

In [1]: `pwd`

Out[1]: 'D:\\Python\_Practice\\Jupyter'

In [2]: `a=10  
b=10.5  
c="Anik"`

In [5]: `type(c)`

Out[5]: `str`

In [6]: `a,b,c=10,10.5,"Anik"`

In [7]: `a`

Out[7]: `10`

In [8]: `b`

Out[8]: `10.5`

In [9]: `c`

Out[9]: `'Anik'`

In [10]: `import keyword  
print(keyword.kwlist)`

`['False', 'None', 'True', 'and', 'as', 'assert', 'async', 'await', 'break', 'class', 'continue', 'def', 'del', 'elif', 'else', 'except', 'finally', 'for', 'from', 'global', 'if', 'import', 'in', 'is', 'lambda', 'nonlocal', 'not', 'or', 'pass', 'raise', 'return', 'try', 'while', 'with', 'yield']`

In [11]: `a=float(9)`

In [12]: `type(a)`

Out[12]: `float`

```
In [13]: a=10  
b=10.5  
c="Anik"
```

```
In [14]: print(a)  
print(b)  
print(c)
```

```
10  
10.5  
Anik
```

```
In [15]: print(a,b,c)
```

```
10 10.5 Anik
```

```
In [17]: print("value of a is",a,"value of b is",b,"value of c is",c)
```

```
value of a is 10 value of b is 10.5 value of c is Anik
```

```
In [21]: print("value of a is %d, value of b is %f, value of c is %s " %(a,b,c))
```

```
value of a is 10, value of b is 10.500000, value of c is Anik
```

```
In [23]: print("value of a is {}, value of b is {}, value of c is {}".format(a,b,c))
```

```
value of a is 10, value of b is 10.5, value of c is Anik
```

```
In [24]: #1st printing val of c then a then b  
print("value of a is {2}, value of b is {0}, value of c is {1}".format(a,b,c))
```

```
value of a is Anik, value of b is 10, value of c is 10.5
```

```
In [31]: # type casting coz input take as string
a=int(input("Enter a val "))
b=int(input("Enter b val "))
c=a+b
print(c)
```

```
Enter a val 10
Enter b val 20
30
```

```
In [32]: # a is int but converting it into float and storing in b(type casting)
a=40
b=float(a)
```

```
In [33]: type(b)
```

```
Out[33]: float
```

```
In [36]: a=int(input("Enter value of a "))
b=int(input("Enter value of b "))
c=int(input("Enter value of c "))
d=int(input("Enter value of d "))

add=a+b
sub=a-b
mul=c*d
div=c/d

print("add is %d, sub is %d"%(add,sub))
print("Value of mul is {}, value of div is{} ".format(mul,div))
```

```
Enter value of a 30
Enter value of b 10
Enter value of c 20
Enter value of d 10
add is 40,sub is 20
Value of mul is 200, value of div is2.0
```

```
In [50]: string='hello welcome to smartbridge 1 2 3'
```

```
In [46]: string[26]
```

```
Out[46]: 'g'
```

```
In [51]: string[10:]
```

```
Out[51]: 'ome to smartbridge 1 2 3'
```

```
In [52]: string[:17] #n-1
```

```
Out[52]: 'hello welcome to '
```

```
In [53]: string[10:17:2]
```

```
Out[53]: 'oet '
```

```
In [58]: string="1,2,3,4,5,6,7,8,"
```

```
In [56]: string[2:7:2]
```

```
Out[56]: '234'
```

```
In [62]: string[-8:-2]
```

```
Out[62]: '5,6,7,'
```

In [93]: `dir(string)`

Out[93]: ['`__add__`',  
'`__class__`',  
'`__contains__`',  
'`__delattr__`',  
'`__dir__`',  
'`__doc__`',  
'`__eq__`',  
'`__format__`',  
'`__ge__`',  
'`__getattribute__`',  
'`__getitem__`',  
'`__getnewargs__`',  
'`__gt__`',  
'`__hash__`',  
'`__init__`',  
'`__init_subclass__`',  
'`__iter__`',  
'`__le__`',  
'`__len__`',  
'`__lt__`',  
'`__mod__`',  
'`__mul__`',  
'`__ne__`',  
'`__new__`',  
'`__reduce__`',  
'`__reduce_ex__`',  
'`__repr__`',  
'`__rmod__`',  
'`__rmul__`',  
'`__setattr__`',  
'`__sizeof__`',  
'`__str__`',  
'`__subclasshook__`',  
`'capitalize'`,  
`'casefold'`,  
`'center'`,  
`'count'`,  
`'encode'`,  
`'endswith'`,  
`'expandtabs'`,  
`'find'`,

```
'format',
'format_map',
'index',
'isalnum',
'isalpha',
'isascii',
'isdecimal',
'isdigit',
'isidentifier',
'islower',
'isnumeric',
'isprintable',
'isspace',
'istitle',
'isupper',
'join',
'ljust',
'lower',
'rstrip',
'maketrans',
'partition',
'replace',
'find',
'rindex',
'rjust',
'rpartition',
'rsplit',
'rstrip',
'split',
'splitlines',
'startswith',
'strip',
'swapcase',
'title',
'translate',
'upper',
'zfill']
```

