In [1]:

```

```

In [4]:

```
#Trick 1: Flatten the lists

import itertools
a = [[1, 2], [3, 4], [5, 6]]
b = list(itertools.chain.from_iterable(a))
print(b)
```

```
[1, 2, 3, 4, 5, 6]
```

In [9]:

```
#Trick 2: Reverse a list

a=["10","9","8","7"]
print(a[::-1])
```

```
['7', '8', '9', '10']
```

In [15]:

```
#Trick 3: Combining different lists
a=["a","b","c","d"]
b=["e","f","g","h"]
for x, y in zip(a, b):
   print(x,y)
```

```
a e
b f
c g
d h
```

In [16]:

```
#Trick 4: Negative indexing lists
a = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
a[-3:-1]
```

Out[16]:

```
[8, 9]
```

In [17]:

```
#Trick 5: Analyzing the most frequent on the list
a = [1, 2, 3, 4, 2, 2, 3, 1, 4, 4, 4]
print(max(set(a), key = a.count))
```

```
4
```

In [23]:

```
#Trick 6: Reversing the string
a="python"
print("Reverse is", a[::-1])
```

```
Reverse is nohtyp
```

In [24]:
```
#Trick 7: Splitting the string
a="Python is the language of the future"
b=a.split()
print(b)
```

```
['Python', 'is', 'the', 'language', 'of', 'the', 'future']
```

In [29]:
```
#Trick 8: Printing out multiple values of strings
print("on"*3+''+"off"*2)


```

```
onononoffoff
```

In [33]:
```
#Trick 9: Creating a single string
a = ["I", "am", "not", "available"]
print(" " .join(a))
```

```
I am not available
```

In [36]:
```
#Trick 10: Checking if two words are anagrams
from collections import Counter
def is_anagram(str1, str2):
 return Counter(str1) == Counter(str2)
print(is_anagram('taste', 'state'))
print(is_anagram('beach', 'peach'))
```

```
True
False
```

In [39]:
```
#Trick 11: Transposing a matrix

mat = [[8, 9, 10], [11, 12, 13]]
new_mat=zip(*mat)
for row in new_mat:
 print(row)
```

```
(8, 11)
(9, 12)
(10, 13)
```

In [40]:
```
#Trick 12: Chaining comparison operators

a = 17
b = 21
c = 11
print(c < a)
print(a < b)
```

```
True
True
```

In [ ]: