Memory Management in Python

Memory allocation can be defined as allocating a block of space in the computer memory to a program. In python memory allocation and deallocation method is automatic as the python developers created a garbage collector for python that the user does not have to do manual garbage collection

Garbage collection: -

Garbage collection is a process in which the interpreter frees up the memory when not in use to make it available for other objects.
Assume a case where no reference is pointing to an object in memory that is it is not in use so, the virtual machine has a garbage collector that automatically deletes that object from the heap memory

Let’s suppose there are two or more variables that have the same value, so, what Python virtual machine does is, rather than creating another object of the same value in the private heap, it actually makes the second variable point to that originally existing value in the private heap. Therefore, in the case of classes, having a number of references may occupy a large amount of space in the memory, in such a case referencing counting is highly beneficial to preserve the memory to be available for other objects.

There are two parts of memory:

1. Stack memory
2. Heap memory

The methods/method calls and the references are stored in stack memory and all the values objects are stored in a private heap.

Work of Stack Memory:

The allocation happens on contiguous blocks of memory. We call it stack memory allocation because the allocation happens in the function call stack. The size of memory to be allocated is known to the compiler and whenever a function is called, its variables get memory allocated on the stack.

It is the memory that is only needed inside a particular function or method call. When a function is called, it is added onto the program's call stack. Any local memory assignments such as variable initializations inside the particular functions are stored temporarily on the function call stack, where it is deleted once the function returns, and the call stack moves on to the next task. This allocation onto a contiguous block of memory is handled by the compiler using predefined routines, and developers do not need to worry about it.

Work of Heap Memory:

The memory is allocated during the execution of instructions written by programmers. Note that the name heap has nothing to do with the heap data structure. It is called heap because it is a pile of memory space available to programmers to allocated and de-allocate. The variables are needed outside of method or function calls or are shared within multiple functions globally are stored in Heap memory.