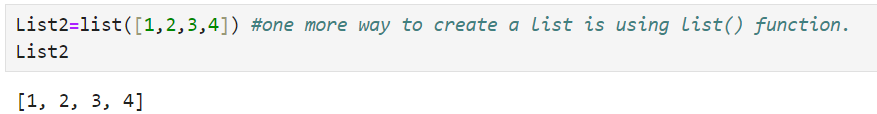
Day: 10

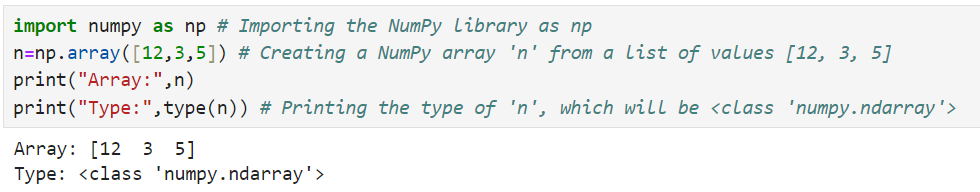
16th November, 2023



**Differentiate between List and Array:**

|  |  |  |
| --- | --- | --- |
| **Feature** | **List** | **Array** |
| **Definition** | A collection that can hold elements of different data types. | A collection of elements, typically of same data type. |
| **Data Types** | Can store elements of mixed data types (e.g: integers, strings, objects). | Typically stores elements of a single data type (e.g: all integers or all floats). |
| **Size** | Dynamic, can grow or shrink in size as needed. | Fixed size; once defined, it cannot be changed without creating a new array. |
| **Memory Efficiency** | Less memory efficient as it stores data along with references. | More memory efficient science elements are stored in contiguous memory location. |
| **Element Access** | Slower than arrays as it involves dynamic memory and references. | Faster, as elements are stored in continuous memory locations. |
| **Example:** | a=[“Hello”1,1,5] | A=[2,3,4,5] |

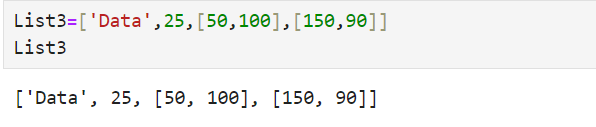
**Converting List to Array:**

****

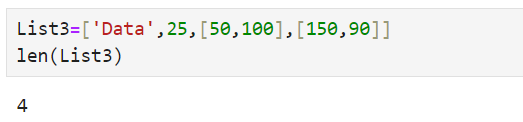
**Nested List:**

Nested List means list inside list.

Example:



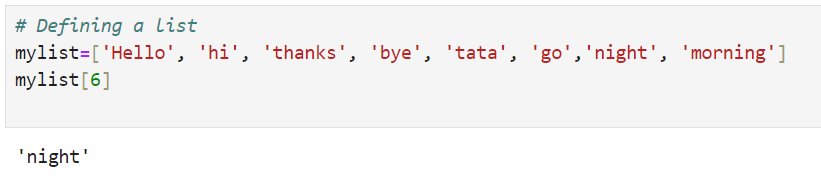
len() is used to check number of elements in list.



**List Indexing and Slicing:**

Indexing allows access to individual elements in a list by their position. In Python, list indices starts at 0 (the first element), and negative indices starts at -1(the last element).

Example:



Slicing allows access to a range of elements from a list.

The general syntax is: list[start:stop:step]

**Example:**



**Add, Remove and Change Items:**

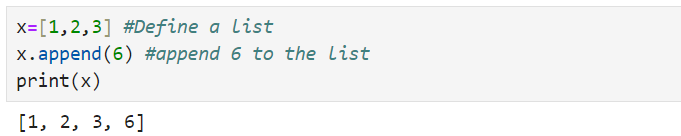
To add, remove and update items/elements from List, Python provides predefined/builtin functions:

**Adding in List:**

1. **append():**

This function adds a new element at the end of the list.

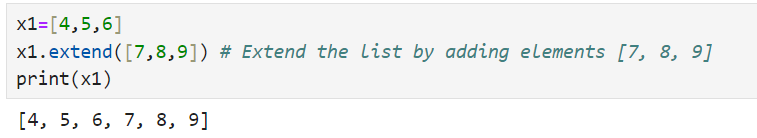
Example:



1. **extend():**

This function adds multiple elements together at the end of the list.

