

Program to identify if a string is Permutation of other provided string

In [1]:

```
# Conditions for permutations
# All the element that are present in one string should be in the other string
# 1.python - pythno -> a permutation
# 2.python - Python -> not a permutation
# 3.python - cython -> not a permutation
# 4.python - ppython -> not a permutation
# 5.python - pythop -> not a permutation
# 6.python - ton -> not a permutaiton

s1 = 'python'
s2 = 'ton'

def is_permutation(orange, apple):
    print(list(orange), list(apple))
    if len(list(orange)) == len(list(apple)):
        test_apple = apple
        test_orange = orange
        for i in orange:
            if i in test_apple:
                test_apple = test_apple.replace(i, '')
        for i in apple:
            if i in test_orange:
                test_orange = test_orange.replace(i, '')
        if len(test_apple) == len(test_orange) == 0:
            return 'Permutation'
        else:
            return 'Not a permutation'
    else:
        return 'Not a permutation by condition 6 and 4'

print(is_permutation(s1, s2))
```

```
['p', 'y', 't', 'h', 'o', 'n'] ['t', 'o', 'n']
Not a permutation by condition 6 and 4
```

In [3]:

```
s3 = 'be ar'
s4 = 'bare0'
is_permutation(s3, s4)
```

```
['b', 'e', ' ', 'a', 'r'] ['b', 'a', 'r', 'e', '0']
```

Out[3]:

```
'Not a permutation'
```

In [4]:

```
s5 = 'bear'  
s6 = 'bare'  
is_permutation(s5,s6)
```

```
['b', 'e', 'a', 'r'] ['b', 'a', 'r', 'e']
```

Out[4]:

```
'Permutation'
```