**String slicing:-**

* String slicing technique is process of accessing subpart of string.
* You can return a range of characters by using the slice syntax.
* This concept is widely used in many programming languages like Python, JavaScript, and others.

**Syntax**:-string[Start:End:Step]

**Start:**-The starting index of the slicing

**End:**The ending index of the slice

**Step:-**The step size or interval between indices.

**Default values:-**

1. Start:-0
2. End:-length of string
3. Step value:-1

**Example:-**

S=’hello’ worl;

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ‘h’ | ‘e’ | ‘l’ | ‘l’ | ‘o’ |  | ‘w’ | ‘o’ | ‘r’ | ‘l’ |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

Syntax:- S=[0:5]

Example: s1=hello

Print(s[:])

**Output:**-‘hello’

**Behind Logic:-**

**Condition:-**

Index < End

Index=0;

End value=5

0<5------🡪true

Current index=0 New index = current index+ step value

=0+1

=1

1<5----------🡪true

Current index=1 New index = current index+ step value

=1+1

=2

2<5----------------------🡪-true

Current index=2 New index = current index+ step value

=2+1

=3

3<5----------------------------🡪true

Current index=3 New index = current index+ step value

=3+1

=4

4<5---------------------------🡪true

Current index=4 New index = current index+ step value

=4+1

=5

5<5------------------------🡪false

Example :-odd number:-

S=’123456789’

S[::2]

**Slicing using negative index:-**

Example:-s=’abcdefghij’

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ‘a’ | ‘b’ | ‘c’ | ‘d’ | ‘e’ | ‘f’ | ‘g’ | ‘h’ | ‘I’ | ‘j’ |
| -10 | -9 | -8 | -7 | -6 | -5 | -4 | -3 | -2 | -1 |

S[-10:-4]

‘abcdef’

2. S[:-4]

‘abccdef’

3. S[-10:]

‘Abcdefghij’

**Basic Rules of slicing:-**

* If you want to give start values=0 in slicing it is not necessary to give, because default value for start value is 0.

Ex:-S[:10] (or) s[0:10] both are same

* If you want to give end value length of string in slicing no need to give, because default value for end value is length of string.

Ex:-S[3:] (or) s[3:11] both are same

* If you want to give start value=0 and end value=length=length of string, no need to give ,because default value for start value is 0 and end value is length of string.

Ex:-S[:]

* If you want to give step value=1 in slicing, no need to give, because default value for step value is 1.

Ex:- S[:3:1] (or) s[:3]

* Don’t use positive value and negative value as start and end values in a string slicing query.