**Enumerate()**

enumerate() is a built-in function in Python that is used to iterate over a sequence (such as a list, tuple, or string) while keeping track of the index of the current item. It returns pairs of index and element, making it convenient to access both the index and the corresponding value in a loop.

Here's the basic syntax of enumerate():

Syntax:

**enumerate(iterable, start=0)**

**Eg:**fruits = ['apple', 'banana', 'orange', 'grape']

# Using enumerate in a loop to iterate over the list

for index, fruit in enumerate(fruits, start=1):

    print(f"Index {index}: {fruit}")

**Note:for index, fruit in ...:** This is a loop that iterates over each pair of (index, fruit) returned by enumerate. In each iteration, index holds the index of the current fruit, and fruit holds the value of the current fruit.

**Xrange()**

In Python, there is no **xrange()** function. Instead, there is **range()** function, which is used to generate a sequence of numbers. The **range()** function can be used in Python 2 and Python 3, but in Python 3, it behaves more like the Python 2 **xrange()** function.

**Muteable and Unmutable:**

**Mutable Objects:**

* **Definition:** Mutable objects are objects whose state or value can be changed after they are created.
* **Examples:** Lists, dictionaries, sets, and user-defined classes (instances of those classes if they are designed to be mutable).

**# Example of a mutable object (list)**

**my\_list = [1, 2, 3]**

**my\_list[0] = 99 # Modifying the content of the list**

**Immutable Objects:**

* **Definition:** Immutable objects are objects whose state or value cannot be changed after they are created.
* **Examples:** Integers, floats, strings, tuples, and frozensets.