

For in Python loop

In this article, we will learn more about for loop in Python as well as its variations, how to use for to repeat a string of elements in Python such as list, string or other iterative objects.

In the previous section we learned quite a bit about the if statement in Python . Read this article to make sure you don't miss out on useful information about it.

In this article, we will learn more about the for loop in the Python programming language as well as its variants, how to use for to repeat a string of elements in Python such as list, string or other iterative objects. .

Syntax for for in Python

```
for bien_lap in chuoi_lap:  
    Kh?i l?nh c?a for
```

In the above syntax, chuoi_lap is the string to repeat, bien_lap is the variable that receives the value of each item inside chuoi_lap on each iteration. The loop will continue until it iterates to the last item in the string.

The for block is indented to distinguish the rest of the code.

Example 1:

```
#L?p ch? cái trong quantrimang  
for chu in 'quantrimang':  
    print('Ch? cái hi?n t?i:', chu)  
#L?p t? trong chu?i  
chuoi = ['b?', 'm?', 'em']  
for tu in chuoi:  
    print('Anh yêu', tu)
```

We have the following output:

```
Ch? cái hi?n t?i: q  
Ch? cái hi?n t?i: u  
Ch? cái hi?n t?i: a  
Ch? cái hi?n t?i: n  
Ch? cái hi?n t?i: t  
Ch? cái hi?n t?i: r  
Ch? cái hi?n t?i: i  
Ch? cái hi?n t?i: m  
Ch? cái hi?n t?i: a  
Ch? cái hi?n t?i: n  
Ch? cái hi?n t?i: g
```

Anh yêu b?
Anh yêu m?
Anh yêu em

Example 2:

```
# Tính tổng tất cả các số trong danh sách A
# Danh sách A
A = [1, 3, 5, 9, 11, 2, 6, 8, 10]
# Biến để lưu trữ tổng các số là tong, gán giá trị ban đầu bằng 0
tong = 0
# Vòng lặp for, a là biến lặp
for a in A:
    tong = tong+a
# In ra: Tổng các số là 55
print("Tổng các số là", tong)
```

When running the above code, you will get the result:

```
The total number is 55
```

Function range ()

You can use the range () function to create a sequence of numbers. For example, range (100) will create a number range from 0 to 99 (100 numbers).

The range function (start number, end number, distance between two numbers) is used to create a custom number sequence. If you do not set a distance between two numbers, Python will understand that it is equal to 1 by default.

The range () function does not store all the values in memory but stores the start value, the end value and the distance between the two numbers resulting in the next number in the sequence.

In order for range () to output all values, you need to use the list () function like the example below:

```
#Lệnh 1
print(range(9))
#Lệnh 2
print(list(range(9)))
#Lệnh 3
print(list(range(2, 5)))
#Lệnh 4
print(list(range(0, 15, 5)))
```

We will have the following output:

```
range(0, 9)
[0, 1, 2, 3, 4, 5, 6, 7, 8]
[2, 3, 4]
[0, 5, 10]
```

Each line corresponds to Orders 1, 2, 3, 4 above.

The range () function can be used in conjunction with len () to loop through an array using index, as in the example below:

Example 3:

```
chuo_i = ['b?', 'm?', 'em']  
for tu in range(len(chuo_i)):  
    print("Anh yêu", chuo_i[tu])
```

We have the same output as example 1 above:

```
Anh yêu b?  
Anh yêu m?  
Anh yêu em
```

Combine for with else

In the previous lesson, you saw if . else and if . elif . else. Else can not only combine with if, but in for loop can also be used.

In for, the else block of statements will be executed when the items in the string have been repeated.

Example 4:

```
B = [0, 2, 4, 5]  
for b in B:  
    print(b)  
else:  
    print("?ã h?t s?.")
```

Here, the for loop will print list B until the entries are gone. When the loop ends it executes else's block and prints. We have the result after running the code as follows:

```
0  
2  
4  
5  
?ã h?t s?.
```

The break command can be used to stop the for loop, and the else part will be ignored. In other words, the else part for for will run when no break is executed.

Example 5:

```
#L?p dăy t? 0 ??n 10  
for num in range(0,10):  
    # Repeat on factors of a number in the sequence
```

```

for i in range(2,num):
#Xác định xem số này có phải là số nguyên tố hay không (phép chia có số dư bằng 0)
if num%i == 0:
j=num/i #Số chia nhỏ nhất của số này
print ('Số %d bằng %d * %d' % (num,i,j))
break #Kết thúc vòng for hiện tại, chuyển đến số tiếp theo trong vòng for tiếp
else: # Phần else trong vòng lặp
print (num, 'là số nguyên tố')

```

The above code repeats the numbers in the range from 0 to 10, with each number running the loop to check if it is a prime number, if it is, print the message and stop the test loop, switch to the next number within First iteration, if not a prime number, then execute another else block. Running the above code we have the following result:

```

0 là số nguyên tố
1 là số nguyên tố
2 là số nguyên tố
3 là số nguyên tố
4 bằng 2 * 2
5 là số nguyên tố
6 bằng 2 * 3
7 là số nguyên tố
8 bằng 2 * 4
9 bằng 3 * 3

```

Above is the most basic knowledge about for loop, in the next article, you will learn about while loop, do you still remember where it was in the introduction series of Python TipsMake.com?

Next article: While loop in Python

Previous lesson: if, if . else, if . elif . else in Python

You finished reading the article "**For in Python loop**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.