

Chapter 6 - Conditional Expressions

Sometimes we want to play PUBG on our phone if the day is Sunday.

Sometimes we order Icecream online if the day is Sunny.

Sometimes we go hiking if our parents allow.

All these are decisions which depends on a condition being met.

In Python programming too, we must be able to execute instructions on a condition(s) being met. This is what conditionals are for!

If else and elif in Python

If else and elif statements are a multiway decision taken by our program due to certain conditions in our code.

Syntax :

<pre>if (condition1): print ("yes") elif (condition2): print ("No") else: print ("Maybe")</pre>	<p>Indentation \leftarrow</p> <p>\Rightarrow if condition 1 is true</p> <p>\Rightarrow if condition 2 is true</p> <p>\Rightarrow otherwise</p>
---	--

Code example :

```
a = 22  
if (a > 9):  
    print ("Greater")  
else:  
    print ("Lesser")
```

Quick Quiz: Write a program to print yes when the age entered by the user is greater than or equal to 18.

Relational Operators

Relational operators are used to evaluate conditions inside the if statements. Some examples of relational operators are:

$=$ $=$ \rightarrow equals

$>=$ \rightarrow greater than/equal to

$<=$, etc.

Logical operators

In python logical operators operate on conditional statements. Example:

and \rightarrow true if both operands are true else false

or \rightarrow true if at least one operand is true else false

not \rightarrow inverts true to false & false to true

elif clause

elif in python means [else if]. An if statement can be chained together with a lot of these elif statements followed by an else statement

```
if (Condition 1):
```

```
    # code
```

```
elif (Condition 2):
```

```
    # code
```

```
elif (Condition 3):
```

```
    # code
```

```
...
```

```
else:
```

```
    # code
```

\Rightarrow This ladder will stop once a condition in an if or elif is met.



Important notes:

1. There can be any number of elif statements.
2. Last else is executed only if all the conditions inside elif fail.

Chapter 6 - Practice Set

1. Write a program to find greatest of four numbers entered by the user.
2. Write a program to find out whether a student is pass or fail, if it requires total 40% and at least 33% in each subject to pass. Assume 3 subjects and take marks as an input from the user.
3. A spam comment is defined as a text containing following keywords:
"make a lot of money", "buy now", "subscribe this", "click this". Write a program to detect these spams.
4. Write a program to find whether a given username contains less than 10 characters or not.
5. Write a program which finds out whether a given name is present in a list or not.
6. Write a program to calculate the grade of a student from his marks from the following scheme:

90 - 100 → Ex

80 - 90 → A

70 - 80 → B

60 - 70 → C

50 - 60 → D

< 50 → F

7 Write a program to find out whether a given post is talking about "Harry" or not.

Chapter 7 - Loops in Python

Sometimes we want to repeat a set of statements in our program. For instance: Print 1 to 1000

Loops make it easy for a programmer to tell the computer, which set of instructions to repeat and how!

Types of loops in Python

Primarily there are two types of loops in Python

- 1> While loop
- 2> For loop

We will look into these one by one!

While loop

The syntax of a while loop looks like this:

While Condition:

Body of the loop

=> The block keeps executing until the condition is true

In while loops, the condition is checked first. If it evaluates to true, the body of the loop is executed, otherwise not!

If the loop is entered, the process of [Condition check & execution] is continued until the condition becomes false.

Quick Quiz: Write a program to print 1 to 50 using a while loop.

An Example

```
i = 0
while i < 5:
    print("Harry")
    i = i + 1
```

⇒ Prints "Harry" - 5 times!

Note: If the condition never becomes False, the loop keeps getting executed.

Quick Quiz: Write a program to print the content of a list using while loops.

For loop

A for loop is used to iterate through a sequence like list, tuple or string [iterables]

The syntax of a for loop looks like this:

```
l = [1, 7, 8]
for item in l:
    print(item)
```

→ print 1, 7 and 8

Range function in Python

The range function in python is used to generate a sequence of numbers.

We can also specify the start, stop and step-size as follows:

```
range(start, stop, step-size)
```

↳ Step size is usually not used with range()

An Example demonstrating range() function

```
for i in range(0, 7):  
    print(i)
```

→ range(7) can also be used
→ prints 0 to 6

For loop with else

An optional else can be used with a for loop if the code is to be executed when the loop exhausts

Example :

```
l = [1, 7, 8]  
for item in l:  
    print(item)
```

else:

```
    print("Done")
```

→ This is printed when the loop exhausts!

Output :

1

7

8

Done

The break statement

'break' is used to come out of the loop when encountered
It instructs the program to - Exit the loop now

Example :

```
for i in range(0, 80):  
    print(i)  
    if i == 3:  
        break
```

→ This will print 0, 1, 2 and 3

The continue statement

'Continue' is used to stop the current iteration of the loop and continue with the next one. It instructs the program to "skip this iteration".

Example:

```
for i in range(4):  
    print("printing")  
    if i == 2:  
        continue  
    print(i)
```

⇒ if i is 2, the iteration is skipped

pass statement

pass is a null statement in python. It instructs to "do nothing".

Example:

```
l = [1, 7, 8]  
for item in l:  
    pass
```

→ Without pass, the program will throw an error.