

The next () function in Python

The next () function in Python returns the next element in the iterator. You can add a default value to return if iterable is already the last element.

Continuing with the topic of built-in functions in Python, the article will introduce you to the next () function with syntax, usage as well as specific examples. Invites you to read the track.

The next () function in Python returns the next element in the iterator. You can add a default value to return if iterable is already the last element.

The syntax of the next () function in Python:

```
next(iterator, default)
```

Parameters of the next () function

The next () function has 2 parameters:

1. `iterator`: iterable object.
2. `default`: the default value will be returned if iterable is the last element.

Return value from next ()

1. The next () function returns the next element in the iterator
2. If the iterator runs out, the function returns the default value
3. If the iterator expires without a default value, the `StopIteration` exception will be raised.

Example 1: how does next () work?

```
random = [5, 9, 'cat'] # chuy?  
n list thành iterator randomIterator = iter(random) print(randomIterator) # Output  
?y ra l?i # iterator h?t print(next(randomIterator))
```

When you run the program, the output will be:

```
5 9 cat Traceback (most recent call last): File "python", line 18, in StopIterat
```

In the above code, the list is an iterable object, and you can get the iterator from there using the built-in function `iter ()` in Python.

Read more: [Iterator object in Python](#)

The above example encountered the `StopIteration` exception in the last command because the `next ()` function returned the last value above, no more elements. In this case, you should include a default value of the second parameter so that no error occurs.

Example 2: Iterator has the default value in `next ()`

```
random = [5, 9] # chuy?
n list thành iterator randomIterator = iter(random) # Output: 5 print(next(random
?ã h?t ph?n t?
# Output: '-1' print(next(randomIterator, '-1')) print(next(randomIterator, '-1'))
```

Return value:

```
5 9 -1 -1 -1
```

See also: [Python built-in functions](#)

You finished reading the article "[The next \(\) function in Python](#)" 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.