Gautham Santhosh Follow
Maker 🚀 Student 🎓
Aug 1 · 3 min read

Python Tricks 101💪

Python tips which are relatively uncommon and are very useful.

> *Talk is cheap show me the code*

Swapping values

```
"""pythonic way of value swapping"""
a, b = 5, 10
print(a, b)
a, b = b, a
print(a, b)
```

Create a single string from all the elements in list

```
a = ["Python", "is", "awesome"]
print(" ".join(a))
```

Find The Most Frequent Value In A List.

```
""" most frequent element in a list """

a = [1, 2, 3, 1, 2, 3, 2, 2, 4, 5, 1]

print(max(set(a),key = a.count))

""" using Counter from collections """

from collections import Counter

cnt = Counter(a)
print(cnt.most_common(3))
```

Checking if two words are anagrams

```
from collections import Counter

Counter(str1) == Counter(str2)
```

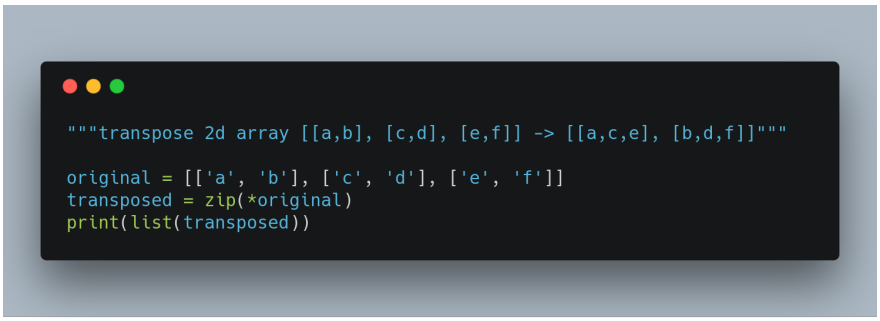
Reverse a String

```
"""reversing string with special case of slice step param"""

a = 'abcdefghijklmnopqrstuvwxyz'
print(a[::-1])


"""iterating over string contents in reverse efficiently."""

for char in reversed(a):
    print(char)

"""reversing an integer through type conversion and slicing."""

num = 123456789
print(int(str(num)[::-1]))
```

Reverse a list

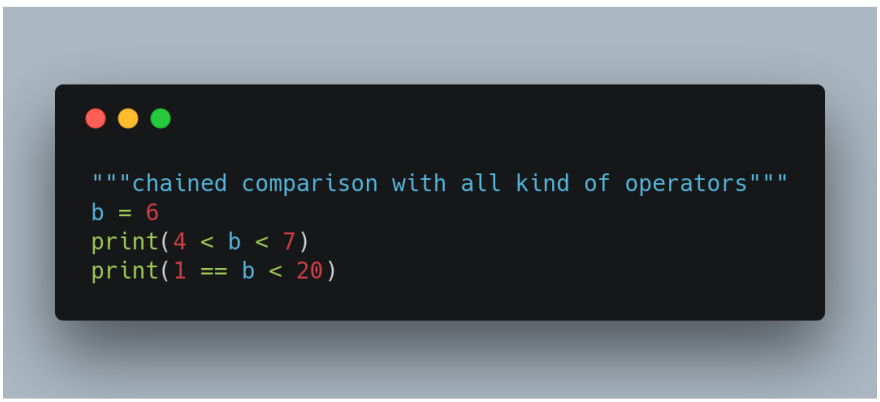
```
"""reversing list with special case of slice step param"""
a = [5, 4, 3, 2, 1]
print(a[::-1])

"""iterating over list contents in reverse efficiently."""
for ele in reversed(a):
    print(ele)
```