In [82]: `string2="hie,how,are,you, how,how"`

```
In [65]: string2.capitalize()
```

```
Out[65]: 'Hie how are you'
```

```
In [66]: string2.casefold()
```

```
Out[66]: 'hie how are you'
```

```
In [76]: string2.split(',')
```

```
Out[76]: ['hie', 'how', 'are', 'you']
```

```
In [77]: string2.center(40,"*")
```

```
Out[77]: '*****hie,how,are,you*****'
```

```
In [83]: string2.count("how")#how many 'how' are there
```

```
Out[83]: 1
```

```
In [84]: string2.count("how",10,20)#how many 'how' are there from index 10-20
```

```
Out[84]: 1
```

```
In [101]: list1=[1,10.5,"John","Anik"]
```

```
In [87]: list1[2]
```

```
Out[87]: 'John'
```

```
In [88]: list1[1:5]
```

```
Out[88]: [10.5, 'John', 'Anik']
```

```
In [89]: list1[2]="Smartbridge"
```

```
In [90]: list1
```

```
Out[90]: [1, 10.5, 'Smartbridge', 'Anik']
```

In [92]: `dir(list1)`

Out[92]:

```
[ '__add__',
  '__class__',
  '__contains__',
  '__delattr__',
  '__delitem__',
  '__dir__',
  '__doc__',
  '__eq__',
  '__format__',
  '__ge__',
  '__getattribute__',
  '__getitem__',
  '__gt__',
  '__hash__',
  '__iadd__',
  '__imul__',
  '__init__',
  '__init_subclass__',
  '__iter__',
  '__le__',
  '__len__',
  '__lt__',
  '__mul__',
  '__ne__',
  '__new__',
  '__reduce__',
  '__reduce_ex__',
  '__repr__',
  '__reversed__',
  '__rmul__',
  '__setattr__',
  '__setitem__',
  '__sizeof__',
  '__str__',
  '__subclasshook__',
  'append',
  'clear',
  'copy',
  'count',
  'extend',
  'index',
```

```
'insert',
'pop',
'remove',
'reverse',
'sort']
```

In [111]: list1.append([1,2,3,4,5,6,7,8])

In [112]: list1

Out[112]: [1, 10.5, 'John', 'Anik', 1, 2, 3, 4, [1, 2, 3, 4, 5, 6, 7, 8]]

In [96]:

In [97]:

Out[97]: []

In [103]: list2=list1.copy()

In [104]: list2

Out[104]: [1, 10.5, 'John', 'Anik']

In [105]: list1.count(10.5)

Out[105]: 1

In [106]: list1.extend([1,2,3,4])

In [113]: list1

Out[113]: [1, 10.5, 'John', 'Anik', 1, 2, 3, 4, [1, 2, 3, 4, 5, 6, 7, 8]]

In [110]: list1.index('John')

Out[110]: 2

```
In [114]: list1.insert(2,40)# 2nd index 40 inserting
```

```
In [115]: list1
```

```
Out[115]: [1, 10.5, 40, 'John', 'Anik', 1, 2, 3, 4, [1, 2, 3, 4, 5, 6, 7, 8]]
```

```
In [116]: list1.pop()#last element element and remoe used to remove a particualr index value
```

```
Out[116]: [1, 2, 3, 4, 5, 6, 7, 8]
```

```
In [117]: list1
```

```
Out[117]: [1, 10.5, 40, 'John', 'Anik', 1, 2, 3, 4]
```

```
In [118]: list1.reverse()
```

```
In [119]: list1
```

```
Out[119]: [4, 3, 2, 1, 'Anik', 'John', 40, 10.5, 1]
```

```
In [120]: list1=[1,8,9,10,5,3,6]
```

```
In [121]: list1.sort()
```

```
In [122]: list1
```

```
Out[122]: [1, 3, 5, 6, 8, 9, 10]
```

```
In [123]: list1[5:-1]
```

```
Out[123]: [9]
```

```
In [125]: list1=list1[5:-1]
```

```
In [126]: list1
```

```
Out[126]: [9]
```

```
In [127]: list3=[i for i in range(10)] # comprehensive technique
```

```
In [128]: list3
```

```
Out[128]: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

```
In [130]: list3=[i for i in range(10)if(i%2==0)]
```

```
In [131]: list3
```

```
Out[131]: [0, 2, 4, 6, 8]
```

```
In [132]: tuple=(1,2,3,"john","Anik")
```

In [133]: `dir(tuple)`

Out[133]: `[ '__add__',  
 '__class__',  
 '__contains__',  
 '__delattr__',  
 '__dir__',  
 '__doc__',  
 '__eq__',  
 '__format__',  
 '__ge__',  
 '__getattribute__',  
 '__getitem__',  
 '__getnewargs__',  
 '__gt__',  
 '__hash__',  
 '__init__',  
 '__init_subclass__',  
 '__iter__',  
 '__le__',  
 '__len__',  
 '__lt__',  
 '__mul__',  
 '__ne__',  
 '__new__',  
 '__reduce__',  
 '__reduce_ex__',  
 '__repr__',  
 '__rmul__',  
 '__setattr__',  
 '__sizeof__',  
 '__str__',  
 '__subclasshook__',  
 'count',  
 'index']`

In [135]: `tuple.count(1)`

Out[135]: 1

In [137]: `tuple.index("Anik")`

Out[137]: 4

In [ ]: