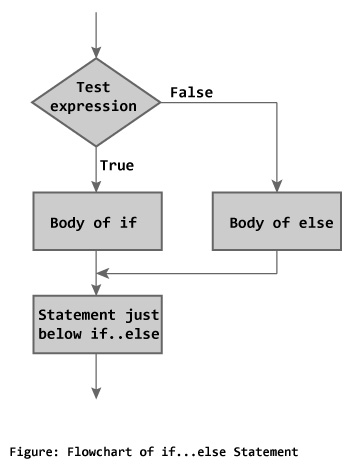
CONDITIONAL STATEMENTS:

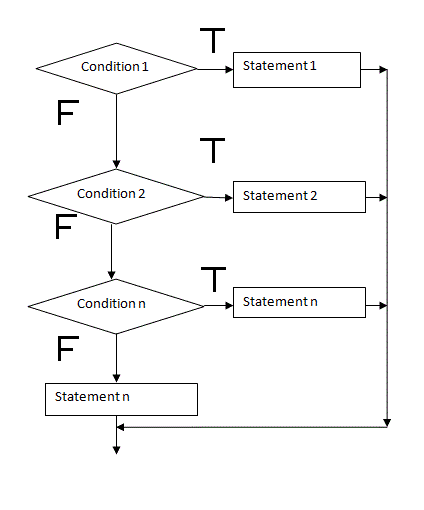
* A conditional statement is a mechanism that allows for conditional execution of instructins based upon the outcome of a conditional statement, which can either be true or false.
* If indentation is wrong it will through an error.
* Indentation should be maintained correctly.
* if else statements
  + syntax
    - if (*condition*):
      * + <*statement-1*>

else:

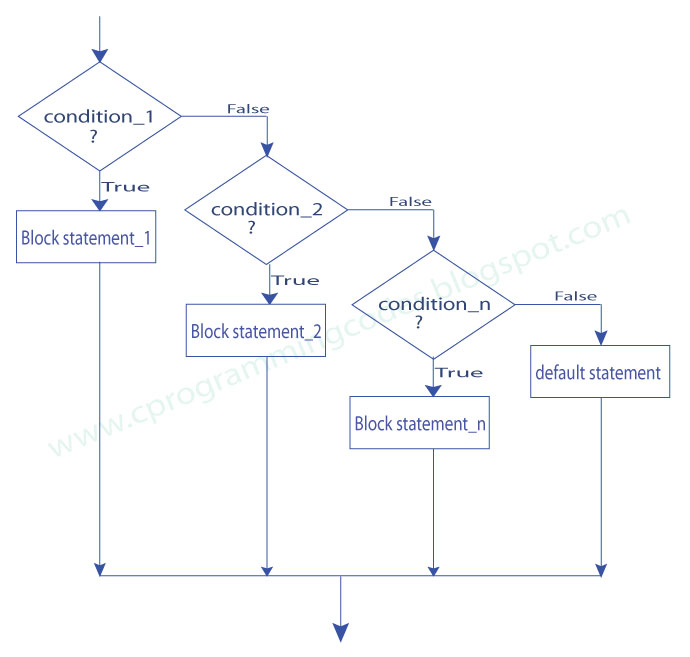
* + - * + <statement-2>
  + flow graph



* if elif else statements
  + syntax
    - if (*condition*):
      * + <*statement-1*>
    - elif(*condition*):
      * + <*statement-2*>
    - else:
      * + <statement-3>
  + flow graph



* Nested if statements
  + syntax
  + if (*condition*):
    - * if (condition):
        + <*statement-1*>
      * else:
        + <*statement-2*>
  + else:
    - * <*statement-3*>
  + flow graph



* Switch is not there in python.

FOR LOOPS , WHILE:

* Loopings are of two types
  + Entry controlled
  + Exit controlled
* Loops are of two types
  + finite loop
  + infinite loop
* In python looping statements can used on collection,strings and numbers.
* Loops are called iterable objects
* for loop
  + syntax:
    - for var in range( starting value , end value , step count )
      * <statements>
    - for var in range <string variable>
      * <statements>
* while loop
  + intialization and increment/decrement is done anywhere in the code.
  + syntax:
    - while (condition):
      * + <*statements*>
* ctrl+c -> is keyword interrupt
* Control statements:
  + pass:The **pass** statement in **Python** is used when a statement is required syntactically but you do not want any command or code to execute. The **pass** statement is a null operation; nothing happens when it executes.
  + Continue:The **continue** statement rejects all the remaining statements in the current iteration of the loop and moves the control back to the top of the loop.
  + Break:the **break** statement provides you with the opportunity to exit out of a loop when an external condition is triggered

CLI ARGUMENTS **:**

* Command Line Arguments are used send the required at the time of execution.
* It considers the given data in string format
* In python we use this by importing sys module
* Syntax:
  + import sys
  + access those by using sys.argv[1........n]
  + But sys.argv[1] is considered only for python file.
  + but from sys.argv[2] to sys.argv[n] we give data values.

STRING LITERALS:

* %d and %s are used in print() to access the data values
  + example:print(“%d x %d=%d”%(a,i,(a\*i)))