Transpose 2d array



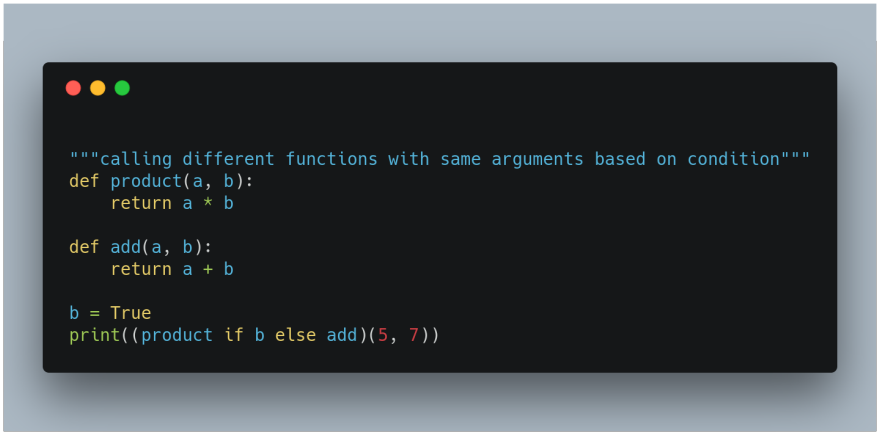
```
"""transpose 2d array [[a,b], [c,d], [e,f]] -> [[a,c,e], [b,d,f]]"""

original = [['a', 'b'], ['c', 'd'], ['e', 'f']]
transposed = zip(*original)
print(list(transposed))
```

Chained Comparison



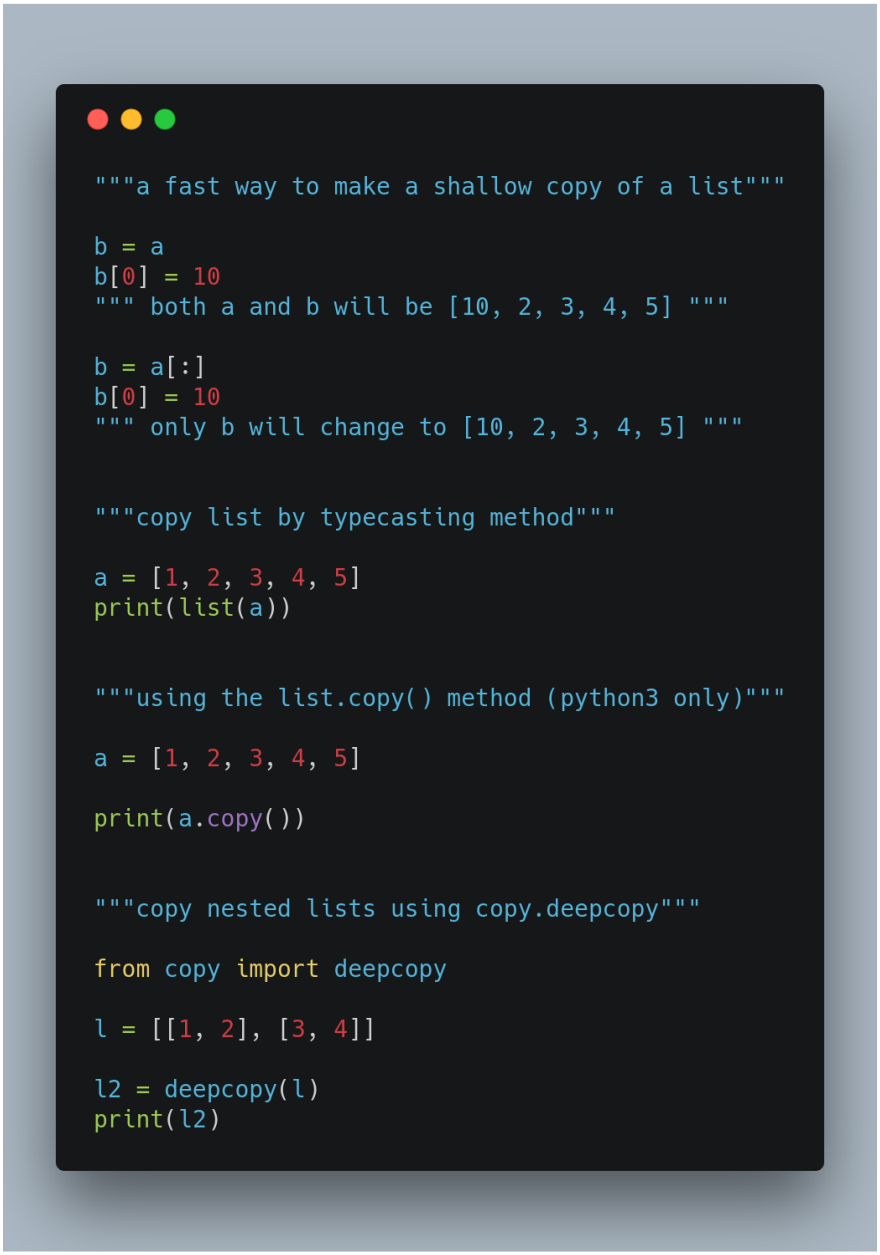
```
"""chained comparison with all kind of operators"""
b = 6
print(4 < b < 7)
print(1 == b < 20)
```

Chained function call



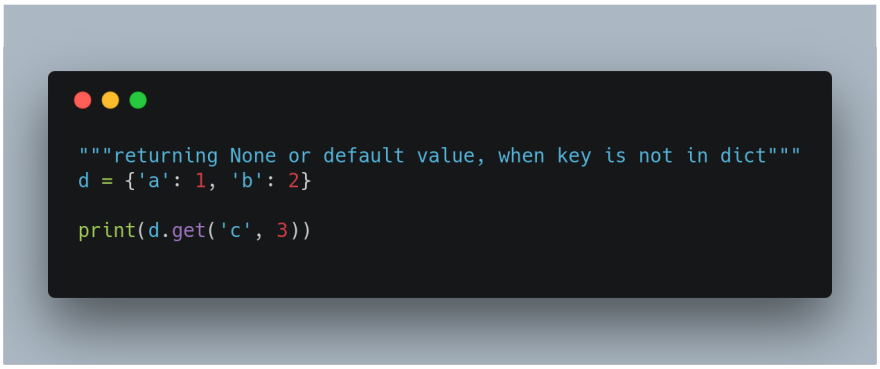
```
"""calling different functions with same arguments based on condition"""
def product(a, b):
    return a * b

def add(a, b):
    return a + b

b = True
print((product if b else add)(5, 7))
```

Copying List



```
"""a fast way to make a shallow copy of a list"""

b = a
b[0] = 10
""" both a and b will be [10, 2, 3, 4, 5] """

b = a[:]
b[0] = 10
""" only b will change to [10, 2, 3, 4, 5] """


"""copy list by typecasting method"""

a = [1, 2, 3, 4, 5]
print(list(a))


"""using the list.copy() method (python3 only)"""

a = [1, 2, 3, 4, 5]

print(a.copy())


"""copy nested lists using copy.deepcopy"""

from copy import deepcopy

l = [[1, 2], [3, 4]]

l2 = deepcopy(l)
print(l2)
```

Dictionary Get

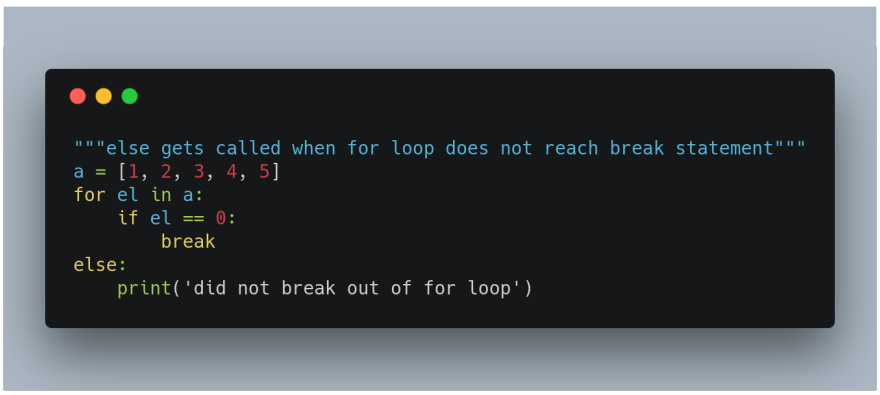


```
"""returning None or default value, when key is not in dict"""
d = {'a': 1, 'b': 2}

print(d.get('c', 3))
```

