Interview Questions :

1: What is Java?

- Java is a high-level, object-oriented programming language developed by Sun Microsystems (now owned by Oracle Corporation).
- It is designed to be platform-independent and write once, run anywhere (WORA), meaning that Java programs can run on any device or operating system that has a Java Virtual Machine (JVM) installed.

2: What are the main features of Java?

a. Object-oriented:
- Java supports object-oriented programming principles such as encapsulation, inheritance, and polymorphism.

b. Platform-independent:
- Java programs can run on any platform with a JVM.

c. Robust and secure:
- Java provides built-in memory management, exception handling, and security features.

d. Multi-threaded:
- Java supports concurrent programming with its built-in threading capabilities.

e. Rich API:
- Java has a vast collection of standard class libraries and APIs for various purposes.

f. Garbage collection:
- Java handles memory management automatically through garbage collection.

g. Simple and readable syntax:
- Java has a clean and easy-to-read syntax, making it user-friendly.

3: What is the difference between JDK, JRE, and JVM?

a. JDK (Java Development Kit):
- It is a software development kit that includes tools, compilers, and libraries necessary for developing and running Java applications.

b. JRE (Java Runtime Environment):
- It is the runtime environment that provides the necessary libraries and resources to run Java applications.

c. JVM (Java Virtual Machine):
- It is the virtual machine that executes Java bytecode. It is responsible for interpreting and executing Java programs.

4: What is the difference between an object and a class in Java?

- In Java, a class is a blueprint or template that defines the properties and behaviors of objects.
- An object, on the other hand, is an instance of a class that represents a specific entity with its own state and behavior.

5: What is the difference between the == operator and the .equals() method in Java?

- The == operator in Java compares the references of two objects to check if they refer to the same memory location.
- On the other hand, the .equals() method is used to compare the content or values of two objects to check if they are logically equal.

6: What are the access modifiers in Java?

- Java provides four access modifiers:

a. public: Accessible from anywhere.
b. protected: Accessible within the same package and subclasses.
c. private: Accessible only within the same class.
d. default (no explicit modifier): Accessible within the same package.

7: What is the difference between abstraction and encapsulation in OOP?

- Abstraction and encapsulation are two important concepts in OOP, but they serve different purposes:
- Abstraction is the process of representing complex real-world entities as simplified models in software.

- It focuses on showing only the essential features and hiding unnecessary details.
- Abstraction is achieved through abstract classes and interfaces in Java, which provide a blueprint for derived classes to implement.

- Encapsulation, on the other hand, is the mechanism of bundling data and methods together as a single unit, known as a class.
- It provides data hiding and protects data from direct access by external entities.
- Encapsulation helps maintain data integrity and promotes modular design.

- In summary, abstraction is about simplifying complexity by defining abstract classes and interfaces, while encapsulation is about bundling data and methods together within a class and controlling access to them.

8: What is Apache Tomcat?

- Apache Tomcat, commonly referred to as Tomcat, is an open-source web server and servlet container developed by the Apache Software Foundation.
- It is primarily used for deploying and running Java-based web applications.
- Tomcat provides a runtime environment for Java Servlets, JavaServer Pages (JSP), and other Java-based web technologies.

9: How does Tomcat handle requests and responses in a Java application?

- Tomcat follows the Java Servlet Specification and uses a request-response model to handle incoming HTTP requests and generate responses.
- When a client sends an HTTP request to a Tomcat server, Tomcat processes the request by invoking the appropriate servlet or JSP.
- The servlet/JSP then generates the response, which is sent back to the client.
- Tomcat maintains a thread pool to handle multiple requests concurrently.
- Each request is processed in a separate thread, allowing for efficient handling of multiple client requests.
- Tomcat manages the lifecycle of servlets and JSPs, ensuring they are initialized, destroyed, and reused as necessary.

10: What is the difference between a Java console-based application and a web-based application?

- A Java console-based application and a web-based application are two different types of applications with distinct characteristics:

a. Java Console-based Application:

- A Java console-based application runs in a command-line environment and interacts with the user through a console or terminal window.
- It typically uses text-based input and output, where the user provides input through keyboard input and receives output through the console window.
- Console-based applications are suitable for tasks that require simple input and output operations, batch processing, or command-line utilities.

b. Web-based Application:

- A web-based application, on the other hand, is accessed and used through a web browser over the internet or a local network.
- It consists of web pages that are displayed in the browser and typically involves a client-server architecture.
- Web-based applications use HTML, CSS, and JavaScript to create interactive user interfaces and communicate with a server-side backend.
- They can support a wide range of functionalities, including data input, processing, storage, and complex user interactions.