Lecture 10
Lists

Objectives

- After completing the lesson, the student will be able to:
  - Understand the basic concept of list.
  - Create a list in python.
  - Print the list items and print list items by using index.
  - Concatenate, repeat and find the length of a list.
  - Iterate a list and slice a list.
  - Add items to a list using different methods.
  - Sort lists in ascending order and descending order.
  - Delete items from a list using different methods.
  - Change items of a list.
  - Search and reverse a list.

Introduction to List

- A list is a sequence of items which is referred by a single name.
- Lists are a series of objects written inside square brackets, separated by commas.
- Lists can be used for numbers and strings.
- Lists grow and shrink automatically as they are used.
- The individual items can be selected by indexing.
- The index of the list starts from zero.

- Example:
  - If the size of the list is 5, the first index of the list is 0 and the last index of the list is 4.

Creating a List

- General Syntax:
  - Name = [value1, value2, value3…]
  - Name is the name of the list.
  - Values are the individual items in the list separated by comma.
  - The items in a list can be of any data type.

- Example:
  - List = [2, 4, 6, 8, 10]

Print the List Item

- Example:
  - FruitList = ["Apple", "Banana", "Mango", "Papaya"]

- To print the items of the list:
  - Use print function followed by the name of the list.
  - Or simply type the name of the list and press enter key.

- Example:
  - print(FruitList)
  - #This statement will print the items of the above list.

Print the List Item

- The individual items of the list can be printed by using the index of the item.

- Example:
  - FruitList = ["Apple", "Banana", "Mango", "Papaya"]

- print(FruitList[0])
  - This statement prints the first item of the above list.

- print(FruitList[3])
  - This statement prints the last item of the above list.

Concatenate Lists

- The addition operator (+) can be used to concatenate two lists.
- Example:
  - L1 = [2,4,6,8]
  - L2 = [10,12,14]
  - L1 + L2 #this statement creates a new list with all the items of L1 and L2.

- It is also possible to add lists with different data types.
- Example:
  - L5 = ['A', 'B','C','D']
  - L6 = [11,12,13,14,15]
  - L5 + L6   #produces : ['A', 'B', 'C', 'D', 11, 12, 13, 14, 15]

Repeat list Items, Length of the List

- The multiplication operator (*) can be used to repeat the items of the list.
- Example:
  - L5 = ['A', 'B','C','D']
  - L5 *2, produces ['A', 'B', 'C', 'D', 'A', 'B', 'C', 'D']

- The len() method can be used to find out the size of the list.
- Example:
  - len(L5)
  - The above statement produces 4 as the size of L5.

Iteration

- A list can be iterated to perform the same action to each object within the list.

- Example:
  - numbers = [99,88,77,66,55,44,33,22,11]
  - for myList in numbers:
    - print(myList)

  - The above for loop iterates till the list is exhausted and produce a list.

Iteration

- A list can be iterated to perform the same action to each object within the list.

- Example:
  - Fruits = ["Apple","Banana","Ciku","Durian","Eggfruit"]
  - for myFruit in Fruits:
    - print(myFruit)

  - The above for loop iterates till the list is exhausted and produce a list of fruits.

Slicing a List

- Slicing is a way of indexing a range of positions in a list.
- Both start and end should be int-valued expressions.
- A slice produces the sub list starting at the position given by start and running up to, but not including, position end.
- Example:
  - numbers = [100,-12,45,0,5, 100, 45]
  - numbers[0:5] # starts from the 1st element up to 6th element, but not including 6th element.
  - [100, -12, 45, 0, 5]

Slicing a List

- Example:
  - FruitList = ['durian', 'fig', 'Ciku', 'apple', 'Eggfruit']
  - FruitList[0:2] # starts from 1st element up to 3rd element, but not including 3rd element.
  - ['durian', 'fig']

- Example:
  - FruitList = ['durian', 'fig', 'Ciku', 'apple', 'Eggfruit']
  - FruitList[2:3] # starts from the 3rd element up to 4th element, but not including 4th element.
  - ['Ciku']

Slicing a List

- Example:
  - FruitList = ['durian', 'fig', 'Ciku', 'apple', 'Eggfruit']
  - FruitList[:3] # starts from 1st element if not specified.
  - ['durian', 'fig', 'Ciku']

  - FruitList[2:] # continues up to the last item of the list if not specified.
  - ['Ciku', 'apple', 'Eggfruit']

  - FruitList[:] # starts from 1st element up to the last element of the list if not specified.
  - ['durian', 'fig', 'Ciku', 'apple', 'Eggfruit']

Adding items to a list

- The append() function can be used to add items to a list.
- Example:
  - numbers = [99, 88, 77, 66, 55, 44, 33, 22, 11]
  - numbers.append(100)   #add 100 as the last element.
  - print(numbers)

- The new list is:
  - [99, 88, 77, 66, 55, 44, 33, 22, 11, 100]

Adding items to a list

- Example:
  - Fruits = ["Apple","Banana","Ciku","Durian"," Eggfruit"]
  - Fruits.append("Fig")   #add “Fig” as the last element.
  - print(Fruits)

- The new list is:
  - ['Apple', 'Banana', 'Ciku', 'Durian', ' Eggfruit‘, 'Fig']

Adding items via indexing

- insert() function can be used to add items to a specific position of a list.
- Index of the position can be used to add an item to a specific position of the list.

