# Operation between mutable and immutable

***Accessing or unpacking is possible in immutable as well as in mutable.***

***But modifying is not possible with immutable. Because we cannot change their internal state once they are created.***

Immutable datatypes used same storage location to store the similar data for memory utilization and to reduce memory fragmentation.

* Some of the **mutable** data types in Python are **list, dictionary, set** and **user-defined classes**.
* Some of the **immutable** data types are **int, float, decimal, bool, string, tuple, and range**.

***Immutable Data Types***

a = (1, 2, 3)

b = (1, 2, 3)

print(a is b) # True check for id of the element

print(a == b) # True check individual element of the array

print(id(a), id(b)) # 23403616 23403616

a = 'saranj'

b = 'saranj'

print(a is b) # True

print(a == b) # True

print(id(a), id(b)) # 30714688 30714688

Integers are also immutable, above operation is same for integers (i.e., for all immutable data type).

***Mutable data types***

a = [1, 2, 3]

b = [1, 2, 3]

print(a is b) # False

print(a == b) # True

print(id(a), id(b)) # 30426616 30427776