]
    d. range (10, 20, 3) – [10, 13, 16, 19]
    e. range (15, 2, -1) – [15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3]
    f. range (10, 2, -2) – [10, 8, 6, 4]
    g. range (10, 5, -3) – [10, 7]
    h. range (10, -5, -3) – [10, 7, 4, 1, -2]

MCQ’s

Range (5, 10, 4) **ANS:** 5.9

Range (-10, 0, 2) **ANS:** [-10, -8, -6, -4, -2]

NOTE: range (5, 4, 2) – [ ]

For loop works on range. The syntax for “for loop” is *for i in range (n)*”.

```
Ex: for i in range (5)
        print (i)
```

OUTPUT:

```
[0, 1, 2, 3, 4]
```

The code runs 5 times as the n value is 5.

Ex: 01

INPUT

```
for i in range (2, 10, 2):
        print ("hello!", i)
```

OUTPUT

```
        hello! 2
        hello! 4
        hello! 6
        hello! 8
```

Ex: 02

INPUT

```
for i in range (5):

        print ("hello!", i)
```

OUTPUT

```
        hello! 0
        hello! 1
        hello! 2
        hello! 3
        hello! 4
```

Ex: 03

```
for i in range (2, 10, 2):

        print ("hello!", i)

print ("done")
```

OUTPUT

```
hello! 2
hello! 4
hello! 6
hello! 8
done
```

Since the last print statement is not in the for loop, “done” is printed only once.

Ex: 04 – Prime numbers between 0 – 100

Code 1

```
for i in range (2, 101, 2):

        if (i % 2 == 0):

                print (i,"is an even number")
```

Code 2

```
for i in range (100):
    if (i % 2 == 0):
        print (i,"is an even number")
```

Both give the same output

OUTPUT

```
0 is an even number
2 is an even number
4 is an even number
6 is an even number
8 is an even number
10 is an even number
12 is an even number
14 is an even number
16 is an even number
18 is an even number
20 is an even number
22 is an even number
24 is an even number
26 is an even number
28 is an even number
30 is an even number
32 is an even number
34 is an even number
36 is an even number
38 is an even number
40 is an even number
42 is an even number
44 is an even number
46 is an even number
48 is an even number
50 is an even number
52 is an even number
54 is an even number
56 is an even number
58 is an even number
60 is an even number
62 is an even number
64 is an even number
66 is an even number
68 is an even number
70 is an even number
72 is an even number
74 is an even number
76 is an even number
78 is an even number
80 is an even number
82 is an even number
84 is an even number
86 is an even number
88 is an even number
90 is an even number
92 is an even number
94 is an even number
96 is an even number
98 is an even number
100 is an even number
```

Ex: 05 – Print even number in reverse [100 – 1]

Code 1

```
for i in range (101, 1, -1):
   if (i % 2 == 0):
       print ("even no is ", i)
```

Code 2

```
for i in range (101, 1, -1):
   if (i % 2 == 0):
        print ("even no is ", i)
```

OUTPUT

```
even no is 100
even no is 98
even no is 96
even no is 94
even no is 92
even no is 90
even no is 88
even no is 86
even no is 84
even no is 82
even no is 80
even no is 78
even no is 76
even no is 74
even no is 72
even no is 70
even no is 68
even no is 66
even no is 64
even no is 62
even no is 60
even no is 58
even no is 56
even no is 54
even no is 52
even no is 50
even no is 48
even no is 46
even no is 44
even no is 42
even no is 40
even no is 38
even no is 36
even no is 34
even no is 32
even no is 30
even no is 28
even no is 26
even no is 24
even no is 22
even no is 20
even no is 18
even no is 16
even no is 14
even no is 12
even no is 10
even no is 8
even no is 6
even no is 4
even no is 2
```

Ex: 06

Given n from user print from [1 - n]

is divisible by 3 print *FIZZ*
if any of them is divisible by 5
if divisible by both *FIZZ*_**FUZZ**

else: print normal

no = 15

1 2 *FIZZ* 4 **FUZZ** *FIZZ* 7 8 *FIZZ* **FUZZ** 11 *FIZZ* 13 14 *FIZZ*_**FUZZ**

Code:

```
user_input = int(input("Enter a number: "))

for i in range(1, user_input + 1):
    if i % 5 == 0 and i % 3 == 0:
        print(i, "FIZZ_FUZZ")
    elif i % 5 == 0:
        print(i, "FUZZ")
    elif i % 3 == 0:
        print(i, "FIZZ")
    else:
        print(i)
```

OUTPUT – Enter a number: 15

```
1
2
3 FIZZ
4
5 FUZZ
6 FIZZ
7
8
9 FIZZ
10 FUZZ
11
12 FIZZ
13
14
15 FIZZ_FUZZ
```

Enter a number: 100

```
1
2
3 FIZZ
4
5 FUZZ
6 FIZZ
7
8
9 FIZZ
10 FUZZ
11
12 FIZZ
13
14
15 FIZZ_FUZZ
16
17
18 FIZZ
19
20 FUZZ
21 FIZZ
22
23
24 FIZZ
25 FUZZ
26
27 FIZZ
28
29
30 FIZZ_FUZZ
31
32
33 FIZZ
34
35 FUZZ
36 FIZZ
37
38
39 FIZZ
40 FUZZ
41
42 FIZZ
43
44
45 FIZZ_FUZZ
46
47
48 FIZZ
49
50 FUZZ
51 FIZZ
52
53
54 FIZZ
55 FUZZ
56
57 FIZZ
58
59
60 FIZZ_FUZZ
61
62
63 FIZZ
64
65 FUZZ
66 FIZZ
67
68
69 FIZZ
70 FUZZ
71
72 FIZZ
73
74
75 FIZZ_FUZZ
76
77
78 FIZZ
79
80 FUZZ
81 FIZZ
82
83
84 FIZZ
85 FUZZ
86
87 FIZZ
88
89
90 FIZZ_FUZZ
91
92
93 FIZZ
94
```

95 **FUZZ**

96 *FIZZ*

97

98

99 *FIZZ*

100 **FUZZ**

Explanation:

User gave an input of 15. So, the program will take the range from 1-15.

