String

* Strings are immutable
* .format. This is used to format the string.Example: ‘Rithvik{}{}’.format(‘Ballal’,’Mitresh’), will print Rithvik Ballal Mitresh. ‘Rithvik{1}{0}’. format(‘Ballal’,’Mitresh’), will print Rithvik Mitresh Ballal.
* You can use assign the string inside of .format to a variable and then use those variable inside the {}. Example: ‘Rithvik{m}{b}’.format(m=’Mitresh’,b=’Ballal) will print Rithvik Mitresh Ballal.

List

* List can have any type of element, i.e it can contain one element of type int another element of type string and so on.
* List comprehension is where-in you created a list from a loop in one liner. Example my\_list=[x for x in range(0,10) if x%2 ==0].

Tuples

* Tuples are immutable meaning the elements of a tuple cannot be reassigned.
* While looping there is a concept called tuple unpacking, where-in you can access the required tuple element from a list of tuples. Example my\_list=[(1,2),(2,3)] for (a,b) in my\_list:print(b). This will print the second element of the tuple. For this to work, the size of tuple in the list must be the same.

Set

* Set are ordered sequence and contains only unique numbers and charaters.

I/O files

* Filename.seek(0) will reset the curse to the starting letter of the file.
* Filename.close() will close the file.
* with open(Filename) as variable. This don’t require Filename.close() since it is implicitly called.

Comparison Operator

* The logical operator in python are and not and or instead of &&,! and ||

Loops

* In for loop, if you don’t want to declare a loop variable, you can just use \_, like for \_ in my\_list: print(‘hi’)
* If you don’t want to fill the body of the loop but want to execute the code, then you can you use the pass keyword, which tells don’t do anything.
* Enumerate function is used to iterator the list. This will return a tuple of index and value.
* Shuffle from random package is used to shuffle the list.
* zip is used to combine the list into a tuple. Example my\_list1=[1,2,3], my\_list2=[‘Rithvik’,’Mitresh’,’Ballal’] will create (1,’Rithvik’),(2,’Mitresh’).

Functions

* The parameter of a function can be assigned a default value. Example: def my\_func(name=’Rithvik’), the name here has a default value as Rithvik, so if you call the function without pass the value to the name then ‘Rithvik’ is assigned to the name variable.
* \*args and \*\*kwargs parameter to a function is used if we don’t know the number of parameter required for a particular function. The output of args is a tuple of parameters pass will the function was called and output of kwargs is a dictionary.