CONCATENATION (+) & REPLICATION (*)

(str, list, tuple)

CONCATENATION (+) (str, list, tuple)

Concatenation is represented by (+) operator

Concatenation (str):

Joining of two or more strings together is called Concatenation.

Note:

1. Only string and string can be concatenated.
2. String cannot be concatenated with int, float, list, tuple.

```
In [9]: "Python"+"Program"
Out[9]: 'PythonProgram'

In [10]: "Hi"+"Hello"
Out[10]: 'HiHello'

In [11]: "Hi"+"Hello"+"Welcome"
Out[11]: 'HiHelloWelcome'
```

```
In [4]: '10'+40
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Input In [4], in <cell line: 1>()
----> 1 '10'+40

TypeError: can only concatenate str (not "int") to str
```

```
In [5]: '10'+'50'
Out[5]: '1050'

In [6]: 1.5+20
Out[6]: 21.5
```

```
In [8]: '1.5'+33.3
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Input In [8], in <cell line: 1>()
----> 1 '1.5'+33.3

TypeError: can only concatenate str (not "float") to str
```

```
In [14]: "Python"+3
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Input In [14], in <cell line: 1>()
----> 1 "Python"+3

TypeError: can only concatenate str (not "int") to str
```

```
In [15]: "Python"+"3.1"
```

```
Out[15]: 'Python3.1'
```

```
In [17]: "Python"+" "+"3"
```

```
Out[17]: 'Python 3'
```

```
In [21]: print("Hello"+"Welcome")
```

```
HelloWelcome
```

```
In [22]: print(49+11)
```

```
60
```

```
In [23]: print("49"+'11')
```

```
4911
```

```
In [18]: "Python"+["abcd"]
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Input In [18], in <cell line: 1>()
----> 1 "Python"+["abcd"]

TypeError: can only concatenate str (not "list") to str
```

```
In [19]: [1,2,3]+"abcd"
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Input In [19], in <cell line: 1>()
----> 1 [1,2,3]+"abcd"

TypeError: can only concatenate list (not "str") to list
```

Concatenation (list):

Joining of two or more lists together is called Concatenation of Lists.

Note:

1. Only list and list can be concatenated.
2. list cannot be concatenated with int, float, str, tuple.

```
In [24]: [1,2,3]+[4,5]
Out[24]: [1, 2, 3, 4, 5]

In [25]: ['a','b','c']+[1,2,3,4,5]
Out[25]: ['a', 'b', 'c', 1, 2, 3, 4, 5]

In [26]: [100,200]+(20,200,300)
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Input In [26], in <cell line: 1>()
----> 1 [100,200]+(20,200,300)

TypeError: can only concatenate list (not "tuple") to list
```

```
In [28]: print([1,1,1]+[2,2,2])
[1, 1, 1, 2, 2, 2]

In [29]: print([1,2]+[4,5])
[1, 2, 4, 5]
```

Concatenation (tuple):

Joining of two or more tuples together is called Concatenation of Tuples.

Note:

1. Only tuple and tuple can be concatenated.
2. tuple cannot be concatenated with int, float, str, list.

```
In [30]: ('x','y','z')+('a','b','c')
Out[30]: ('x', 'y', 'z', 'a', 'b', 'c')

In [32]: print((9,9)+(10,10))
         (9, 9, 10, 10)

In [33]: (1,2,3)+(10,20,30)
Out[33]: (1, 2, 3, 10, 20, 30)

In [34]: (90,29,11)+[68,5,44]
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Input In [34], in <cell line: 1>()
----> 1 (90,29,11)+[68,5,44]

TypeError: can only concatenate tuple (not "list") to tuple

In [36]: print((1,2,3)+"123")
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Input In [36], in <cell line: 1>()
----> 1 print((1,2,3)+"123")

TypeError: can only concatenate tuple (not "str") to tuple
```

REPLICATION (*) (str, list, tuple)

Replication is represented by (*) operator

Replication (str):

Making copies of a single string, list, tuple to multiple (n-number of times) is called Replication.

```
In [38]: "word"*3
Out[38]: 'wordwordword'

In [39]: "20"*2
Out[39]: '2020'

In [40]: 'Zz'*10
Out[40]: 'ZzZzZzZzZzZzZzZzZzZz'

In [45]: [5,10,15,20]*2
Out[45]: [5, 10, 15, 20, 5, 10, 15, 20]

In [46]: (7,7,7)*3
Out[46]: (7, 7, 7, 7, 7, 7, 7, 7, 7)
```

```
In [50]: "bye"*2.5
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Input In [50], in <cell line: 1>()
----> 1 "bye"*2.5

TypeError: can't multiply sequence by non-int of type 'float'

In [51]: [1,2]*1.5
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Input In [51], in <cell line: 1>()
----> 1 [1,2]*1.5

TypeError: can't multiply sequence by non-int of type 'float'
```