**Module 14) Python – Collections, functions and Modules**

1. Understanding how to create and access elements in a list.

* A list is an **ordered**, **changeable (mutable)**, and can hold **duplicate values**
* You define one using **square brackets [...]**, allowing any data types—even mixed ones across items .

my\_list = [1, 2, 3]

fruits = ["apple", "banana", "cherry"]

mixed = [1, "two", 3.0, True]

* Accessing Elements by Index :
  + positive Indexing

Python lists are **zero-indexed**, i.e. first element is index 0.

Attempting to access an index outside the list range raises

* + negative Indexing

Negative indices count from the end: -1 is the last element, -2 is the second last

* + Lists are created with [] or list(iterable).
  + Access elements via my\_list[index]:
  + positive indexes from the start (0, 1, 2, …).
  + negative indexes from the end (−1, −2, …).

2 . Indexing in lists (positive and negative indexing).

* positive indexing :
  + starts at **0** for the first element, goes up by 1 for each subsequent element.
  + if lst = ['a', 'b', 'c', 'd']:

lst[0] : 'a'

lst[2] : 'c'

* + using an index ≥ length of the list raises an indexerror.
* negative indexing :
  + starts at **–1** for the **last element**, –2 for the second last, and so on.
  + for lst = ['a', 'b', 'c', 'd']:

lst[-1] : 'd'

lst[-3] : 'b'

* + this is useful when you don’t know the list’s length but need the end elements.
  + invalid negative indices (e.g. lst[-5] for a 4‑item list) also raise indexerror.

1. Slicing a list: accessing a range of elements.

* what is slicing?
* slicing is a way to extract a **subsequence** (a new list) from an existing list using the start:end:step pattern. it's written as:

sub = lst[start:end:step]

* **start**: index to begin (inclusive). defaults to 0 if omitted.
* **end**: index to stop (exclusive). defaults to len(lst) if omitted.
* **step**: increments between elements. defaults to 1.

Example :

fruits = ['a','b','c','d','e','f','g']

print(fruits[:3])

print(fruits[-4:-1])

print(fruits[::2])

print(fruits[::-2])

print(fruits[5:1:-1])