Q37. What is JVM and explain me the Java memory allocation \*

JVM (Java Virtual Machine) is an abstract machine that provides runtime environment in which Java bytecode can be executed.

JVMs are available for many hardware and software platforms (i.e. JVM is platform dependent).

Java memory allocation:

1. Classloader: It is a subsystem in JVM which is used to load class files. Whenever we run the java program, it is loaded first by the classloader.
2. Class(Method) Area: Area stores pre-class structures such as the runtime constant pool, field and

Method data, code for methods.

1. Heap: It is a runtime data area in which objects are allocated.
2. Stack: It stores frames. It holds local variables and partial results. A new frame is created each time a method is invoked. A frame is destroyed when its method invocation completes.
3. PC(Program Counter)register: It contains the address of the JVM instruction currently being executed.
4. Native Method stack: It contains all the native methods used in the applications.
5. Execution Engine: It contains Virtual processor, Interpreter and JIT compiler.
6. Java Native Interface: JNI is a framework which provides an interface to communicate with another application written in another language like c, c++ etc. JNI to send output to console to interact with OS libraries.