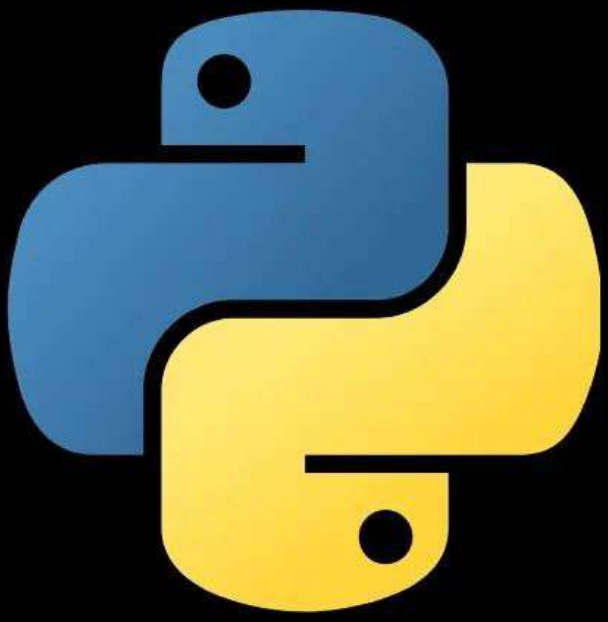
Python List
Simplified



01

Introduction:

Python List is used for storing multiple elements under one name.

List is created with comma- seperated values between square brackets.

```
my_list = [1, 2, 3, 4, 5, 6, 7]
```

A list that stores integer numbers.

02

List features:

Indexing:

We can use index values to access each element of list.

Mutable:

We can change the value of the list element.

Heterogeneus:

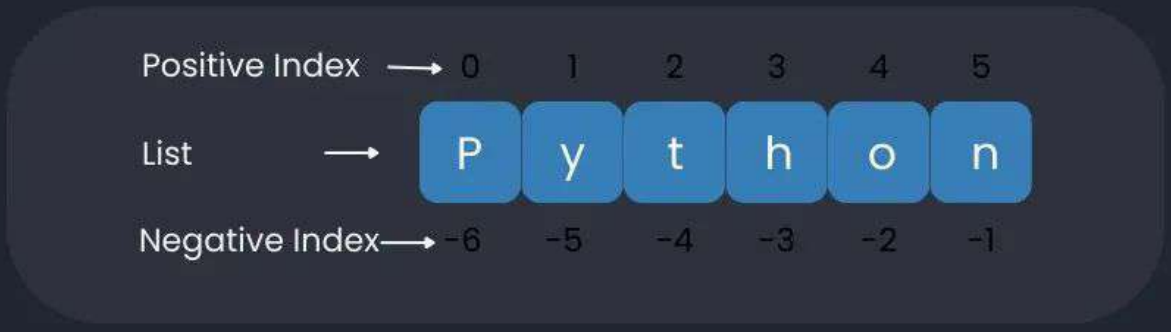
It can store more types of data elements in a single object.

03

List Indexing:

Index helps us to access each list element

Each list element is stored at a specific place inside a list and this place is termed an index. the index always starts from zero and ends at the total number of the element present in the list subtract 1 (n – 1).



04

List Indexing Example:

```python
my_list = ['element 1', 'element 2', 'element 3']

print("Positive Indexing")
print(my_list[0])
print(my_list[1])
print(my_list[2])

print("Negative Indexing")
print(my_list[-1])
print(my_list[-2])
print(my_list[-3])
```

```
Positive Indexing
element 1
element 2
element 3
Negative Indexing
element 3
element 2
element 1
```

Using the wrong index value will lead to IndexError

05

Mutability:

Mutable which means we can update or change the element value placed at a specific index.

```
my_list = ['element 1', 'element 2', 'element 3']

print(my_list)

# updating the value at index 0
my_list[0] = 'value 1'
print(my_list)
```

```
['element 1', 'element 2', 'element 3']
['value 1', 'element 2', 'element 3']
```

06

Heterogeneus:

The list allows us to store more than one type of data element in a single object/name.

```
my_list = ['element', 1, True, 3.14, 3+7j]
print(f'{my_list = }')

# Output:
# my_list = ['element', 1, True, 3.14, (3+7j)]
```

In the above code, the list stored string, int, float, bool and complex type of elements.

List Methods:

append(element)

Adds an element at the end of the list.

appendList.py

```
# Instagram : @dynamic.coding

test_list = [1, 2, 3, 4] # create list

test_list.append(5) # append operation
print(test_list)

# Output : [1, 2, 3, 4, 5]
```

insert(index, element)

Adds an element at the specified index in the list.

insertList.py

```
# Instagram : @dynamic.coding

test_list = [1, 2, 3, 4] # create list

test_list.insert(1,5) # insert operation
print(test_list)

# Output : [1, 5, 2, 3, 4]
```

remove(element)

Removes the first item with the specified value.

removeList.py

```
# Instagram : @dynamic.coding

test_list = [1, 2, 3, 4] # create list

test_list.remove(3) # remove operation
print(test_list)

# Output : [1, 2, 4]
```

pop(index)

Removes the element from the specified index in a list.

popList.py

```
# Instagram : @dynamic.coding

test_list = [1, 2, 3, 4] # create list

test_list.pop(3) # pop operation
print(test_list)

# Output : [1, 2, 3]
```

sort()

Sorts list in ascending order. pass parameter reverse=True for descending order.

sortList.py

```
# Instagram : @dynamic.coding

test_list = [5, 3, 2, 1, 4, 6]

test_list.sort() # sorting operation
print(test_list)

# Output : [1, 2, 3, 4, 5, 6]
```

count(element)

Removes the element from the specified index in a list.

countList.py

```
# Instagram : @dynamic.coding

test_list = [1, 2, 3, 4, 3, 3]

frequency = test_list.count(3) # count method
print(frequency)

# Output : 3
# element 3 appears 3 times in the input list
```