**Python -theory Question:**

1. **What is Python?**

Python is a high-level, interpreted and general-purpose language. Python supports objects, module, thread, exception handling and automatic memory managaement.

1. What is Interpreted Language?

An Interpreted language executes statement line by line.

1. What is List & Tuples? What are their key difference?

List and tuples are sequence data type that can store a collection of objects in python.

List are mutable while tuples are immutable objects.

1. What is Pass?

Pass keyword represent null operation in python.

1. What are modules and Packages in Python?

Modules are python file with .py extension and can have set of functions, classes or variables defined and implemented. They can be imported and initialized using import statement.

Packages allow for hierarchical structure of the module namespace using dot notation.

1. What are Global, Protected and Private attributes in Python.

Global – Variable is defined by using a key word “global”.

Protected – Variable is defined by single underscore eg:\_sara. They can still be accessed and modified outside the class.

Private – Attributes are defined by double underscore. Eg:\_\_sara. They cannot be accessed or modified from outside and will result in Attribute error.

1. What is the use of Self in Python?

Self is used to represent instance of a class. With this keyword, you can access attributes and methods of a class.

1. What is \_\_init\_\_?

\_\_init\_\_ is a constructor method in python and is automatically called to allocate memory when a new object/instance is created.

1. What is break, continue and Pass in Python?

Break – terminates the loop and the control flow to the statement after the body of the loop.

Continue – Terminate the current iteration of the statement. Skip rest of the code in current iteration and control flow to the next iteration.

Pass – It generally fill up empty block.

1. What is slicing in Python?

Slicing will take part of sequence objects. [start, stop, step]

1. What is difference between Array and List in Python?

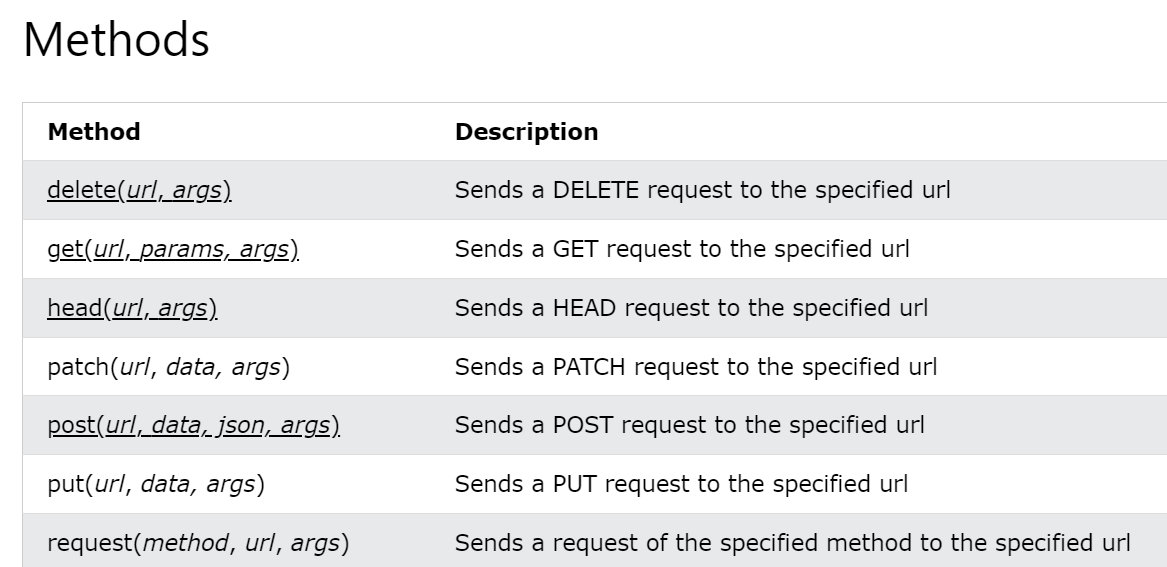
Array – contain elements of same data type.

List – Contain elements of different data type.

1. Queue - A queue is a linear data structure that stores a collection of elements, with operations to enqueue (add) elements at the back of the queue, and dequeue (remove) elements from the front of the queue.

Deque - A deque (double-ended queue) is a linear data structure that stores a collection of elements, with operations to add and remove elements from both ends of the deque.

1. Request module: The requests module allows you to send HTTP requests using Python. The HTTP request returns a [Response Object](https://www.w3schools.com/python/ref_requests_response.asp) with all the response data (content, encoding, status, etc).



1. Functools module: <@lru\_cache(maxsize=2)> lru\_cache means least recently used, it store max two elements or

Cached module: <@cached(cache=TTLCache(maxsize=2, ttl=900))>

1. API – Application Programming Interface

Below are the error codes,

200 – request succeeded. No error.

301 – The server is re-directing you to different endpoint

400 – Bad request. This happen if u don’t send right data/parameter along with other data

401 - The server think you are not authenticated

403 - The resource you are accessing is forbidden

404 – The folder/file is not available at the specified path

503 – The server is not ready to handle the request.

**Binary Search:**

**Binary Search** is defined as a [searching algorithm](https://www.geeksforgeeks.org/searching-algorithms/) used in a sorted array by **repeatedly dividing the search interval in half**. The idea of binary search is to use the information that the array is sorted and reduce the time complexity to O(log N).

In the below list if we want to find 56 we need to go through all the element in list that will make the complexity O(N), if data grow code will be slower.

**[2, 5, 0 ,89, 93, 56, 88, 23, 08, 44] -> O(N)**

**Binary search:**

Sort -> **[0, 2, 5 ,08, 23, 44, 56, 88, 89, 93]**

Select middle element 23 and check 56 > 23, if yes then gray out the lower values

[**44, 56, 88, 89, 93**]

Again elect middle element 88 and search 56 > 88, if no then gray out higher elements

**[44, 56]**

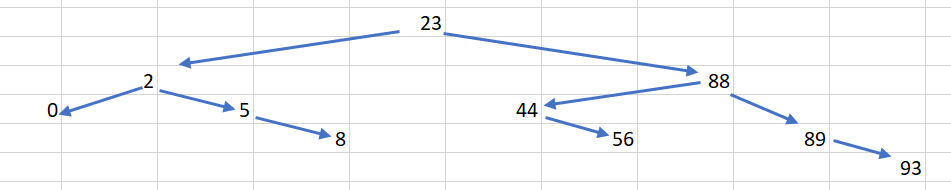
Again select middle element 44 and search 56 > 44, if yes then gray out lower elements

**[56]**

**Binary Tree:**

A Binary tree is represented by a pointer to the topmost node (commonly known as the “root”) of the tree. If the tree is empty, then the value of the root is NULL. Each node of a Binary Tree contains the following parts:

1. Data
2. Pointer to left child
3. Pointer to right child



Unlike linear data structures (Array, Linked List, Queues, Stacks, etc) which have only one logical way to traverse them, trees can be traversed in different ways.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| |  | | --- | |  | | 14 | |  | | --- | |  | |  |
|  |  |  |  |
| 9 |  | 22 | |  | | --- | |  | |
|  | |  | | --- | |  | |  |  |
|  | 13 |  | 34 |

1. Inorder Traversal -> ordered list or printing a tree

Go Left , then check , go right (9, 13, 14, 22,34)

1. Preorder Traversal -> saving a tree so it can be re-constructed later

Check, go left, go right (14, 9, 13, 22, 34)

1. Postorder Traversal -> useful for deleting the tree

Go left, go right, then check (13, 9, 34, 22, 14)