



[Hindi] Advanced List Slicing In Python Explained | Advanced python tutorials in Hindi



[Overview](#) [Q&A](#) [Downloads](#) [Announcements](#)

Advanced List Slicing In Python Explained | Advanced python tutorials

Introduction:

In this blog we will learn list slicing, reversing a list with one line of code, double colon list slicing, etc.

Basic List Slicing:

If you have ever done list slicing then you must have done it in this way:

```
list1 = [11,24,32,45,75,68,38,87]
print(list1[2:7])
```

Tip: Whatever is in those square brackets, subtract it and you will get the number of items in the output list. For example, in the code mentioned above, square brackets have 2:7, 7-2=5 so number of items in output would be 5.

Output:

```
[32, 45, 75, 68, 38]
```

As you can see there are 5 list items in output.

Negative list slicing:

This type of slicing helps a lot in big lists. When you are not certain about the list's length. You can directly access list from the end. Here is an example:

```
list1 = [11,24,32,45,75,68,38,87]
print(list1[-3:7])
```

If we don't write the ending number then it will print the rest of the list. Eg:-

```
list1 = [11,24,32,45,75,68,38,87]
print(list1[-3:])
```

If you still couldn't understand it then look at this example:

```
list1 = [11,24,32,45,75,68,38,87]
print(list1[:-2])
print(list1[:6])
```

Output:

```
[11, 24, 32, 45, 75, 68]
[11, 24, 32, 45, 75, 68]
```

Both print statements printed the same thing. With negative we just had to count some numbers from behind but in the second print statement we had to count from the start.

Double Colon List Slicing:

In this we write like this `::2`. In this the last number is how many list items(list items-1, if there is `::2` then it will skip 1 item) we want to skip. Like:

```
list1 = [1,2,3,4,5,61,71,8,9,90]
print(list1[::2])
```

Output:

```
[1, 3, 5, 71, 9]
```

```
list2 = [1,2,3,4,5,61,71,8,9,90]
```

```
print(list2[::3])
```

Output:

```
[1, 4, 71, 90]
```

In [::2] it is skipping one number and giving output, in [::3] it is skipping 2 numbers.

Like this we can also target it for a specific part of the list. Example is as follows:

```
list1 = [1,2,3,4,5,61,6,7,0,9,8,90,81]
```

```
print(list1[1:10:3])
```

Output:

```
[2, 5, 7]
```

```
list2 = [1,2,3,4,5,61,6,7,0,9,8,90,81]
```

```
print(list2[5:12:2])
```

Output:

```
[61, 7, 9, 90]
```

In list1 1:10 is the part of the list from which it is skipping 2 numbers and printing. In list2 5:12 is the part from which it is skipping 1 number and printing.

Reversing a list:

```
list1 = [1,2,3,4,5,61,6,7,0,9,8,90,81]
```

```
print(list1[::-1])
```

Output:

```
[81, 90, 8, 9, 0, 7, 6, 61, 5, 4, 3, 2, 1]
```

See how easy it is to reverse a list with the help of double colon list slicing.

```
list2 = [1,2,3,4,5,61,6,7,0,9,8,90,81]  
print(list2[::-2])
```

[Copy](#)

Output:

```
[81, 8, 0, 6, 5, 3, 1]
```

You can also reverse the list and simultaneously skip the numbers as shown above.

[Previous](#)[Next](#)

CodeWithHarry

Copyright © 2022 CodeWithHarry.com