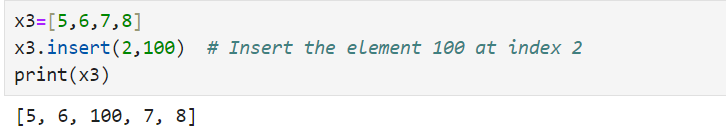
Example:



1. **insert():**

insert() will add a new element at a specific index location.

Example:

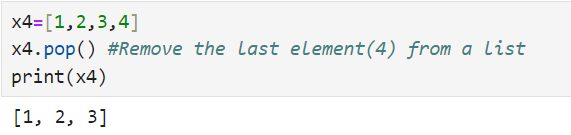


**Removing in List:**

1. **pop():**

It is used to remove last element by default in the list.

Example:

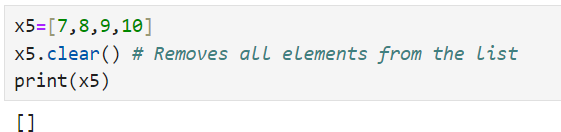




1. **clear():**

This function empties the entire list. All the elements will be deleted from list.

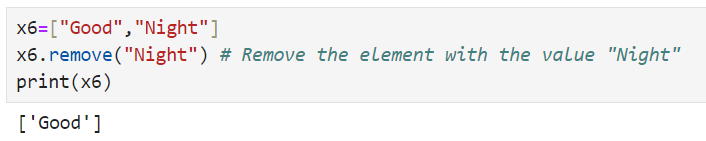
Example:



1. **remove():**

The remove() method in Python is used to remove a specific element from a list.

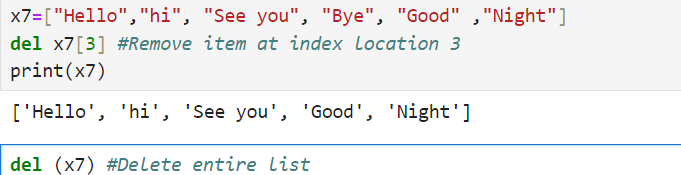
Example:



1. **del():**

The del statement in Python is used to delete objects. It can be used to delete variables, list elements, or even entire lists.

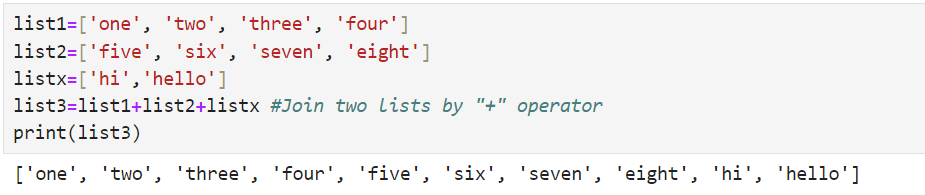
Example:



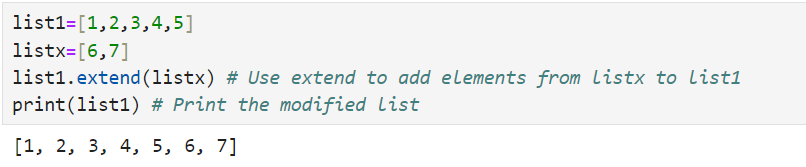
If you try to access x7 after it has been deleted, you'll encounter a NameError, indicating that the name x7 is not defined.

**Joining 2 Lists:**

1. **Using + operator:**



1. **Using extend():**

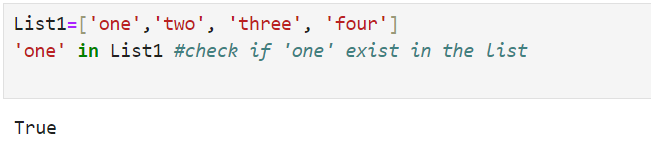
****

**List Membership:**

**in:**

‘in’ is membership operator and will check if given operator is present in sequence.

Example:



‘not in’ returns True if any element is not present in the list.

Example:

