# Slicing

**`Slicing` is the extraction of a part of a string, list, or tuple. It enables users to access the specific range of elements by mentioning their indices.**

**Syntax:**

1. **Object [start : stop : step/direction]**
2. **Object [start : stop]**

`start:` The start parameter in the slice function is used to set the starting position or index of the slicing. The default value of the start is 0.
`stop:` The stop parameter in the slice function is used to set the end position or index of the slicing[(n-1) for positive value and (n+1) for negative value.
`step:` The step parameter in the slice function is used to set the number of steps to jump. The default value of the step is 1.

## Rules for working

`Step1:`-Need to check step direction by default it's goes to positive direction.
`Setp2:` Need to check start-point and end-point direction.
`Step3:` If both directions are matched, then working fine.
`Step4:` Otherwise it gives empty subsequence.

Example:

```
var = "I love python"
print(var[::])

# Output
I love python
```

Example:

```
var = "I love python"
print(var[::-1])

# Output
nohtyp evol I
```

Example:

```
var = "I love python"
print(var[-2:-5:])

# Output
empty string/blank output
```

Example:

```
var = "I love python"
print(var[2:5:-1])

# Output
empty string/blank output
```

Example:

```
var = "I love python"
print(var[::2])

# Output
Ilv yhn
```

Example:

```
var = "I love python"
print(var[::-2])

# Output
nhy vlI
```

Example:

```
var = "WELCOME TO MY BLOG"
print(var[3:18])
print(var[2:14:2])
print(var[:7])
print(var[8:-1:1])
print(var[-6:-9:-3])
print(var[-9:-9:-1])
# Output
COME TO MY BLOG
LOET Y
WELCOME
TO MY BLO
Y
```