Lists: mutable(changeable) – uses square brackets []

Tuple: immutable (not changeable) – uses round brackets ()

Tup = (21,36,14,25)

Tup[1]

Tup[1] = 33 – this doesn’t allow assignment

Tup.count/index (we have only two methods) – **clarification of tup count**

Iteration in tuple is faster than list, hence it is used to enhance the speed of execution.

Set – simply a collection of unique elements. Which uses {} – different types

Set never follows a sequence – try giving duplicate numbers and print the output

Set doesn’t support indexing so if you try s[1] it will give the error.

Explore s.(options)

Dictionary – {}: key-value pairs ex: phonebook where we will search with name and phone number

Ex : dictionary = {1:’bunny’, 2:’rishi’, 5:’puja’}

Dictionary[5]

Dictionary.get(5)

Dictionary.get(9)

Print(dictionary.get(9)) – will result in none

Dictionary.get( 1, ‘not found’)

Dictionary.get(9, ‘not found’) – since 9 is not available with the dictionary it will display not found.

Now let’s create 2 lists and merge them into a dictionary

Keys = [‘uday’, ‘bunny’, ‘rishi’, ‘puja’]

Values = [‘python’, ‘java’, ‘c’, ‘cobol’]

Data = dict(zip(keys,values)) – this will merge the keys and values

Now if we want to add two values

Data[‘sai’] = ‘bash’

To delete values – del data [‘uday’]

We can nest the dictionary inside of another dictionary

Prog = {‘JS’:’Atom’, ‘CS’:’VS’, ‘python’:[‘pycharm’.’sublime’], ‘java’: {‘JSE’:’Netbeans’,’JEE’:’Eclipse’}}

Here we had added a list inside a dictionary and a dictionary inside a dictionary

now prog[‘JS’] – it should result Atom

prog[‘python’] – it should result in pycharm and sublime

prog[‘java’] – it should display the dictionary

prog[‘java’][‘JEE’] – it will result in eclipse