Loops


```
int n = 0;

while ( n != 0 )
{

    ___
    ___ = process
    ___
}
___
```

```
int n = 0;
do
{

    ___
    ___ = process
    ___
}while( n != 0 );
```

Note: *Ham kabhi bhi kisi ka bhi use kar sakta ha koi compulsion nahi hai ki kiska use karna kai. Waisa dono same hi hai bas minor change kar ka ek dusa form mai convert kar sakta hai.*

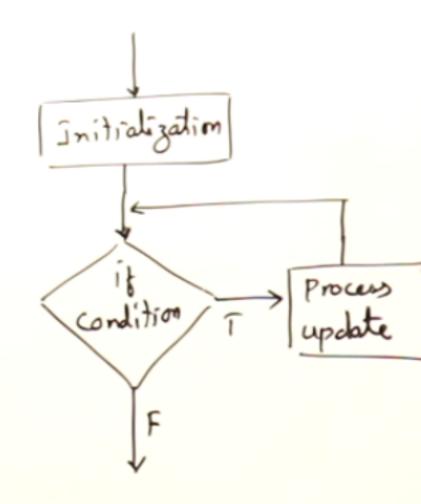**1. First check condition then execute the program.**
2. If condition wrong, then ya loop chlaga hi nahi.

1. First execute program then check the condition.
2. Ek minimum ek baar to chlta hi chlta hai, phir as per condition, kaam karta hai.

Loops

counter
↓

$$\text{for}\,(\underset{①}{\text{initialization}}\,;\, \underset{②⑤⑧}{\text{condition}}\,;\, \underset{④⑦}{\text{updation}})$$
{
$\underset{③⑥\ldots}{\equiv}$
}

2.50

FOR ANY PATTERNS:

※ Outer loop -> no. of rows

※ Inner loop -> no. of column (no. of element in each loop)

1) No. of lines

2) In each line what is happening

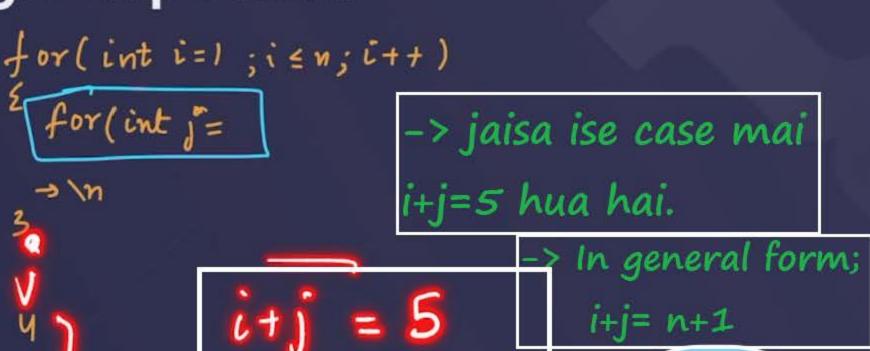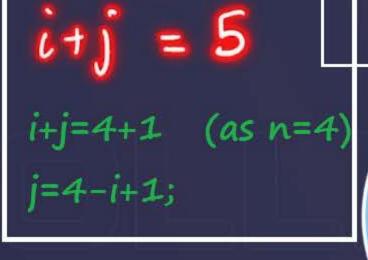hmasa koi relation find karo row and column mai.

2.50

Ques : Print the given pattern

$n = 4$

```
****
***
**
*
```

$n = 4$

```
i
1    *   *   *   k
2    *   *   *
3    *   *
4    *
```

$$for(int\ i=1\ ; i \leq n; i++)$$
$$\{$$
$$for(int\ j=$$
$$\to \backslash n$$

	4
	3
	2
	1

-> jaisa ise case mai $i+j=5$ hua hai.

-> In general form; $i+j = n+1$

$$i + j = 5$$

$i+j=4+1$ (as n=4)

$j=4-i+1;$

Star Triangle Ulta

Decrementing for loop

```
for( i = 7; i !=0 ; i --){
        System.out.println (i);
}
```

This for loop keeps running until i becomes 0.

Quick Quiz : Write a program to print first n
natural numbers in reverse order

break statement
The break statement is used to exit the loop
irrespective of whether the condition is true
or false.
Whenever a "break" is encountered inside the
loop, the control is sent outside the loop.

Continue statement
The continue statement is used to immideately
move to the next iteration of the loop.
The control is taken to the next iteration
thus skipping everything below "Continue" inside
the loop for that iteration.

In a Nut shell...
1. break statement completely exits the loop
2. Continue statement skips the particular iteration
of the loop.