Now for i = 1, the if statement would be false. It will move to the first elif statement (i % 5 == 0 and i % 3 == 0) and will check the elif condition. For i = 1, i % 5 == 0 is false. even for the second elif statement (i % 3 == 0) the condition is false.

So, the program will go to the else statement and will print (i).

Now for i = 2, the if statement would be false. It will move to the first elif statement (i % 5 == 0 and i % 3 == 0) and will check the elif condition. For i = 2, i % 5 == 0 is false. even for the second elif statement (i % 3 == 0) the condition is false.

So, the program will go to the else statement and will print (i).

Now for i = 3, the if statement would be false. It will move to the first elif statement (i % 5 == 0 and i % 3 == 0), because for i % 3 == 0, yet it is false for I % 5 == 0. False and true is false as both conditions need to be true in ‘and’. Now the program will check the elif condition. For i = 3, i % 5 == 0 is false. But for the second elif condition, (i % 3 == 0) is true.

So, the program will enter the second elif statement and will print “**3 FIZZ**”

NOTE:

Whenever you use print, it by default adds a new line. If you want to suppress the new line there is a syntax.

print (“text”, end=””) → This means after printing it will not add a new line.

Ex:

INPUT:

```
print ("Hello", end=" and Bye")
print ("bye", end="")
print ("yellow", end="")
```

OUTPUT:

```
Hello and Byebyeyellow
```

INPUT:

```
print ("Hello", end=" and Bye")
print ("bye", end="")
print ()
print ("yellow", end="")
```

OUTPUT

Hello and Byebye

Yellow

NESTED LOOPS:

```
for i in range (5):
    for j in range (5):
        print (i, j)
```

The value of i – [0, 1, 2, 3, 4] and the range of j would be the same.

So, for the first value of i = 0, j will have [0, 1, 2, 3, 4]

So, for i=0, j=0. And after printing (0, 0), the program will go to second line and print the (0, 1) and will keep repeating until the value of j = 4.

For the first loop cycle, we will get the values as

(0,0)
(0,1)
(0,2)
(0,3)
(0,4)

After j = 4, the program will come to the first line and will take the value for i = 1 and will repeat the “for j” loop for j=0,1,2,3,4.

So, after the second loop, we will get

(1,0)
(1,1)
(1,2)
(1,3)
(1,4)

INPUT:

```
for i in range (3):
    print ("j loop starts for i", i)
    for j in range (3):
        print (i, j)
    print ("j loop ends for i", i)
```

OUTPUT:

```
j loop starts for i 0      j loop starts for i        j loop starts for i 2
0 0                        1 0                        2 0
0 1                        1 1                        2 1
0 2                        1 2                        2 2
j loop ends for i 0        j loop ends for i 1        j loop ends for i 2
```

INPUT

```
for i in range(3):
    for j in range(3):
        for k in range(3):
            print(i, j, k)
```

OUTPUT

```
0 0 0
                0 2 1           1 1 2           2 1 0
0 0 1           0 2 2           1 2 0           2 1 1
0 0 2           1 0 0           1 2 1           2 1 2
0 1 0           1 0 1           1 2 2           2 2 0
0 1 1           1 0 2           2 0 0           2 2 1
0 1 2           1 1 0           2 0 1           2 2 2
0 2 0           1 1 1           2 0 2
```

Ex: 07

Print a table till n X 10 = ; by taking user_input.

INPUT

```
user_input = int(input("Enter a Number: "))
```

```
for i in range(1, 11):
    print (user_input, "X", i, "=", user_input*i)
```

OUTPUT

```
Enter a Number: 2

2 X 1 = 2
2 X 2 = 4
2 X 3 = 6
2 X 4 = 8
2 X 5 = 10
2 X 6 = 12
2 X 7 = 14
2 X 8 = 16
2 X 9 = 18
2 X 10 = 20
```

Ex: 08

```
for i in range (1, 21):
    print ("**************")
    print ("Table of ", i)
    for j in range (1, 11):
        print (j, "X", i, "=", j*i)
    print ("**************")
```

MCQ's

```
sum = 0

for j in range (2, 3):
    for i in range (1, 3):
        sum += (i + j)

what will be sum at end
```

ANS:

Sum = 0 and for j in range (2,3) we have only one j value, 2. And for i in range (1,3) we have will have 2 values for i, (i = 1, 2).

i = 1, 2   j = 2

So, sum += (i + j) = sum + (i + j)

So, sum = 0 + 1 + 2 = 3.

After the first loop, the value of sum is 3.

For the second loop,

sum = 3+2+2 = **7.**

```
Sum = 0

For j in range (2,4):
    for I in range (1,3):
        If I % 2 == 0;
            sum += (I + J)
```

ANS: 9