code: CSA0810

Name: python programming

Name: G. Jeevan

Reg. No: 192211714

ASSIGNMENT-I

List in python:

A list is a structure in python used to arrange data in ordered sequence of elements.

→ each element or value that is inside of a list is called an item.

→ list is a collection Data type. It allows multiple values to be stored within the same field.

→ Suppose we need to record the ages of 5 students. Instead of creating 5 separate variables, we can simply create a list.

Create a list:

We can create a list by placing elements inside [], separated by commas.

Eg:

```
ages = [19, 26, 23]
print(ages)
```

output: [19, 26, 23]

List items:

List items are ordered, changable & allow duplicate values.

→ List items are indexed, the first element has index [0].

Access list items:

List items are indexed and you can access them by referring to Index Numbers.

→ We can access the element by giving the index value.

Eg:

```
Names = ["Jack", "steve", "Mike"]
print (Names[1])
```

Output: steve.

change item:

To change value of specific item, refer to index number.

Eg:

```
fruits = ["apple", "banana", "orange"]
fruits[1] = "blackcurrent"
```

Output:

apple, blackcurrent, Orange.

Add list:

To add an item to end of the item, use the "append list method.

Eg:

```
Names = ["Mike", "Jack", "TIN"]
Names.append("steve")
```

Output:

aMike, Jack, TIN, steve.

Remove list:

It is used to remove an specified item.

→ It can remove the data or delete the data from the list.

Eg:

```
Names = ["Mike", "Jack", "Tin"]
Names.remove("Jack")
print(Names)
```

→ If there are more items with the specified value, the remove() method removes the first occurence.

copy list:

we cannot copy a list simply by typing list2 = list1, because: list 2 will only be a reference to list 1 and changes made in list 1 will automatically also be made in list 2.

Eg:

```
fruits = ["apple", "banana", "cherry"]
mylist = fruits.copy()
print(mylist)
```

output:

apple, banana, cherry.

Another way to make a copy is to use to built method list().

```
Mylist = list(fruits).
```

List Methods:

append():

Adds an element at the end of the list.

clear():

Removes all elements from list.

copy():

Return a copy of the list.

count():

Return number of elements with specified value.

extend():

Add elements to a list to end of concurrent list.

Index():

Return index of first element with specified value.

Insert():

Add an element at specified position.

pop():

Removes the element with specified position.

Remove():

Remove item with specified value.

Reverse():

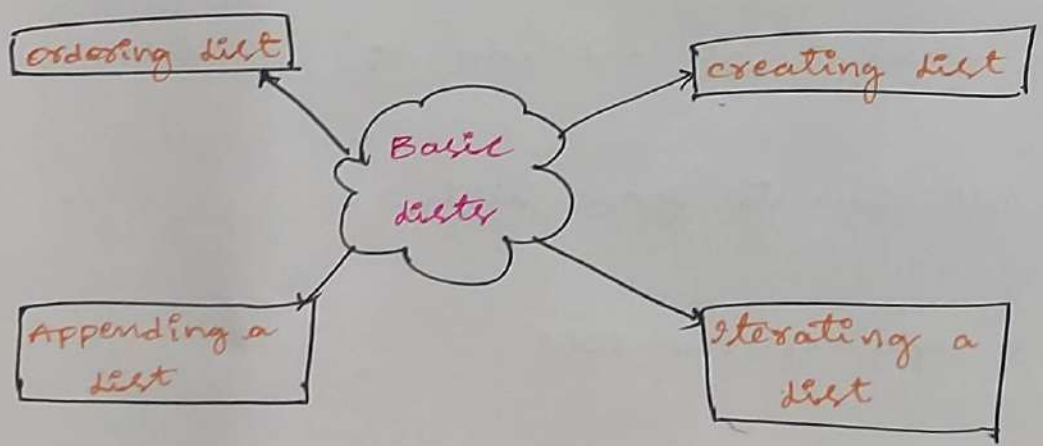
Reverse the order of list.

Sort():

Sort the list.

there are four Basic lists in python:

List data type configure with all the values that fields store.



Ordering a list:

When we say that list are ordered, it mean that the item have defined order, and the order will not change.

→ If you add new item to list, then new item will be placed at end of the list.

Eg:

```
a = [1, 2, "RAM", "Rahul", 5, 6]
b = [1, 2, 5, "RAM", 5, 6]
a == b
```

Output: false.

The indistinguishable components we remembered for the two record, however, the subsequent rundown changed the file position to fifth component.

→ The sorted() function returns a sorted list of the specified iterable object.

→ We can print ordered list by using enumeration map() function.

Eg:

```
Mylist = [3, 1, 2, 5, 4]
Sortedlist = sorted(mylist)
print(sortedlist)
```

output: 1, 2, 3, 4, 5.

→ We can order a list in python using the sorted function.

→ It can contains both integers & strings in the ordered list.

Creating a list:

TO create a list in python, we use square brackets ([]).

→ Unlike sets, a list doesn't need a build in function for its creation of a list.

→ The list can accept any data type which means you can have a list of integers & strings.

Eg:

```
list = [1, 9, 7, 8]
print(list)
```

Here, we created a list of 4 integer items.

Output:

1, 9, 7, 8.

→ It can store elements of different data types. (integer, float, String, etc).

→ store duplicate element.

```
#list with different element
list1 = [1, "Hello", 3.4]
#list with duplicate element
list2 = [1, "Hello", 3.4, "Hello"]
# empty list
list2 = [].
```

→ We can also create a list using list construction. list().

→ In python, lists are address (or) ordered & each item in a list associated with a number. The number is known as list index.

Appending a list:

Adding items to a list with python's .append() python lists reserve extra space for new items at the end of the list.

→ A call to .append() will place new items in the available space.

→ Lists are sequences and can hold different data types.

Eg:

```
currencies = ["Dollar", "Euro", "pound"]
currencies.append("yen")
print(currencies)
```

Here, operand is used to add the element 'yen' in the currencies.

Output:

Dollar, Euro, pound, yen.

→ the syntax is append() or listappend().

→ the method takes a single argument & any item can be added at end of the list.

Eg:

```
animals = ["cat", "dog", "pig"]
animals.append("horse")
print(animals)
```

output:

cat, dog, pig, horse.

→ append() is a built-in python function that adds a single element at the end of the existing list.

→ Every time append() is called an existing list.

Iterating a list:

the list is equivalent to arrays in other languages, with extra benefit of being dynamic in size.

→ In python, the list is a type of container in Data structures, which is used to store multiple data at same time. Unlike sets, lists in python are ordered and have definite count.

Eg:

```
languages = ["telugu","Tamil","Hindi"]
for language in languages:
print(language)
```

We can use the keyword to check if an item exists in the list.

```
languages = ["telugu","Tamil","Hindi"]
print ("english" in languages). # false.
print ("telugu" in languages) # true.
```

Output:

telugu, Tamil, Hindi.

→ the classic for-loop is actually the fastest way to iterate a list in python.

→ the method using the enumerate function is the fastest between the ones that are offering all items & indexes.

→ Iterating over a list is faster than iterating over a set.

→ loop & range() function together used for the iterating the list.