B. Bhavana

Assignment - 1 (Versions of java)

Java 1.0
The first version was released on January 23, 1996. The first stable version, JDK 1.0.2, is called Java 1.

Java 1.1 - February 19, 1997
Inner classes added to the language
Java beans
Java Database Connectivity (JDBC)
Collections framework

Java 1.3 - MAY 8, 2000
JavaSound
Synthetic proxy classes
HotSpotJVM

Java 1.4 - February 6, 2002
Regular Expressions
Exception Handling
Preferences API
Non-blocking I/O

Java 5.0 - September 30, 2004
Generics
Metadata
Enumerations
Static imports

Java 6 - November 11, 2006
Scripting Language Support
Java Compiler API
Many GUI improvements

Java 7 - July 28, 2011
Strings in switch
Automatic resource management in try-statement aka try-with-resources statement
Binary integer literals

Java 8 - March 18, 2014
Repeating annotations
Date and Time API
Remove the permanent generation

Java 9 - September 21, 2017
JavaDB was removed from JDK
Compact Strings

Java 10 - March 20, 2018
Garbage Collector Interface

Thread-Local Handshakes
Local Variable Type interface

Java 11 - September 25, 2018
Dynamic Class-File Constants
Local-Variable Syntax for Lambda Parameters
HTTP Client (Standard)

Java 12 - March 19, 2019
JVM Constants API
Switch Expressions

Java 13 - September 17, 2019
Reimplement the Legacy Socket API
Dynamic CDS Archives

Java 14 - March 17, 2020
Pattern Matching for instanceof
Packaging Tool (Incubator)
Switch Expressions (Standard)

Java 15 - September 15, 2020
Sealed Classes (Preview)
Hidden Classes
Deprecate RMI Activation for Removal

Java 16 - March 16, 2021
Elastic Metaspace
Packaging Tool
Records

Java 17 - September 14, 2021
Strongly Encapsulate JDK Internals
Remove RMI Activation
Vector API (Second Incubator)

Java 18 - March 22, 2022
Simple Web Server
Vector API

Java 19 - September 20, 2022
Record Patterns (Preview)
Virtual Threads
Structured Concurrency

Java 20 - March 21, 2023
Scoped Values (Incubator)
Pattern Matching for switch (Fourth Preview)
Vector API (Fifth Incubator)

Java 21 - September 19, 2023
String Templates (Preview)
Key Encapsulation Mechanism API
Unnamed Classes and Instance Main Methods (Preview)

Java 22 - March 19, 2024
Class-File API (Preview)

Stream Gatherers (Preview)
Structured Concurrency (Second Preview)

Assignment - 2 (Features of java)

1) Simple and easy to learn:

- Java comprises the same syntax as C, and C++.
- It holds automatic garbage collection features.
- Java eliminated its unused features.

2) object-oriented programming

Object: Object is a real-world entity in Java that encompasses state, functionality, and identity.
Class: Class is a logical entity which includes a group of objects with common properties. It contains fields, methods, constructors, blocks, nested classes and interfaces.
Inheritance: It's a concept in Java through which developers can create new classes built upon existing classes to achieve runtime polymorphism.
Polymorphism: A Mechanism in Java through which you can perform a single action in multiple ways. Polymorphism can be of two types- Compile time and runtime.
Abstraction: It's a method to hide internal processing and show only essential things to the users.

3) Platform Independent:

It uses a runtime environment of its own, i.e. JVM.
It is a write-once, run-anywhere language.
It is a software-based platform that runs on top of other hardware-based platforms.
Its code can be executed on multiple platforms, including Windows, Linux, Sun Solaris, and Mac/OS.
The Java code is compiled by the compiler and converted into bytecode.

4) security:

Java programming language runs inside a virtual machine.
It uses its own runtime environment- JVM.
Java includes a security manager, which determines what resources a class can access, such as reading and writing to the local disk.
In Java run time, a class loader separates the package for the classes of the local file system from the files imported from network sources.
Java also consists of Bytecode Verifier, which checks the code fragments for illegal code.

5) Rich API

Java Advanced Imaging (JAI)
Java Data Objects (JDO)
Java Media Frameworks (JMF)
Java Persistence API (JPA)
Java 3D (J3D)

6) Multithreading:

It lets Java developers execute multiple threads at the same time.
It's used to achieve multitasking.
It saves time.
It's mostly used in games and animation.

Threads work independently and don’t impact other threads, even if created simultaneously.

7) High-performance:

Java uses bytecode that can be easily translated into native machine code.
It has multiple easy-to-use frameworks.
It is compatible with multiple platforms, including Windows, Linux, Sun Solaris, and Mac/OS.
It is a write-once, run-anywhere language.
Java also automatically clears the garbage to enhance its performance.

8) scalability:

Java is an object-oriented programming language.
It has the ability to handle large databases.
Java doesn’t require multiple resources while running.
It uses multithreading and multiprocessing.
Java includes a higher volume of code.