

# Lists

⚙ Importance	★★★★★
≡ Author	CodingYogini
≡ Purpose	My Progress in learning Data Structures and Algorithms
≡ Youtube Channel	<a href="https://www.youtube.com/@CodingYogini">https://www.youtube.com/@CodingYogini</a>

## ▼ What is List?

1. List is a built in data structure.
  - a. Which means that it comes with the standard python library.
2. There are other built in data structures in python.
  - a. Tuple
  - b. Dictionary
  - c. Set
3. A list is a data structure that holds an ordered collection of items.
  - a. You can store a sequence of items in a list.
  - b. Ordered means that way that declare a list, it stays as it is place of elements is not changing.



### Example : Shopping list

where you have a list of items to buy except that you have each item on a separate line in your shopping list.

Whereas in python list you put commas between your items.

4. The list of items should be enclosed in square brackets so that python understands.
5. The values in list are called elements or items.

## ▼ Access and Traversal

### ▼ Syntax for Accessing Elements:

- The syntax for accessing the elements of a list is the same as for accessing the elements of an array.
- Use the bracket operator `[ ]` with the index inside to access elements.
- Example: `shoppingList[0]` will access the first element of the list.

### ▼ Key Points:

#### 1. Access Elements:

- Access elements through index number.
  - Example:

```
pythonCopy code
shoppingList = ['Milk', 'Cheese', 'Butter']
print(shoppingList[0])
print(shoppingList[1])
```

- Check if an element exists in the list using the `in` operator.
  - Example:

```
pythonCopy code
print('Milk' in shoppingList)
print('Bread' in shoppingList)
```

- Access elements from backward using Negative Index.
  - Example:

```
pythonCopy code
print(shoppingList[-1])
```

## 2. Traverse Elements:

- Traverse the elements of a list with a loop.
  - Example:

```
pythonCopy code
for item in shoppingList:
    print(item)
```

- Traverse the indexes of elements of a list with the `range()` function. This is useful to perform mathematical operations on the list.
  - Example:

```
pythonCopy code
for i in range(len(shoppingList)):
    shoppingList[i] = shoppingList[i] + "+"
    print(shoppingList[i])
```

- Traverse through an empty list.
  - Example:

```
pythonCopy code
empty = []
for item in empty:
    print("I am empty")
```

These notes provide an overview of list operations and accessing elements in Python, including syntax and key points for accessing and traversing lists.