Sort Dictionary by Value

```
""" Sort a dictionary by its values with the built-in sorted() function and a 'key' argument. """

d = {'apple': 10, 'orange': 20, 'banana': 5, 'rotten tomato': 1}
print(sorted(d.items(), key=lambda x: x[1]))


""" Sort using operator.itemgetter as the sort key instead of a lambda"""


from operator import itemgetter


print(sorted(d.items(), key=itemgetter(1)))


"""Sort dict keys by value"""


print(sorted(d, key=d.get))
```

For Else



```
"""else gets called when for loop does not reach break statement"""
a = [1, 2, 3, 4, 5]
for el in a:
    if el == 0:
        break
else:
    print('did not break out of for loop')
```

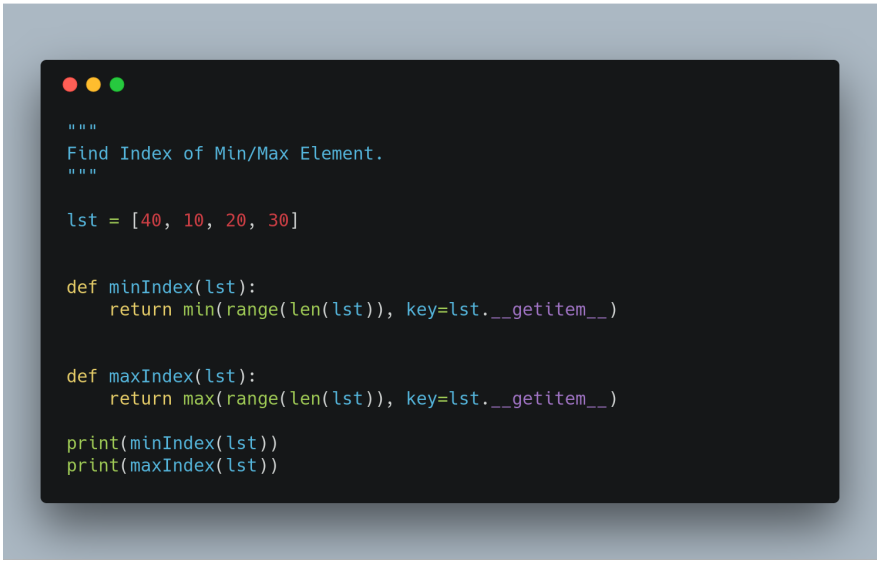
Convert list to comma separated

```
"""converts list to comma separated string"""

items = ['foo', 'bar', 'xyz']

print (','.join(items))

"""list of numbers to comma separated"""
numbers = [2, 3, 5, 10]

print (','.join(map(str, numbers)))

"""list of mix  data"""
data = [2, 'hello', 3, 3.4]

print (','.join(map(str, data)))
```

Merge dict's

```
"""merge dict's"""

d1 = {'a': 1}
d2 = {'b': 2}

#  python 3.5
print({**d1, **d2})

print(dict(d1.items() | d2.items()))

d1.update(d2)
print(d1)
```

Min and Max index in List



```
"""
Find Index of Min/Max Element.
"""

lst = [40, 10, 20, 30]


def minIndex(lst):
    return min(range(len(lst)), key=lst.__getitem__)


def maxIndex(lst):
    return max(range(len(lst)), key=lst.__getitem__)

print(minIndex(lst))
print(maxIndex(lst))
```

Remove duplicates from a list

```
"""remove duplicate items from list. note: does not preserve the original list order"""

items = [2, 2, 3, 3, 1]

newitems2 = list(set(items))
print(newitems2)

"""remove dups and keep order"""

from collections import OrderedDict

items = ["foo", "bar", "bar", "foo"]

print(list(OrderedDict.fromkeys(items).keys()))
```

If you think i should i add anymore or have suggestions please do comment. i'll keep on updating this blog.

Most of the tricks and tips are taken from PyTricks and a few blogs .

Thank you for reading 🤣*. If you like the Article give it a clap* 👏.

Do consider Buying me a Coffee https://www.buymeacoffee.com/gautham , If you loved the article.

If you wish to have a chat, DM me at https://twitter.com/gauthamzzz.

I am a Masters student at the Indian Institute of Information Technology, Allahabad. My Website http://gauthamzz.com.