Module 2:

Lists and Tuples

By Ninad Gaikwad

Reference - “Core Python Programming”
Dr. R. Nageshwara Rao
Dreamtech Press

Lists: Introduction

- List is a group elements of different types
- Lists are mutable

1. Creating a list using range
   - Eg lst = list(range(4,9,2))
     # will generate list containing [2, 4, 6, 8]
2. Printing / accessing list:
   - for i in list:
                print(i)
     # will print all numbers in list
3. Updating elements of list
   - lst[1:3] = 10, 11
     # will update list to [2, 10, 11, 8]
4. Concatenation of two lists
   - Eg a=[1, 2, 3]
     b=[4, 5]
     then a+b=[1, 2, 3, 4, 5]

List: Introduction

5 Repetition of lists
  - print(a*2)
    # will give [1, 2, 3, 1, 2, 3]

6 Membership in lists
  - print(2 in a) # will return true
  - print(5 not in a) # will return true

7 Aliasing and cloning list
  - Aliasing x=a # will refer to a with additional reference x
  - Cloning x=a[:] # will make a separate copy of a by the name x
  - x=a.copy() # will also clone

Methods / Functions to process lists

- Methods: list.sum(), list.index(x), list.append(x), list.insert(i, x), list.copy(), list.extend(list1), list.count(x), list.remove(x), list.pop(), list.sort(), list.reverse(), list.clear()

1. min / max functions
   - min(list), max(list)
2. Finding common elements in two lists:
   - Convert the two lists to sets
   - Perform intersection operation
   - Eg s1=set(list1)
     s2=set(list2)
     s3= s1.intersection(s2)
     reslist=list(s3)
3. Nested lists
   - Eg a=[80, 90]
     b=[10, 20, 30, a]
   - Nested lists can be used as matrices

Methods / Functions to process lists

4 List comprehensions

- Creating of a new list from an iterable object
- Syntax: [expression for item1 in iterable1 if statement1
  for item2 in iterable1 if statement2
  for item3 in iterable1 if statement3]
- Eg num1 = [1, 2, 3, 4, 5]
  num2 = [10, 11, 1, 2]
  num3=[i for i in num1 if i not in num2]

  OR

  dict1 = {value:key for key, value in dict.items()}

Experiment 3:

Q: Write a menu-driven program to demonstrate the use of list in python

a) **Put Even and Odd elements between 1 to 20 into Two Different Lists.**

b) **Merge and sort the two lists.**

c) **Update the first element with X value and delete the middle element of the list.**

d) **Find max and min elements from the list.**

e) **Add N names into the existing number list and check if word python is present in the list.**

Tuples: Introduction

- Group of immutable elements of different types

1. Creating tuple
    - tup1=(10, 20, 30)
2. Accessing tuple
    - print(tup1[1])
    - print(tup1[-2:])
3. Basic operations in tuple:
    - Finding length, concatenation, repetition, membership and iteration
4. Functions available for tuples
    - len(tpl), min(tpl), max(tpl), sorted(tpl)
5. Methods available for tuples
    - tpl.count(), tpl.index()

Tuples: Introduction

5 Nested tuples and their sorting

- ○ Eg emp = ((10, “vijay”, 9000.90), (20, “bijoy”, 5500.50), (30, “vanaja”, 9800.00), (40, “kapoor”, 5000.00))

  ```
  print(sorted(emp)) # sorts on default id
  print(sorted(emp, reverse= true)) # sort id in reverse
  print(sorted(emp, key = lambda x: x[1])) # sort on name
  print(sorted(emp, key = lambda x: x[2])) # sort on salary
  ‘’’Output
  ((10, “vijay”, 9000.90), (20, “bijoy”, 5500.50), (30, “vanaja”, 9800.00), (40, “kapoor”, 5000.00))
  ((40, “kapoor”, 5000.00), (30, “vanaja”, 9800.00), (20, “bijoy”, 5500.50), (10, “vijay”, 9000.90))
  [(20, 'bijoy', 5500.5), (40, 'kapoor', 5000.0), (30, 'vanaja', 9800.0), (10, 'vijay', 9000.9)]
  [(40, 'kapoor', 5000.0), (20, 'bijoy', 5500.5), (10, 'vijay', 9000.9), (30, 'vanaja', 9800.0)]
  ’’’
  ```

6 Inserting, Modifying and deleting elements in a tuple

- ○ Copy elements before the location in another tuple
- ○ Add / delete / modify element to new tuple
- ○ Copy the rest of elements to the old tuple by overriding it

Experiment 4:

Q: Write a menu-driven program to demonstrate the use of tuples in python

a) **Add and show N student roll number, name, and 3 subject marks in a list of tuples.**

b) **Display student roll number and marks whose name is Python**

c) **Demonstrate nested tuples and sort nested tuples by name**.