Table of Contents

Chapter 8

Conditional Expression

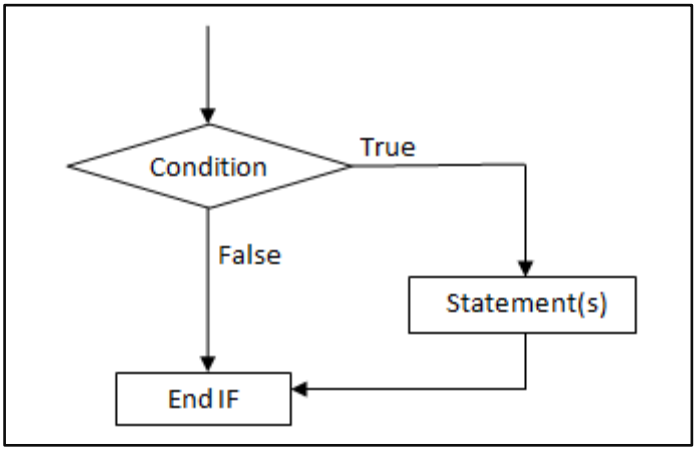
A conditional expression in Python is an expression (in other words, a piece of code that evaluates to a result) whose value depends on a condition.
Python's conditional statements carry out various calculations or operations according to whether a particular Boolean condition is evaluated as true or false. In Python, IF statements deal with conditional statements.

Equals: a == b # Relational Operator
Not Equals: a != b # Relational Operator
Less than: a < b # Comparison Operator
Less than or equal to: a <= b # Relational Operator
Greater than: a > b # Comparison Operator
Greater than or equal to: a >= b # Relational Operator
These conditions can be used in several ways, most commonly in "if statements" and loops.

The if statement:

To make decisions, utilize the if statement in Python. It has a body of instructions that only executes whenever the if statement's condition is met. The additional else statement, which includes some instructions for the else statement, runs if the if condition is false.
Ex.

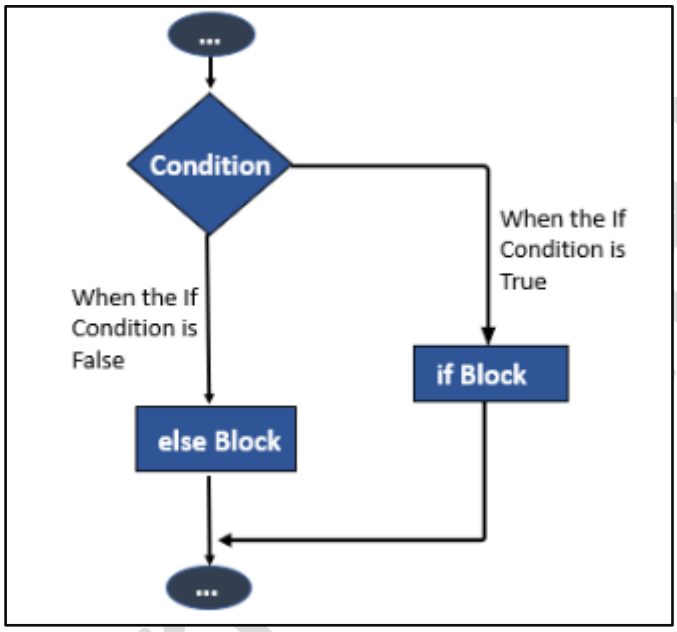
```
a, b = 6, 5
if a > b:
   code = "a is greater than b"
   print(code)
# a is greater than b
```



The Else Statement

Ex.

```
if a < b:
   code = "a is less than b"
   print(code)
else:
   print("a is greater than b")
# a is greater than b
```



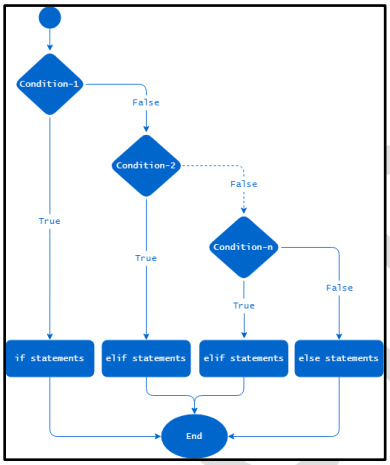
The indentation

```
# Indentation
# Python relies on indentation (whitespace at the beginning of a line) to define scope in the code. Other programming languages often use curly-brackets for this purpose.
# 4spaces or tab
# a = 33
# b = 200
# if b > a:
# print("b is greater than a") # you will get an error
```

The use of elif statement

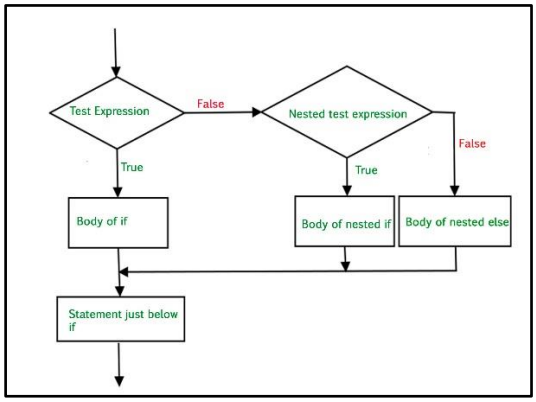
```
a, b = 9, 9
if a < b:
   code = "a is less than b"
elif a == b:
   code = "a is equal to b"
else:
   code = "a is greater than b"

print(code)  # a is equal to b
```



The Nested if-else statement

```
x = 41
if x > 10:
  print("Above ten,")
  if x > 20:
    print("and also above 20!")
  else:
    print("but not above 20.")
# Above ten,
# and also above 20!
```



The Short hand if

```
# If you have only one statement to execute, you can put it on the same line as the if statement.
print("Short Hand If")
a = 7
if a<18: print("Number is smaller than 18")
```

The Short hand if else

```
# if you have only one statement to execute, one for if, and one for else, you can put it all on the same line:
print("Short Hand If ... Else")
print("You are not legally official") if age<18 else print("You are legally official")
```

The use of And

```
# The and keyword is a logical operator, and is used to combine conditional statements:
print("Use of And")
a = 200
b = 33
c = 500
if a > b and c > a:
  print("Both conditions are True")
```

The use of Or

```
# The use of Or
# The or keyword is a logical operator, and is used to combine conditional statements:
print("Use of Or")
if a > b or a > c:
  print("At least one of the conditions is True")
```

The use of Not

```
# The not keyword is a logical operator, and is used to reverse the result of the conditional statement:
print("Use of not")
if not a > b:
  print("a is NOT greater than b")
```

The use of pass statement

```
# The pass Statement
# if statements cannot be empty, but if you for some reason have an if statement with no content, put in the pass statement to avoid getting an error.
print("Use of pass")
a = 33
b = 200
if b > a:
  pass
```

The use of in and is

```
# in and is
print("use of in and is")
a=7
b=5
if a is 5:
   print("a is 5")
elif a is 7:
   print("a is 7")
else:
   print("a is not 5 or 7")

a = [1,2,3]
if 2 in a:
   print("2 is present in a")
else:
   print("2 is not present in a")
```