There are two ways via which memories can be allocated for storing data. The two ways are:

1. **Compile time allocation or static allocation of memory:** where the memory for named variables is allocated by the compiler. Exact size and storage must be known at compile time and for array declaration, the size has to be constant.
2. **Runtime allocation or dynamic allocation of memory:** where the memory is allocated at runtime and the allocation of memory space is done dynamically within the program run and the memory segment is known as a heap or the free store. In this case, the exact space or number of the item does not have to be known by the compiler in advance. **Pointers** **play** a major role in this case.

Memory in your C++ program is divided into two parts:

* **stack**: Memory from the stack is used by all the members which are declared inside functions. **Note that main is also a function..**
* **heap:** It is the unused memory of the program and can be used to dynamically allocate the memory at runtime.

We can initialize a variable while dynamical allocation in the following two ways.

**int \*ptr = new int (4);**

**int \*ptr = new int {4};//also**

Example:Dynamic Allocation For Array

har\* val  = NULL;       // Pointer initialized with NULL value

val = new char[40];     // Request memory for the variable

**We can also dynamically allocate objects. We again use pointers while dynamically allocating memory to objects.**

*class A*

*{*

*public:*

*A() {*

*cout << "Constructor" << endl;*

*}*

*~A() {*

*cout << "Destructor" << endl;*

*}*

*};*

***int*** *main()*

*{*

*A\* a = new A[4];*

*delete [] a; // Delete array*

***return*** *0;*

*}*

The Constructor will be called four times since we are allocating memory to four objects of the class 'A'. The Destructor will also be called four times during each of these objects.

**Where object stored in java and C++**

In Java, all **objects** are dynamically allocated on Heap. This is different from **C++** where **objects** can be allocated memory either on Stack or on Heap. In C++, when we allocate the **object** using new(), the **object** is allocated on Heap, otherwise on Stack if not global or static.