Java Interview Questions

① Explain the components of the JDK.

→ ① appletviewer → This tool is used to run and debug java applets without a web browser.

② apt → It is an annotation-processing tool

③ java → The loader for Java applications. This tool is an interpreter and can interpret the class files generated by the javac compiler.

④ javac → It specifies the java compiler, which converts source code into java bytecode.

⑤ jar → The specifies the achiver, which packages related class libraries into a single jar file. This tool also helps manage JAR files.

⑥ javadoc → The documentation generator, which ~~package related class libr~~ automatically generates documentation from source code comments.

② Differentiate bet^n JDK, JVM & JRE

→ JVM → JVM is an abstract machine. It is called a virtual machine because it doesn't physically exists. It is a specification that provides a runtime environment in which Java bytecode can be executed.

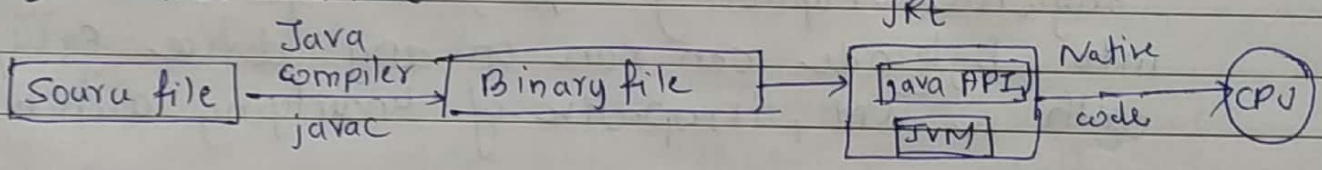
JRE → JRE is a set of software tools which are used for developing java applications. It is used to provide the runtime enviroment. It is the implementation of JVM. It physically exists. It contains a set of libraries + other files that JVM uses at runtime.

JDK: The JDK is a Software development environment which is used to develop java applications and applets. It physically exists. It contains JRE + development tools.

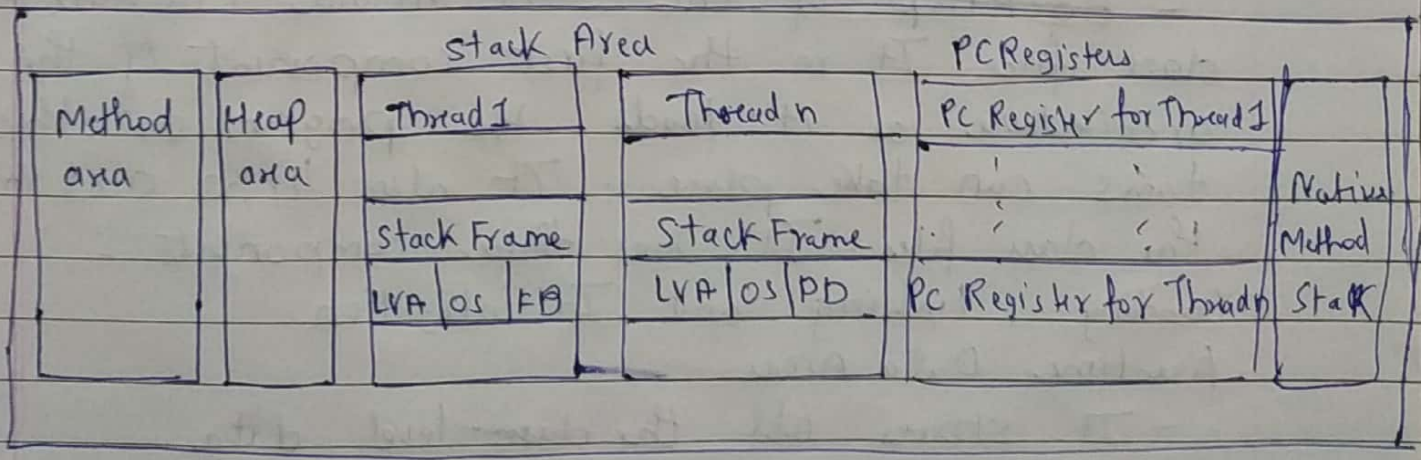
3) What is the role of the JVM in Java? How does the JVM execute Java code?

→ JVM enable Java programs to be platform-independent, meaning that java code can run on any device or operating system that has JVM installed. The JVM acts as an intermediary between java bytecode & the machine code that is a computer's hardware understands

Execution Flow:



4) Explain the memory management system of the JVM.



Shrikrupe

5) What are the JIT compiler and its role in the JVM? what is the bytecode and why is it important for java?

→ The JIT compiler is a component of the Java Virtual Machine (JVM) that improves the performance of Java applications by compiling bytecode into native machine code at runtime.

Role in the JVM are Performance Improvement, Dynamic Compilation, Adaptive Optimization and garbage collection Integration.

Bytecode is an intermediate, platform independent code generated by the java compiler from java source code. Bytecode is the key to java's 'write once, run anywhere' capability. A java program compiled into bytecode can run on any device equipped with a JVM, irrespective of underlying architecture.

6) Describe the architecture of the JVM

→ Class Loader System : Class loader is a subsystem - ~~Bootstrap~~ of the Java Virtual Machine loads class files. It is the first component of the architecture, as it loads the program so other tasks can take place. It also links and initialize the class files. It has three components - Loading, Linking and Initialization.

Runtime Data Areas

- It stores all the class-level data
- Static Variables, static Blocks, static methods, Instance methods are stored in this area.
- Every JVM has only one method area.

Execution Engine : The Execution engine executes the bytecode. It reads and executes and has different components.

⑦ How does Java achieve platform independence through the JVM.

→ ① Compilation to Bytecode

- Source code to bytecode : When a java program is compiled, it is not converted into machine code specific to any specific ~~h~~ to any particular hardware or operating system. Instead, the Java compiler (javac) translates java source code into an intermediate ~~from~~ form called bytecode.

② Java Bytecode

- Platform - Independent Instructions : Bytecode is a set of instructions that is platform-independent. It is not tied to any particular machine or operating system.

③ Java Virtual Machine (JVM)