Decision Making or Conditional Statements:

Decision Making statements are used to control the flow of execution of a program depending upon condition.

Python language provide the following conditional statements:

if statement

if...else statement

if...elif...else statement

Nested if statement

If statement:

- If statement contains a logical expression or condition, depending on which the decisions are made.
- The expression is checked and if it returns True then statement(s) will be executed, otherwise they are skipped.

If...else statement:

The block of code inside if statement is executed if the expression in if statement returns True, else the block of code inside else is executed.

If...elif...else statement:

- If...elif...else statement is used to test additional conditions apart from the initial test expression.
- It works in the same way as usual to if...else statement.
- It is not necessary that every if statement should have an else block.
- Also called chained conditional.

Nested if statement:

- In a nested if construct, we can write an entire if...else statement within another if...else statement.
- An if...elif...else construct also placed inside an if...elif...else construct.

Looping and Iteration Statements:

A loop statement allows us to execute a statement or group of statements multiple times.

Python language provide the following looping statements:

For loop

While loop

Nested loop

For Loop:

- For loop is a repetitive control structure which is used to execute a set of instructions repeatedly.

- It is also used to iterate over items of any sequence, such as a list or a string.
- In each iteration new value assigned for looping variable.

The range() function returns a sequence of numbers, starting from 0 by default, and increments by 1 (by default), and ends at a specified number

Syntax:

range*(start, stop, step*)

While Loop:

- A while loop statement repeatedly executes a target statement as long as the given condition is true.
- It tests the condition before executing the loop body.

Loop Control Statements:

Loop control statements change execution from its normal sequence.

Python language provide the following loop control statements:

Break statement

Continue statement

Pass statement

The break statement is used to terminate the execution of the loop at a specific point. It brings control out of the loop.

The continue statement causes the loop to skip the remainder of its body and immediately retest its condition prior to reiterating. It returns the control to the beginning

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 do nothing. It is also used in places where the program code can be written in future.

For with else:

The else block just after for/while is executed only when the loop is NOT terminated by a break statement.