PYTHON Control Structures:
In PYTHON Programming Control Structures are classified into:
1. Sequential Control Structures
2. Selection Control Structures
3. Iterative Control Structures

1. Sequential Control Structures:
It get excutes the lines of code in sequential order.

Example:
```
print("First Line")
print("Second Line")
print("Third Line")
```

Example:
```
print("First Line");print("Second Line");print("Third Line")
```

2. Selection Control Structures: (Conditional Control Statements)
It is popularly known as Python Decision Making. Python programming language provides following types of decision making statements.
1 if statement (One-Way Decisions)
2 if .. else statement (Two-Way Decisions)
3 if .. elif .. else statement (Multi-Way Decisions)
4 Nested if .. else (inner Decisions)
5 Negative Conditions (Using Member-Ship Operators)

In Conditions, the following Comparison or Relational Operators Commonly Using:
1 > greater than

2 >= greater than equalto

3 < less than

4 <= less than equalto

5 == equal
6 != not equal

Python 'if' Statement
It executes a set of statements conditionally, based on the value of a logical expression.

Syntax:
```
if expression :
    statement_1
    statement_2
    ....
```

Example:
```
num = 3
if num > 0:
    print(num, "It is a Positive Number.")
    print("This is always printed.")

num = -1
```

```
if num > 0:
    print(num, "It is a Positive Number.")
    print("This is also always printed.")
```

Example:

```
num=input("Enter any Number: ")
if int(num) > 0:
  print(num, "It is a Positive number.")
```

Example:

```
x=int(input("Enter Any Number: "))
if x>0:
    print("Value is +VE");
    print("It is always get Executed");
y=int(input("Enter Any Number: "))
if y<0:
    print("Value is -VE");
    print("It is always get Executed");
```

Example:(Try in 2.x)

```
num=input("Enter Any Data: ")
print("Type of the Data is: ")
print(type(num))
```

2 if .. else Statement

An else statement can be combined with an if statement. An else statement contains the block of code that executes if the conditional expression in the if statement resolves to 0 or a FALSE value.

Syntax

```
if expression:
   Statement(s)_1
   Statement(s)_2
else:
   Statement(s)_3
   Statement(s)_4
```

Example:

```
num=input("Enter any Number: ")
if int(num) > 0:
  print(num, "It is a Positive number.")
else:
  print(num, "It is a Negative number.")
```

Example:

```
num=input("Enter any Number: ")
print(type(num))
if int(num) > 0:
  print(num, "It is a Positive number.")
else:
  print(num, "It is a Negative number.")
```

Example:

```
num = 3
if num > 0:
    print(num, "It is a Positive Number.")
    print("This is always Printed.")
```

```
else:
    print(num, "It is a Negative Number.")
    rint("This is also Printed.")

Example:
num = int(input("Enter a number: "))
if (num % 2) == 0:
   print("is Even")
else:
   print("is Odd")
```

Python if...elif...else
The elif statement allows you to check multiple expressions for TRUE and execute a block of code as soon as one of the conditions evaluates to TRUE.

Syntax

```
if Expression1:
   statement(s)_1
   statement(s)_2
elif Expression2:
   statement(s)_3
   statement(s)_4
elif Expression3:
   statement(s)_5
   statement(s)_6
else:
   statement(s)_7
   statement(s)_8
```

NOTE:
Core Python does not supports switch or case statements as in other languages.

Example:

```
num=input("Enter any Number: ")
print(type(num))
if int(num) > 0:
  print(num, "It is a Positive number.")
elif int(num)<0:
  print(num, "It is a Negative number.")
else:
  print(num, "It is a ZERO.")
```

Example:

```
a=200;b=2
if b>a:
    print("b is bigger")
elif a==b:
    print("Both are equal")
else:
    print("a is Bigger")
```

Example:

```
x=int(input("Enter Any Number: "))
y=int(input("Enter Any Number: "))
if x>0 and y>0:
```

```
    print("X and Y are Positives: ")
elif x==0 and y==0:
    print("Values are ZEROS")
else:
    print("Values are Negatives")
    print("Thank U")
```

Example:

```
grade=int(input("Enter Your Marks: "))
if grade >= 90:
    print("Your Grade is A+")
elif grade >=80:
    print("Your Grade is A")
elif grade >=70:
    print("Your Grade is B+")
elif grade >=60:
    print("Your Grade is B")
elif grade >=50:
    print("You are Pass")
else:
    print("Sorry You Failed")
```