- Example:
  - Fruits = ["Banana","Ciku","Durian"," Eggfruit"]
  - Fruits.insert(0,"Apple")
  - print(Fruits)   # produces the following list in that order.
  - # ['Apple', 'Banana', 'Ciku', 'Durian', ' Eggfruit']
  - # The index of the item ‘Apple’ is 0.

Adding items via indexing

- Numbers also can be add to a specific position of the list by using the insert() function.

- Example:
  - numbers = [-12,5,45,0,100]
  - # Insert the number 500 as the 4th item of the list.
  - # Remember that index starts from 0
  - numbers.insert(3,500)
  - print(numbers)
  - #Output: [-12, 5, 45, 500, 0, 100]

Adding items using loops

- Lists are built up one piece at a time using the append function.
- Here is a piece of code that fills a list with positive numbers typed by the user:

- Example:

```
nums = []
n = int(input("Enter the number of elements: "))
while n > 0:
    x = input("Enter a number: ")
    nums.append(x)
    n = n-1
print(nums)
```

- nums is being used as an accumulator. It starts out empty, and a new value is added in each iteration of the loop.

Sorting a List

- The sort() function can be used to sort the items of a list.
- It change the list in-place and don’t create a brand new list object.
- Example:
  - FruitList = ['Durian', 'Fig', 'Ciku', 'Apple', 'Eggfruit']
  - print(FruitList)
  - #Output: ['Durian', 'Fig', 'Ciku', 'Apple', 'Eggfruit']

Sorting a List

- Example:
  - FruitList = ['Durian', 'Fig', 'Ciku', 'Apple', 'Eggfruit']
  - FruitList.sort()
  - print(FruitList)   # Prints the following sorted list.
  - #Output: ['Apple', 'Ciku', 'Durian', 'Eggfruit', 'Fig']
  - # The items in the list are sorted in ascending order.

- Example:
  - FruitList = ['durian', 'fig', 'Ciku', 'apple', 'Eggfruit']
  - # Sort the list in descending order:
  - FruitList.sort(reverse=True)
  - print(FruitList)
  - # Output: ['fig', 'durian', 'apple', 'Eggfruit', 'Ciku']

Sorting a List

- Example:
    - FruitList = ['durian', 'fig', 'Ciku', 'apple', 'Eggfruit']
    - FruitList.sort()
    - print(FruitList)
    - #Output: ['Ciku', 'Eggfruit', 'apple', 'durian', 'fig']
    - #Words with capital letters come first in a list with capital letters and simple letters.
- Example:
    - numbers = [100,-12,45,0,5]
    - numbers.sort()
    - print(numbers)   # prints the ascending order list.
    - #Output: [-12, 0, 5, 45, 100]

Deleting items from a list

- The pop() function can be used to delete items of the list.
- Example:
  - FruitList = ['durian', 'fig', 'Ciku', 'apple', 'Eggfruit']
  - FruitList.pop() # delete the last item of the list.
  - # 'Eggfruit'
  - print(FruitList)
  - # Output: ['durian', 'fig', 'Ciku', 'apple‘]

Deleting items from a list

- Example:
  - numbers = [100,-12,45,0,5]
  - numbers.pop()   # deletes the last item of the list and returns it.
  - print(numbers)
  - #Output: [100, -12, 45, 0]

- del statement can also be used to delete items from a list.
- Example:
  - numbers = [100,-12,45,0,5]
  - del numbers[3]   # deletes the 4th item of the list.
  - print(numbers)
  - #Output: [100, -12, 45, 5]

Deleting items from a list

- The remove function can be used to delete the first occurrence of the specified number.

- Example:
  - numbers = [100,-12,45,0,5, 100, 45]
  - numbers.remove(100) # deletes the 1st 100 in the list.
  - print(numbers)
  - #Output: [-12, 45, 0, 5, 100, 45]

Change items of a list

- Example:
  - numbers = [100,-12,45,0,5]
  - numbers[2] = 75 # index assignment, which assign 75 as the 3rd value of the list.
  - print(numbers)
  - #Output: [100, -12, 75, 0, 5]
  - numbers[0:2] = [99,87] #slice assignment: delete + insert.
  - #Replaces items indexed at 0 and 1.
  - print(numbers)
  - #Output: [99, 87, 75, 0, 5]

Searching a List

- Example:
  - numbers = [100,-12,45,0,5, 100, 45]
  - 5 in numbers # check to see if 5 is in the list and returns a Boolean result.
  - True

- The count() function can be used to count the number of occurrences of a specific item in the list.
- Example:
  - numbers = [100,-12,45,0,5, 100, 45]
  - numbers.count(100) # return the number of occurrence of 100 in the list as 2.

Reverse a List

- The reverse() function can be used to reverse a list.
- Example:
  - numbers = [100,-12,45,0,5, 100, 45]
  - numbers.reverse() # reverses the above list.
  - print(numbers)
  - #Output: [45, 100, 5, 0, 45, -12, 100]
- Example:
  - FruitList = ['durian', 'fig', 'Ciku', 'apple', 'Eggfruit']
  - FruitList.reverse() # reverses the above list.
  - print(FruitList)
  - #Output: ['Eggfruit', 'apple', 'Ciku', 'fig', 'durian']