Q.3) What is role of JVM in java? & How does JVM executes Java code?

→ JVM is crucial in Java as it serves as runtime enviroment for executing Java code. Its primary role includes interpreting or compiling bytecode, managing memory, handling exceptions, ensuring platform independence, & providing security features such as bytecode verification & sandboxing. Essentially JVM allows Java progs. to run on any device or operating system that has a compitable JVM implementation.

JVM executes Java code in steps→

i) Source code is compiled by Java compiler into byte code.
ii) JVM loads bytecode classes dynamically as needed during execution.
iii) JVM verifies bytecode to ensure it is Java language specification & does not violate security constraints.
iv) JVM interprets bytecode instructions or use JIT compiler to translate bytecode into machine code.
v) JVM manages memory allocation, deallocation, & garbage collection to ensure efficient memory usage.

vi) Exception handing. JVM handles exceptions & runtime errors gracefully, allowing Java prog. to recover from unexpected situations.

vii) JVM implementations does optimizations like, JIT compilation to improve performance.