Example:

```
Marks=int(input("Enter Valid Marks: "))
if Marks>=90 and Marks<=100:
    print("Grade is A+")
elif Marks>=80 and Marks<=89:
    print("Grade is A")
elif Marks>=70 and Marks<=79:
    print("Grade is B+")
elif Marks>=60 and Marks<=69:
    print("Grade is B")
elif Marks>=50 and Marks<=59:
    print("Grade is C")
elif Marks>=40 and Marks<=49:
    print("JUST PASS")
else:
    print("Invalid Marks/Failed")
```

EXAMPLE:

```
First=int(input("Enter Any Number:"))
Second=int(input("Enter Any Number:"))
Third=int(input("Enter Any Number:"))
if First>Second and First>Third:
    print("Biggest Number is:",First)
elif Second>Third:
    print("Biggest Number is:",Second)
else:
    print("Biggest Number is:",Third)
```

Nested if .. else statement

In general nested if-else statement is used when we want to check more than one conditions. Conditions are executed from top to bottom and check each condition whether it evaluates to true or not.

Syntax:

```
     if expression1 :
```

```
          if expression2 :
           Statement_3
           Statement_4
         ....
       else :
          Statement_5
          Statement_6
         ....
      else :
        Statement_7
        Statement_8
```

Example:

```
num = int(input("Enter a number: "))
if num >= 0:
    if (num == 0):
        print("ZERO")
    else:
        print("Positive number")
else:
    print("Negative number")
```

Example:

```
x=int(input("Enter Any Number: "))
if x!=0:
    if x<0:
        print("-VE")
    else:
        print("+Ve")
else:
    print("ZERO")
```

Example:

```
x=int(input("Enter Any Number: "))
if x==0:
    print("ZERO")
else:
    if x>0:
        print("+VE")
    else:
        print("-VE")
```

Example:

```
x=int(input("Enter Any Number: "))
if x>=0:
    if not x==0:
        print("+VE")
    else:
        print("ZERO")
else:
    print("-VE")
```

Example:

```
grade=int(input("Enter Your Marks: "))
if grade >= 90:
    print("Your Grade is A+")
else:
```

```
    if grade >=80:
        print("Your Grade is A")
    else:
            if grade >=70:
                print("Your Grade is B+")
            else:
                    if grade >=60:
                        print("Your Grade is B")
                    else:
                            if grade >=50:
                                print("You are Pass")
                            else:
                                    print("Sorry You Failed")
```

Example:

```
year =int(input("Enter Any Year: "))
if (year % 4) == 0:
   if (year % 100) == 0:
       if (year % 400) == 0:
           print("is a leap year")
       else:
           print("is not a leap year")
   else:
       print("is a leap year")
else:
   print("is not a leap year")
```

Example: Use the and operator in an if statement

```
x = False
y = True
if x and y:
    print('Both x and y are True')
else:
    print('x is False or y is False or both x and y are False')
```

Negative Conditions in PYTHON:

If a condition is true the not operator is used to reverse the logical state, then logical not operator will make it false.

Example:

```
x = int(input("Enter Any Number: "))
print(x)
if not x == 50:
    print('the value of x different from 50')
else:
    print('the value of x is equal to 50')
```

Example:

```
x=int(input("Enter Any Number: "))
if not x>0:
    print("Value is -VE")
else:
    print("Value is +VE")
```

RealTime Case Studies:

Example:

```
# Python example of "in" and "not in" Operators
PyStr = "Hello world"
PyList = [10, 20, 30, 40, 50]
# Check 'w' (capital exists in the str1 or not
if 'w' in PyStr:
        print("Yes! w found in ", PyStr)
else:
        print("No! w does not  found in " , PyStr)
# check 'X' (capital) exists in the str1 or not
if 'X' not in PyStr:
        print("yes! X does not exist in ", PyStr)
else:
        print("No!  X exists in ", PyStr)
# check 30 exists in the list1 or not
if 30 in PyList:
        print("Yes! 30 found in ", PyList)
else:
        print("No! 30 does not found in ", PyList)
# check 90 exists in the list1 or not
if 90 not in PyList:
        print("Yes! 90 does not exist in ", PyList)
else:
        print("No! 90 exists in ", PyList)
```

Example2:

```
x=1
print(--x)#1
print(++x)#1
print(x--) #SyntaxError: invalid syntax
```

NOTE:
The reason for this is, in python integers are immutable and hence cannot be changed. So, we will have to do the following for incrementing.

```
x = 1; x=x+1
print(x)

x=x-1
print(x)

x+=2
print(x)

x-= 1
print(x)
```