Set

In [ ]:
```
# Removing duplicates from a list
my_list = [1, 2, 2, 3, 4, 4, 5]
```

In [23]:
```
my_list_no_duplicates = list(set(my_list))
print(my_list_no_duplicates)
```

In [24]:
```
# set does not allow duplicate values after set(mylist) values are [1,
print(my_list_no_duplicates)
```

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

OrderedDict

In [45]:
```
# Removing duplicates from a list
my_list = [1, 2, 2, 3, 4, 4, 5]
from collections import OrderedDict
```

```
Each key in an OrderedDict must be unique, and when you assign a
value to an existing key, it updates the value associated with that
key rather than creating a new key with the same name.
```

In [46]:
```
# Removing duplicates from a list while preserving order
#O_dist=OrderedDict.fromkeys(my_list)
#print(O_dist)
```

In [47]:
```
# Iterating over the OrderedDict
#for key, value in O_dist.items():
#print(key, value)
```

In [48]:
```
my_list_no_duplicates = list(OrderedDict.fromkeys(my_list))
print(my_list_no_duplicates)
```

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

For Loop

In [49]:
```
# Removing duplicates from a list
my_list = [1, 2, 2, 3, 4, 4, 5]
```

In [50]:
```
Empty_List=[]
for i in my_list:
    if i not in Empty_List:
        Empty_List.append(i)
print(Empty_List)
```

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

Create a Function

In [51]:
```
def remove_duplicates(my_list):
    Empty_List = []
    for i in my_list:
        if i not in Empty_List:
            Empty_List.append(i)
    return Empty_List

# Example usage:
my_list = [1, 2, 2, 3, 4, 4, 5]
result = remove_duplicates(my_list)
print(result)
```

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

In [ ]: