Memory Management:

# Stack and Heap:

Two areas of memory to take care about:

Heap: where the objects live.

Stack: where the local variables and methods invocations live.

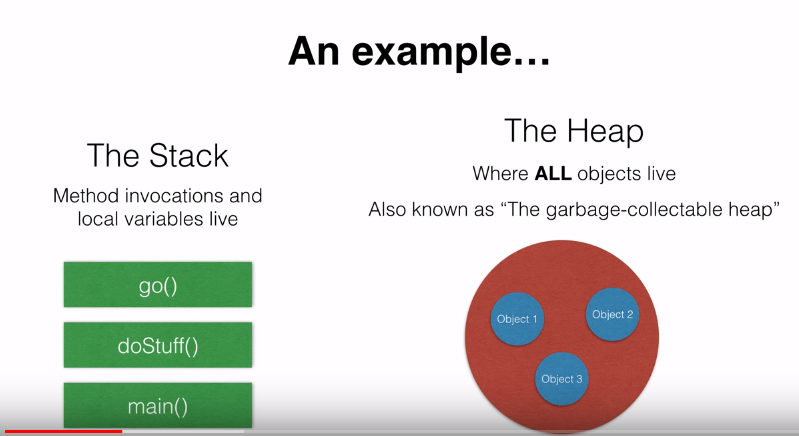
**What happens when a JVM Starts up?**

When JVM Starts up it gets chunk of memory from underlying operating system and it uses it to run java program.

**How much memory does it get?**

Depends upon the JVM Version (and on which platform).

All objects are garbage collectable on the heap.



**Local variables that store objects?**

Firstly, let’s remember that variables are nothing but reserved memory locations to store values.

When we create a variable, we are reserving space in memory. Based on the data type of the variable the operating system allocates memory and decides what can be stored in reserved memory.

Example: integers, characters and decimals can store in these variables.

There are two data types available in java: Reference/Object data types.

**Referencing an Object:**

A non-primitive variable holds a reference to an Object, not the object itself. Remember that objects live in the heap.

The reference to the object is stored in the stack.

Go() (d)

