

```
String operations

In [1]: "Micheal Jackson"
Out[1]: 'Micheal Jackson'

In [2]: 'Micheal Jackson'
Out[2]: 'Micheal Jackson'

In [3]: '1 2 3 4 5 6 '
Out[3]: '1 2 3 4 5 6 '

In [4]: '@#2_#]&*$%'
Out[4]: '@#2_#]&*$%'

In [5]: print("hello!")
hello!

In [6]: name="Michael Jackson"
name
Out[6]: 'Michael Jackson'

In [7]: print(name[0])
M

In [8]: print(name[6])
l

In [9]: print(name[13])
o

In [10]: print(name[-1])
n

In [11]: print(name[-15])
M

In [12]: len("Michael Jackson")
Out[12]: 15

In [13]: name[0:4]
Out[13]: 'Mich'

In [14]: name[8:12]
Out[14]: 'Jack'

In [16]: name[::2]
Out[16]: 'McalJcsn'

In [17]: name[0:5:2]
Out[17]: 'Mca'

In [20]: s =name + "is the best"
s
Out[20]: 'Michael Jacksonis the best'

In [21]: 3*"Michael Jackson"
Out[21]: 'Michael JacksonMichael JacksonMichael Jackson'

In [22]: name="Michael Jackson"
name=name + "is the best"
name
Out[22]: 'Michael Jacksonis the best'

In [24]: print("Michael Jackson \n is the best")
Michael Jackson
is the best

In [25]: print("Michael Jackson \t is the best")
Michael Jackson      is the best

In [28]: print("Michael Jackson \\ is the best")
Michael Jackson \ is the best

In [29]: print(r"Michael Jackson \ is the best")
Michael Jackson \ is the best

In [30]: a= "Thriller is the sixth studio album"
print("before upper:",a)
b=a.upper()
print("After upper:",b)
before upper: Thriller is the sixth studio album
After upper: THRILLER IS THE SIXTH STUDIO ALBUM

In [31]: a="Michael Jackson is the best"
b=a.replace('Michael','Janet')
b
Out[31]: 'Janet Jackson is the best'

In [32]: name="Michael Jackson"
name.find('el')
Out[32]: 5

In [33]: name.find('Jack')
Out[33]: 8

In [34]: name.find('Jasdfasdasdf')
Out[34]: -1

Tuples in python

In [35]: tuple1=("disco",10,1.2)
tuple1
Out[35]: ('disco', 10, 1.2)

In [36]: type(tuple1)
Out[36]: tuple

In [37]: print(tuple1[0])
print(tuple1[1])
print(tuple1[2])
disco
10
1.2

In [38]: print(type(tuple1[0]))
print(type(tuple1[1]))
print(type(tuple1[2]))
<class 'str'>
<class 'int'>
<class 'float'>

In [39]: tuple1[-1]
Out[39]: 1.2

In [40]: tuple1[-2]
Out[40]: 10

In [41]: tuple1[-3]
Out[41]: 'disco'

In [42]: tuple2=tuple1 +("hard rock",10)
tuple2
Out[42]: ('disco', 10, 1.2, 'hard rock', 10)

In [43]: tuple2[0:3]
Out[43]: ('disco', 10, 1.2)

In [44]: tuple2[3:5]
Out[44]: ('hard rock', 10)

In [45]: len(tuple2)
Out[45]: 5

In [46]: Ratings=(0,9,6,5,10,8,9,6,2)

In [47]: RatingsSorted=sorted(Ratings)
RatingsSorted
Out[47]: [0, 2, 5, 6, 6, 8, 9, 9, 10]

In [48]: NestedT=(1,2,("pop","rock"),(3,4),("disco",(1,2)))

In [49]: print("Element 0 of Tuple:",NestedT[0])
print("Element 1 of Tuple:",NestedT[1])
print("Element 2 of Tuple:",NestedT[2])
print("Element 3 of Tuple:",NestedT[3])
print("Element 4 of Tuple:",NestedT[4])
Element 0 of Tuple: 1
Element 1 of Tuple: 2
Element 2 of Tuple: ('pop', 'rock')
Element 3 of Tuple: (3, 4)
Element 4 of Tuple: ('disco', (1, 2))

In [50]: print("Element 2,0 of Tuple:",NestedT[2][0])
print("Element 2,1 of Tuple:",NestedT[2][1])
print("Element 3,0 of Tuple:",NestedT[3][0])
print("Element 3,1 of Tuple:",NestedT[3][1])
print("Element 4,0 of Tuple:",NestedT[4][0])
print("Element 4,1 of Tuple:",NestedT[4][1])
Element 2,0 of Tuple: pop
Element 2,1 of Tuple: rock
Element 3,0 of Tuple: 3
Element 3,1 of Tuple: 4
Element 4,0 of Tuple: disco
Element 4,1 of Tuple: (1, 2)

In [52]: NestedT[2][1][0]
Out[52]: 'r'

In [53]: NestedT[2][1][1]
Out[53]: 'o'

In [54]: NestedT[4][1][0]
Out[54]: 2

In [55]: NestedT[4][1][1]
Out[55]: 2
```