Wanna take DECISIONS in Python?

Four ways:

|  |  |
| --- | --- |
| 1. if condition:   St1  St2 | 1. if condition:   St1  St2  else:  St3  St4 |
| 1. if condition1:   St1  St2  elif conditon2:  St3  St4  elif conditon3:  St5  else:  St3 | 1. if condition1:   St1 #optional  if condition2:  St2  St3  else:  St4  St5  St6 #Optional  else:  St7 #optional  if condition3:  St8  St9  else:  St10  St11  St12 #Optional |

Relational Operators:

< > <= >= == !=

if a < b < c: # b falls between a and c.

if a == b == c: # all three are equal.

if 10 != 20 != 10 # Answer is True.

Logical Operators:

**and**, **or**, **not**

Every time we may not get True/False

1. **and** evaluates all expressions. It returns last expression if all expressions evaluate to True, otherwise, first value that evaluates to False.

`a = 40

b = 30

x = **(75 and a >= 20 and b< 60 and 35)** # assigns 35 to x

y = -30 and a >= 20 and b < 15 and 35 # assigns False to y

z = -30 and a >= 20 and 0 and 35 # assigns 0 to z

1. **or** evaluates all expressions. It returns first expression that evaluates to True, otherwise, last value that evaluates to False.

a = 40

b = 30

x = 75 and a >= 20 and b< 60 and 35: # assigns 35 to x

x = 75 or a >= 20 or 60 # assigns 75 to x

y = a >= 20 or 75 or 60 # y = True

z = a < 20 or 0 or 35 # z = 35

1. **not** is used to negate the result of the condition.

a = 10

b = 20

not ( a <= b) # False , Same as a > b

Conditional Expressions(Short-hand if-else):

**<expr1> if < conditional expression> else < expr 2>**

<conditional expression> is evaluated first. If it is True, the expression evaluates to <expr1>, otherwise <expr2>

Age = 18

status = ‘minor’ if age < 18 else ‘major’

all() and any()

**and** and **or** can be replaced with built-in functions all() and any().

a, b, c = 10, 20 , 30

res = all (a>5, b > 20,c > 15)

print (res) # prints false

res = any (a>5, b > 20, c > 15)

print (res) # prints True