**List methods**

| **Methods** | **Description** |
| --- | --- |
| append() | Adds an element at the end of the list |
| count() | Returns the number of elements with the specified value |
| extend() | Add the elements of a list (or any iterable), to the end of the current list |
| index() | Returns the index of the first element with the specified value |
| insert() | Adds an element at the specified position |
| pop() | Removes the element at the specified position |
| remove() | Removes the first item with the specified value |
| reverse() | Reverses the order of the list |
| sort() | Sorts the list |

Example 8: All examples are corresponding to the list methods stated above.

>>> list1 = ['Hello', 'Python', 2018, 814]

>>> list1.append(2049) #append() takes exactly one argument

>>> print(list1)

['Hello', 'Python', 2018, 814, 2049]

>>> list2 = [23, 2018, 814, 2049,2018]

>>> list2.count(2018) #count numbers of 2018

2

>>> list1.extend(list2) #add list2 into list1

>>> print("Extended List : ", list1)

Extended List : ['Hello', 'Python', 2018, 814, 23, 2018, 814, 2049, 2018]

>>> print("Index for python : ", list1.index('Python'))

1 #index value 'Python'

>>> print("Index for 2018 : ", list1.index(2018)) #find the first position of the indexed value

2

>>> list = ['Hello', 'Python', 2018, 814, 23, 2018, 814, 2049, 2018]

>>> list.insert(3, 2009) #list.insert(index, object),insert value 2009 in the index3

>>> print("New List : ", list)

New List : ['Hello', 'Python', 2018, 2009, 814, 23, 2018, 814, 2049, 2018]

>>> list = ['Hello', 'Python', 2018, 2009, 814, 23, 2018, 814, 2049, 2018]

>>> list.pop(2) #delete index2 value and return this value

>>> print('List now : ',list)

List now : ['Hello', 'Python', 2009, 814, 23, 2018, 814, 2049, 2018]

>>> list.remove('Hello') #remove certain value

>>> print('List now : ',list)

List now : ['Python', 2009, 814, 23, 2018, 814, 2049, 2018]

>>> list = ['Python', 2009, 814, 23, 2018, 814, 2049, 2018]

>>> list.reverse() #reverse list

>>> print('reverse list : ',list)

reverse list : [2018, 2049, 814, 2018, 23, 814, 2009, 'Python']

>>> vowels = ['e', 'a', 'u', 'o', 'i']

>>> vowels.sort() #reverse = True(descending), reverse = False(ascending, if no parameters in(), they will return default value, ascending)

>>> vowels

['a', 'e', 'i', 'o', 'u']

>>> vowels = ['e', 'a', 'u', 'o', 'i']

>>> print('vowels ascending : ',vowels) # sort by ascending

vowels ascending : ['a', 'e', 'i', 'o', 'u']

>>> vowels.sort(reverse = True)

>>> print('vowels descending : ',vowels) # sort by descending

vowels descending : ['u', 'o', 'i', 'e', 'a']

**Dict methods**

| **Methods** | **Description** |
| --- | --- |
| fromkeys() | creates dictionary from given sequence |
| get() | Returns value of the key, default=None |
| items() | Returns view of dictionary's (key, value) pair |
| keys() | Returns view object of all keys |
| contains(key) | Return bool value by checking whether the key is in dict |
| pop() | Returns & removes element having given key |
| values() | Returns view of all values in dictionary |
| update() | Updates the Dictionary |

Example 12: All examples are corresponding to the list methods stated above.

>>> seq = ['Chico', 'Ivy', 'Ri']

>>> dict = dict.fromkeys(seq) #get/create keys from the list

>>> print("New\_dict : %s" % str(dict))

New\_dict : {'Chico': None, 'Ivy': None, 'Ri': None}

>>> dict = dict.fromkeys(seq, 'A+') #give all keys value A+

>>> print("New\_dict : %s" % str(dict))

New\_dict : {'Chico': 'A+', 'Ivy': 'A+', 'Ri': 'A+'}

>>> dict = {'Name':'Chico','Gender':'Male','Age':'23'}

>>> print("Age : %s" % dict.get('Age')) #get key value

Age : 23

>>> print("Gender : %s" % dict.get('Gender'))

Gender : Male #if you get a wrong key, it will return None

>>> dict = {'Name':'Chico','Gender':'Male','Age':'23'}

>>> print("dict\_values : %s" % dict.items()) #view dict's items

dict\_values : dict\_items([('Name', 'Chico'), ('Gender', 'Male'), ('Age', '23')]) #return a tuple

>>> dict = {'Name':'Chico','Gender':'Male','Age':'23'}

>>> print("dict\_keys : %s" % dict.keys()) #view all keys

dict\_keys : dict\_keys(['Name', 'Gender', 'Age'])

>>> dict = {'Name':'Chico','Gender':'Male','Age':'23'}

>>> print("has\_key : %s" % dict.\_\_contains\_\_('Age')) #two '\_', check out keys

has\_key : True

>>> print("has\_key : %s" % dict.\_\_contains\_\_('School'))

has\_key : False

>>> dict = {'Name':'Chico','Gender':'Male','Age':'23'}

>>> pop\_value = dict.pop('Gender') #drop out key

>>> print(pop\_value)

Male

>>> print(dict)

{'Name': 'Chico', 'Age': '23'}

>>> dict = {'Name': 'Chico', 'Age': '23'}

>>> print("Value : %s" % dict.values()) #get all values

Value : dict\_values(['Chico', '23'])

>>> dict1 = {'Name': 'Chico', 'Age': '23'}

>>> dict2 = {'Gender':'Male'}

>>> dict1.update(dict2) #update dict2 in dict1

>>> print('new\_dict : %s' % dict1)

new\_dict : {'Name': 'Chico', 'Age': '23', 'Gender': 'Male'}