PYTHON LIST
CLASS XI
(MODULE - 3)
BY
Mrs. SUJATA PRADHAN,
PGT(SS),AECS,ANUPURAM

-

Python List Methods

Python has a lot of list methods that allow us to work with lists.
Methods or **functions**,used in lists are used to manipulate the data quickly.
syntax:
  list name.method name(element/index/list)

-

Python List Method : append()

The **append()** method adds an item to the end of the list.
The syntax of the append() method is:
  list.append(item)
append() method takes a single argument(item) which is added at the end of the list.The item can be numbers, strings, dictionaries, another list, and so on.

Example:

```
A = [10,20,30 ]
A.append(15)
print('Updated list: ', A)
```

Output

```
Updated list: [10,20,30,15]
```

- **Python List Method : extend()**

  The **extend()** method adds all the elements of an iterable (list, tuple, string etc.) to the end of the list.
  The syntax of the extend() method is:
  list1.extend(iterable)

  All the elements of iterable are added to the end of list1.
  extend() method takes an iterable such as list, tuple, string etc.
  Example:

  ```
  language = [‘Hindi', 'English']
  language1 = [‘Urdu', ‘Sanskrit’]
  language.extend(language1)
  print('Language List:', language)
  ```

  Output

  ```
  Language List: [‘Hindi', 'English’, ‘Urdu', ‘Sanskrit’]
  ```

- **Python List Method : insert()**

  The list **insert()** method inserts an element to the list at the specified index.
  The syntax of the insert() method is

list.insert(index, element)

Here, element is inserted to the list at the i^{th} index.

insert() method takes two parameters:

index - the index where the element needs to be inserted

element - this is the element to be inserted in the list

Example:

```
vowel = ['a', 'e', 'i', 'u']
vowel.insert(3, 'o')
print('Updated List:', vowel)
```

Output

```
Updated List: ['a', 'e', 'i', 'o', 'u']
```

Python List remove()

The **remove()** method removes the matching element (which is passed as an argument) from the list.
The syntax of the remove() method is:
list.remove(element)

remove() method takes a single element as an argument and removes it from the list.
list.remove(x)
Example:

```
animals = ['cat', 'dog', 'rabbit', ‘cow']
animals.remove('rabbit')
print('Updated animals list: ', animals)
```

Output

```
Updated animals list: ['cat', 'dog', ‘cow']
```

Python List count()

The **count()** method returns the number of times the specified element appears in the list.
The syntax of the count() method is:
list.count(element)

The **count()** method takes a single argument:

element - the element to be counted

Example:

```
vowels = ['a', 'e', 'i', 'o', 'i', 'u']
count = vowels.count('i')
print('The count of i is:', count)
count = vowels.count('p')
print('The count of p is:', count)
```

Output

```
The count of i is: 2
The count of p is: 0
```

-

Python List index()

The **index()** method returns the index of the element.
The syntax of the list index() method is:
list.index(element, start, end)

The list index() method can take a maximum of three arguments:
element - the element to be searched
start (optional) - start searching from this index
end (optional) - search the element up to this index

Example:

```
vowels = ['a', 'e', 'i', 'o', 'i', 'u']
index = vowels.index('e')
```

```
 print('The index of e:', index)
 index = vowels.index('i')
print('The index of i:', index)
```

Output

```
The index of e: 1
The index of i: 2
```

- **Python List pop()**

The **pop()** method removes the item at the given index from the list and returns the removed item.

The syntax of the pop() method is:

```
list.pop(index)
```

The pop() method takes a single argument (index).

The argument passed to the method is optional. If not passed, the default index **-1** is passed as an argument (index of the last item).

```
languages = ['Python', 'Java', 'C++', 'French', 'C']
return_value = languages.pop(3)
 print('Return Value:', return_value)
print('Updated List:', languages)
```

Output

```
Return Value: French Updated List: ['Python', 'Java', 'C++', 'C‘]
```

- ## Python List reverse()

The **reverse()** method reverses the elements of the list.
The syntax of the reverse() method is:

list.reverse()

reverse() method doesn't take any arguments.

Example:

```
systems = ['Windows', 'macOS', 'Linux']
print('Original List:', systems)
systems.reverse()
print('Updated List:', systems)
```

Output

```
Original List: ['Windows', 'macOS', 'Linux']
Updated List: ['Linux', 'macOS', 'Windows']
```

- ## Python List sort()

The **sort()** method sorts the elements of a given list in a specific ascending or descending order.
The syntax of the sort() method is:

list.sort(key=..., reverse=...)

sort() doesn't require any extra parameters.

Example 1: Sort a given list

```
vowels = ['e', 'a', 'u', 'o', 'i‘]
vowels.sort()
print('Sorted list:', vowels)
```

Output

```
Sorted list: ['a', 'e', 'i', 'o', 'u']
```

Example 2: Sort the list in Descending order

```
vowels = ['e', 'a', 'u', 'o', 'i']
vowels.sort(reverse=True)
print('Sorted list (in Descending):', vowels)
```

Output

```
Sorted list (in Descending): ['u', 'o', 'i', 'e', 'a']
```

- **Python List copy()**

The **copy()** method returns a copy of the list.A list can be copied using the **=** operator.

Example:

```
old_list = [1, 2, 3]
new_list = old_list
```

If we modify new_list, old_list is also modified.

```
old_list = [1, 2, 3]
new_list = old_list
new_list.append('a')
print('New List:', new_list)
print('Old List:', old_list)
```

Output

```
Old List: [1, 2, 3, 'a']
New List: [1, 2, 3, 'a']
```

-

Python List clear()

The **clear()** method removes all items from the list.
The syntax of clear() method is:
list.clear()
clear() method doesn't take any parameters.

Example:

```
list = [5, ('a'), ['1.1', ‘xyz']]
list.clear()
print('List:', list)
```

Output

```
List: []
```

Example:

```
list = [{1, 2}, ('a'), ['1.1', '2.2']] # clearing the list
del list[:] print('List:', list)
```

Output

```
List: []
```

- **remove(), pop() and clear() methods**

remove(item): Removes specified item from list.

pop(index): Removes the element from the given index.

pop(): Removes the last element.

clear(): Removes all the elements from the list.

Example:

```
X = ['A', 'F', 'B', 'Z', 'O', 'L']
X.remove('B')
print(X)
X.pop(1)
print(X)
X.clear()
print(X)
```

Output:

```
['A', 'F', 'Z', 'O', 'L']
['A', 'Z', 'O', 'L']
 []
```

- SUMMARY

Different List methods with syntaxes and examples

  - **append()**
  - **extend()**

- **insert()**
- **index()**
- **sort()**
- **count()**
- **reverse()**
- **pop()**
- **remove()**
- **clear()**

Thank You