

Loops (Iteration)

- Repetition of instructions are called loops

Iterate = Value
change

Concept Explanation

Q- Take input 500 times and print sum of all the 500 numbers.

- Can not waste time writing input 500 times, so loop repeats the input action for us.

Types of Loops

- Count-controlled Loop
- Post-conditioned Loop
- Pre-conditioned Loop.

] Used when loop is run based on conditions.

① Count-controlled
Loop

② Post-conditional
Loop

FOR... TO... NEXT

REPEAT ... UNTIL

③ Pre-Conditional Loops

WHILE ... DO ... ENDOF WHILE

CHARACTERISTICS

FOR...TO...NEXT

WHILE...DO...ENDWHILE

REPEAT UNTIL

- | | | |
|---|--|--|
| <ul style="list-style-type: none">• Repetition known• Count- Controlled Loop• No condition• No use of $Count = Count + 1$• Logic: Fixed iteration always execute | <ul style="list-style-type: none">• Repetition unknown• Conditional Loop• Pre - conditional• Use of $Count = Count + 1$• Logic: Loop will run until condition remains True.E.g : Password Approval | <ul style="list-style-type: none">• Repetition unknown• conditional Loop• Post- Conditional• Use of $Count = Count + 1$• logic: Loop will run until condition remains FalseE.g : Password Approval |
|---|--|--|
- Whenever loop breaks, bahir aajaye gi execution
 - Loops have 2 natures :- Type 1 consists of repeating instructions by a fixed number of times

- Type 2 consists of validation. (E.g : Lecture 6 : Password validation) when repetition has unknown number of times of repetition

Syntax

FOR ... TO ... NEXT

```
FOR Count = 1 TO 10  
| <steps>  
ENFOR
```

Initial and starting value of variable

FOR Count = 1 TO 10 Ending Value
| Any variable

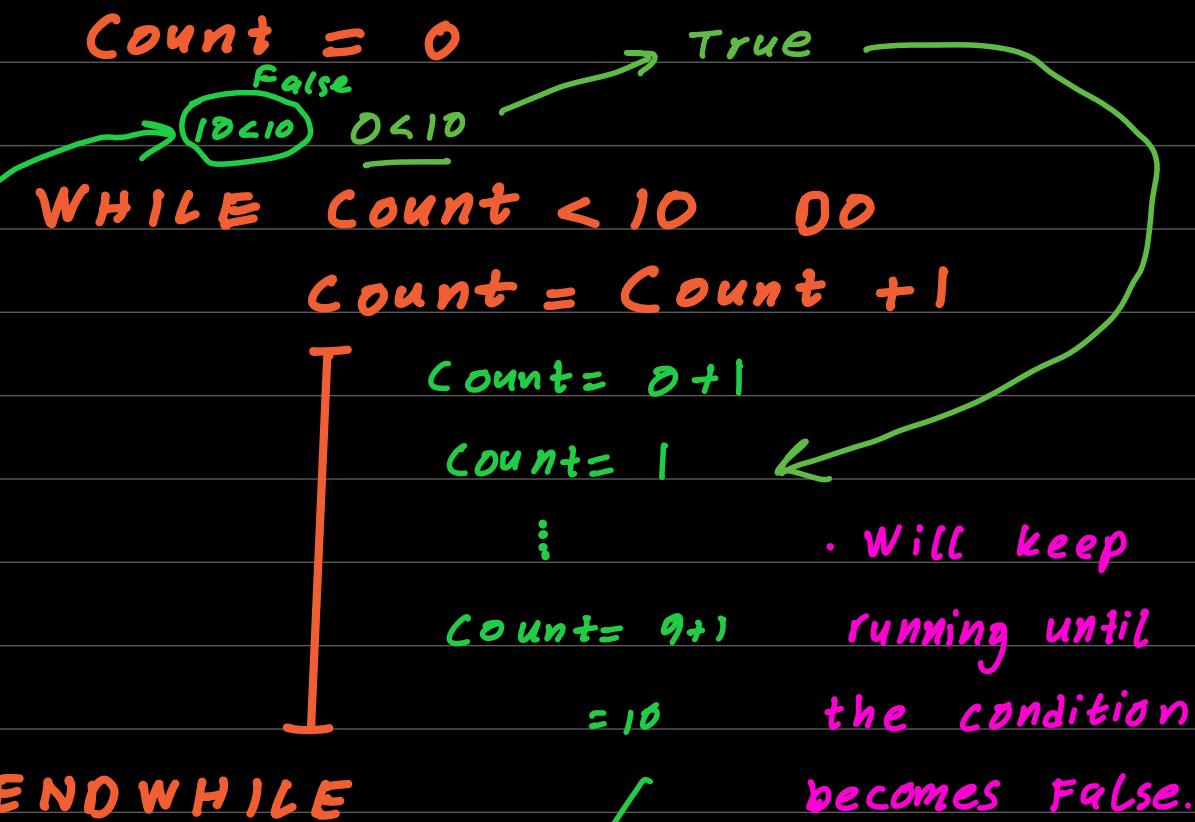
PRINT Count

END FOR

- Variable 'count' increases by one each time the loop repeats the steps

- Loop will run the tenth time, different from python

WHILE ... DO... ENDWHILE



- Used when number of repetitions are unknown and when loop may not be entered at all.

REPEAT...UNTIL

Count = 0

Any Variable

REPEAT

Count = Count + 1
⋮
Count = 10

• Will keep running
until condition becomes

UNTIL Count = 10 True.

10 = 10

Terminate

Used when number of repetitions are unknown and when loop must be executed at least once.

Q- Input 500 names and print in the format Hello 'name' with suitable message.

①

- DECLARE Name: STRING
- DECLARE Count: INTEGER
-

```
FOR Count = 1 TO 500
    PRINT "Enter Your Name"
    INPUT Name
    PRINT "Hello" & ' ' & Name
ENDFOR
```

Q- Which loop should be used and why?

· Count-Controlled Loop should be used b/c number of repetitions are known

· Whenever strings are to be printed together, they have to be concatenated

(· PRINT "Hello", Name) ^X Wrong, in case of string and character

(2)

· DECLARE Count : INTEGER

· DECLARE Name : STRING

·

Count = 1

WHILE Count ≤ 500 DO

Count = Count + 1

PRINT "Enter Your Name : "

INPUT Name

PRINT "Hello" & ' ' & Name

END WHILE

③

DECLARE Count: INTEGER

DECLARE Name : STRING

Count = 1

REPEAT

Count = Count + 1

PRINT "Enter Your Name : "

INPUT Name

PRINT "Hello" & ' ' & Name

UNTIL Count = 500

Programming Techniques

① Sum

Sum = 0

(outside Loop)

Sum = Sum + Number

(inside Loop)

Q- Input 10 numbers and print the sum of those 10 numbers

DECLARE Count, Sum, Number : INTEGER

Sum = 0 → Variable has to be initialised before usage.

FOR Count = 1 TO 10

PRINT "Enter Number: "

INPUT Number

Sum = Sum + Number

END FOR

PRINT Sum → Should be outside loop

Q- Input 465 numbers and print the **sum** of those 465 numbers.

(a) FOR ... TO... NEXT

(b) WHILE ... DO ... ENDWHILE

(c) REPEAT... UNTIL

(a)

DECLARE sum, Number, Count: INTEGER

Sum=0

FOR Count= 1 TO 465

INPUT Number

Sum= Sum + Number

END FOR

PRINT Sum

(b)

DECLARE sum, Number, Count: INTEGER

Count=0

Sum=0

WHILE Count < 465 DO

Count = Count + 1

INPUT Number

Sum = Sum + Number

END WHILE

PRINT Sum

(c)

DECLARE Sum, Number, Count: INTEGER

Count = 0

Sum = 0

REPEAT

Count = Count + 1

INPUT Number

Sum = Sum + Number

UNTIL Count = 465

② Counting

ThingCount = 0

(outside loop)

ThingCount = ThingCount + 1

(inside loop)

- Never use Loop variable for counting numbers

Q- Input 500 numbers and print how many numbers are positive.

Whenever the word "how many" is used, counting technique is to be used.

DECLARE N1, Positive, Count : INTEGER

Positive = 0

FOR count = 1 TO 500

 INPUT N1

 IF N1 > 0

 THEN

 Positive = Positive + 1

 END IF

END FOR

PRINT Positive

Q- Input 499 numbers and print how many numbers are positive, negative , and zero with suitable message (Use repeat until)

DECLARE Positive, Count, Negative , Zero : INTEGER

Positive = 0

Negative = 0

Zero = 0

Count = 0

REPEAT

Count = Count + 1

PRINT "Enter Your Number : "

INPUT Number

CASE OF Number

> 0 : Positive = Positive + 1

< 0 : Negative = Negative + 1

= 0 : Zero = Zero + 1

END CASE

```
UNTIL Count = 499
PRINT "Count of numbers that are positive : ", Positive
PRINT "Count of numbers that are negative : ", Negative
PRINT "Count of numbers that are zero : ", Zero
```

Flag Looping (2nd Nature of Loop)

Q- Ask for input again and again until the correct format of the password is used. (First character must be capital)

• Always use conditional loops.

```
DECLARE Correct : BOOLEAN
```

```
DECLARE Pass : STRING
```

```
DECLARE FirstCharacter : CHAR
```

```
Correct = FALSE
```

```
WHILE Correct = FALSE DO
```

PRINT "Enter your Password : "

INPUT Pass

FirstCharacter = LEFT (Pass, 1)

IF (FirstCharacter ≥ 'A') AND (FirstCharacter ≤ 'Z')

THEN

PRINT "Approve"

Correct = True

ELSE

PRINT "Re-write your password"

ENDIF

ENDWHILE