1. Python supports negative indexing of an iterable object. We can access the elements of a object from the end by indexes starting from -1, -2, -3... There is no repercussion if we use either positive or negative indexing.

2. We can use \* operator for this.

3. We use list slicing operations to acheive this.

my\_list = [1,2,3,4,5,6,7,8,9,10,11,12]

new\_list = my\_list[1:len(my\_list):2]

print(new\_list)

[2, 4, 6, 8, 10, 12]

4. Indexing is the way of accessing a single element of the object by its index or position, where as, slicing enables us to access the sub parts of object. Slicing can be done using indexes.

my\_list = [1,2,3,4,5,6,7,8,9,10,11,12]

new\_list = my\_list[1:len(my\_list):2] # this is list sclicing.

print(new\_list)

print(new\_list[2]) # this is list indexing

5. The slicing operation doesn’t raise an error if one of the indices are out of range. This is in not in the case of indexing;if you index an element that is out of bounds, Python will throw an index out of bounds error. However, with slicing it simply returns an exisiting accessed elements. It stops accessing elements after the end of the object is reached.

6. The slicing operation doesn’t raise an error if one of the indices are out of range. This is in not in the case of indexing;if you index an element that is out of bounds, Python will throw an index out of bounds error. However, with slicing it simply returns an exisiting accessed elements. It stops accessing elements after the end of the object is reached.

7. A matrix is balanced if all cells in the matrix are balanced and a cell of the matrix is balanced if the number of cells in that matrix that are adjacent to that cell is strictly greater than the value written in this cell. Adjacent cell means cells in the top, down, left, and right cell of each cell if it exists.

import numpy as np

mat = np.matrix([[1, 2, 3],[4, 5, 6],[7, 8, 9]])

mat

# This is unbalanced matrix

matrix([[1, 2, 3],

[4, 5, 6],

[7, 8, 9]])

8. In Python 3, we can use numpy library to create large matrices instaed of list comprehension or a loop.