In [5]:
```
def is_Sublist(list_a, list_b):
    if len(list_b) == 0:
        return True
    if len(list_a) == 0:
        return False
    if list_b == list_a:
        return True
    if len(list_b) > len(list_a):
        return False
    for i in range(len(list_a)):
        if list_a[i] == list_b[0]:
            if list_a[i:i+len(list_b)] == list_b:
                return True
    return False
a = [2, 4, 3, 5, 7]
b = [4, 3]
c = [3, 7]
print(is_Sublist(a, b))
print(is_Sublist(a, c))
```

```
True
False
```

In [7]:
```
def find_common_items(list1, list2):
    set1 = set(list1)
    set2 = set(list2)
    common_items = set1.intersection(set2)
    return common_items
color1 = ["Red", "Green", "Orange", "White"]
color2 = ["Black", "Green", "White", "Pink"]
common_colors = find_common_items(color1, color2)
print(common_colors)
```

```
{'Green', 'White'}
```

In [9]:
```
def get_list_difference(list1, list2):
    set1 = set(list1)
    set2 = set(list2)
    difference = set1.difference(set2)
    difference_list = list(difference)
    return difference_list
list1 = [1, 2, 3, 4]
list2 = [1, 2]
result = get_list_difference(list1, list2)
print(result)
```

```
[3, 4]
```

In [10]:
```
import itertools

def generate_permutations(lst):
    permutations = list(itertools.permutations(lst))
    return permutations
input_list = [1, 2, 3]
permutations = generate_permutations(input_list)
print(permutations)
```

```
[(1, 2, 3), (1, 3, 2), (2, 1, 3), (2, 3, 1), (3, 1, 2), (3, 2, 1)]
```

In [11]:
```
def remove_duplicates(lst):
    unique_list = list(set(lst))
    return unique_list
a = [10, 20, 30, 20, 10, 50, 60, 40, 80, 50, 40]
result = remove_duplicates(a)
print(result)
```

```
[40, 10, 80, 50, 20, 60, 30]
```