8/09

5) Identity operators :-

is is not

a = 9

b = 9

a is b

o/p :- True

a is not b

o/p :- False

6) Membership operator :-

in not in

I am a member of your family - False in

I am not a member of your family - True not in

you are a member of your family - True in

you are not a member of your family - False not in

Pet = ['dog', 'cat', 'cow', 'rabbit']

'cat' not in pet

o/p :- False

'lion' not in pet

o/p :- True.

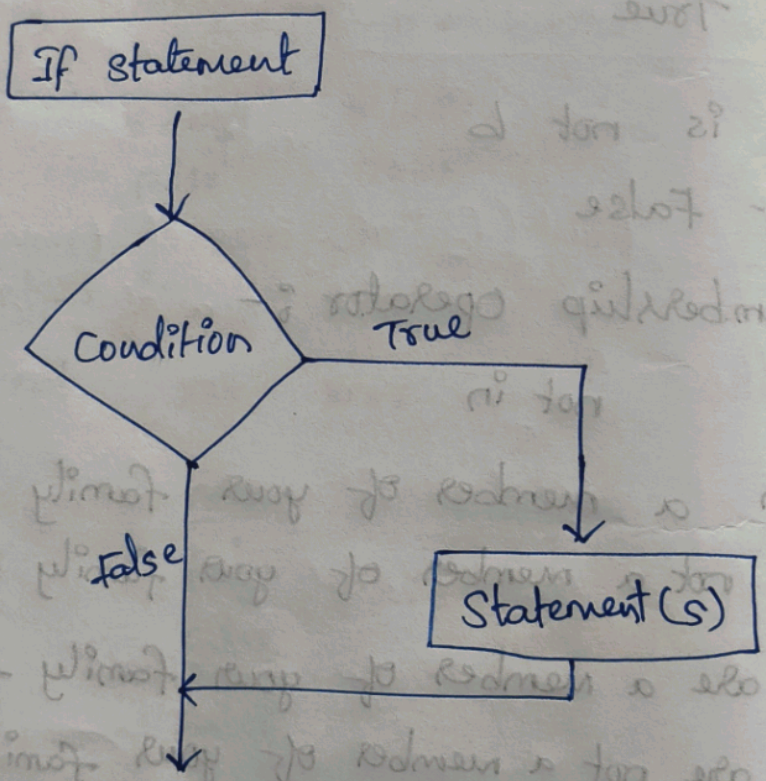
Conditional Statements :-

It is allow us to make decisions in code. they check conditions (expression that result in True or False) and execute different blocks of code accordingly.

Types of conditional statements :-

1) if statement - execute a block of only if the condition is True.
2) if...else statement - provides two paths: one if condit^n is True, another if False.
3) if ... elif ... else ladder - multiple conditions checked one by one.
4) nested if - using one if inside another.

(1)



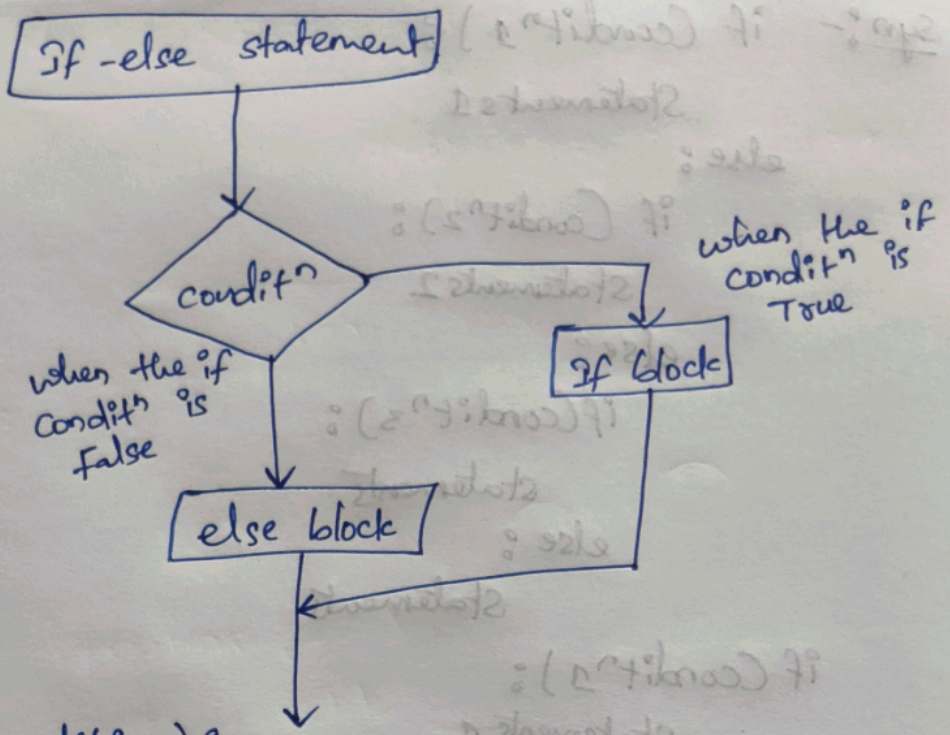
Syntax :-

```
if (condition):
    Statements
```

Prg :-

```
x = 2
if (x>5):
    print (" x is greater than 5")
```

② If-else statement

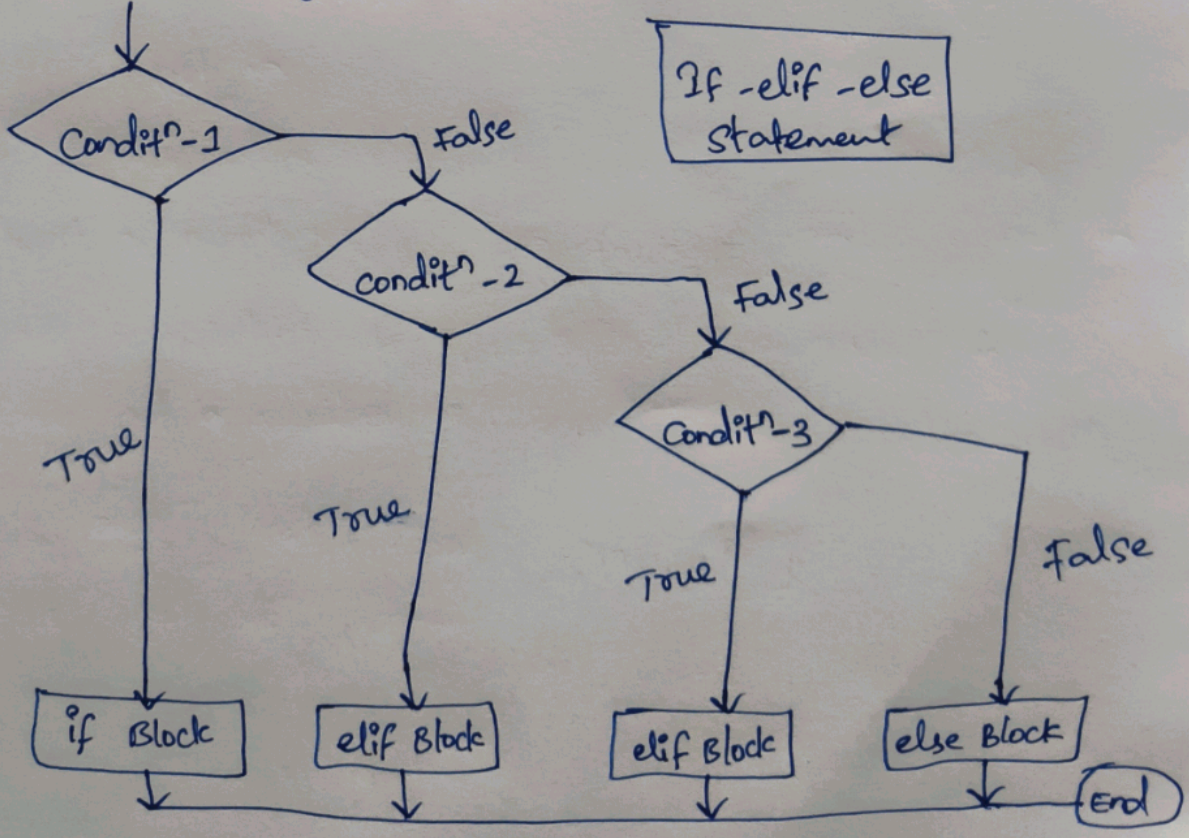


Syn :- if (condition) :
    statements
else :
    statements

Prog :- x = 2
if (x > 5) :
    print ("x is greater than 5")
else :
    print ("x is not greater than 5")

o/p :- x is not greater than 5.

③ If-elif-else statement



Syn:-

```
if (condit^n 1):
    Statements 1
else:
    if (condit^n 2):
        statements 2
    else:
        if(condit^n 3):
            statements
        else:
            statements
```

```
if (condit^n 1):
    statements 1
elif (condit^n 2):
    statements 2
elif (condit^n 3):
    statements 3
else:
    statements
```