Java Programming Language

Last Updated : 16 Nov, 2021

Java is one of the most popular and widely used programming languages.

- Java has been one of the most popular programming languages for many years.
- Java is Object Oriented. However, it is not considered as pure object-oriented as it provides support for primitive data types (like int, char, etc)
- The Java codes are first compiled into byte code (machine-independent code). Then the byte code runs on **J**ava **V**irtual **M**achine (JVM) regardless of the underlying architecture.
- Java syntax is similar to C/C++. But Java does not provide low-level programming functionalities like pointers. Also, Java codes are always written in the form of classes and objects.
- Java is used in all kinds of applications like Mobile Applications (Android is Java-based), desktop applications, web applications, client-server applications, enterprise applications, and many more.
- When compared with C++, Java codes are generally more maintainable because Java does not allow many things which may lead to bad/inefficient programming if used incorrectly. For example, non-primitives are always references in Java. So we cannot pass large objects (like we can do in C++) to functions, we always pass references in Java. One more example, since there are no pointers, bad memory access is also not possible.
- When compared with Python, Java kind of fits between C++ and Python. The programs are written in Java typically run faster than corresponding Python programs and slower than C++. Like C++, Java does static type checking, but Python does not.

Simple Hello World Program :

```
// A Java program to print "Hello World"
public class GFG {
    public static void main(String args[])
    {
        System.out.println("Hello World");
    }
```

```
}
```

Output:

```
Hello World
```

Recent Articles on Java

Java Programs – Basics to Advanced

Overview, Basics, Input/Output, Flow Control, Operators, Strings, Arrays, OOPs Concepts, Inheritance, Abstraction, Encapsulation, Polymorphism, Constructors, Methods, Interfaces, Wrapper Classes, Keywords, Access Modifiers, Memory Allocation, Classes, Packages, Collection Framework, List, Queue, Map, Set, Exception Handling, Multithreading, Synchronization, File Handling, Regex, Java IO, Networking, Java 8 Features, Date & Time, JDBC, Miscellaneous, Interview Questions

Overview of Java

- Introduction to Java
- History of Java
- Java vs C++ Python
- How to Download and Install Java?
- Setting Up the Environment in Java
- How to Download and Install Eclipse on Windows?
- Java Development Kit (JDK) in Java
- JVM and its architecture
- Differences between JDK, JRE, and JVM
- Just In Time Compiler
- Difference Between JIT and JVM
- Difference Between Byte Code and Machine Code
- How is the Java platform independent?

Basics of Java

- Java Basic Syntax
- First Java Program (Hello World)
- Datatypes in Java
- Difference between Primitive and Non-Primitive Datatypes
- Java Identifiers
- Operators in Java
- Java Variables

Packages in Java

- Java Packages
- How to create a package in Java
- java.util package
- java.lang package
- java.io package

Collection Framework

- Java Collection Framework
- Collections class in Java
- Collection Interface in Java
- How to learn Java collections
- List Interface in Java
- Queue Interface in Java
- Map Interface in Java
- Set in Java
- Iterator in Java
- Comparator in Java
- Difference between Comparator and Comparable in Java

List

- Scope of Variables
- Wrapper Classes in Java

Input/Output in Java

- How to take Input from users in Java
- Scanner class in Java
- BufferedReader class in Java
- Scanner vs BufferedReader in Java
- Ways to Read Input from Console in Java
- Print Output in Java
- Difference between print() and println() in Java
- Formatted Outputs in Java
- Fast Input-Output for Competitive Programming in Java

Flow Control in Java

- Decision making in Java
- If Statement in Java
- If-Else Statement in java
- If-Else-If ladder in Java
- Loops in Java
- For loop
- While Loop
- Do while loop
- For each loop
- Continue Statement in java
- Break Statement In Java
- Usage of Break in Java
- Return Statement in Java

Operators in Java

- Arithmetic Operator
- Unary Operator
- Assignment Operator
- Relational Operator
- Logical Operator
- Ternary Operator
- Bitwise Operator

- ArrayList in Java
- Vector class in Java
- Stack class in Java
- LinkedList in Java
- AbstractList
- AbstractSequentialList
- CopyOnWriteArrayList
- Custom ArrayList in Java

Queue

- AbstractQueue
- ArrayBlockingQueue
- ConcurrentLinkedQueue
- LinkedBlockingQueue
- LinkedTransferQueue
- PriorityBlockingQueue
- Deque in Java
- ArrayDeque
- Concurrent LinkedDeque
- LinkedBlocking Deque
- Priority Queue in Java

Map

- EnumMap
- HashMap
- Working of HashMap
- Traverse through a HashMap in Java
- WeakHashMap
- LinkedHashMap
- IdentityHashMap
- ConcurrentHashMap
- Dictionary
- HashTable
- SortedMap
- TreeMap
- Stack
- Vector

Set

- AbstractSet

Strings in Java

- Introduction of Strings in Java
- String class in Java Set-1 | Set-2
- Why strings are immutable in Java?
- StringBuffer class in Java
- StringBuilder class in Java
- Strings vs StringBuffer vs StringBuilder in Java
- StringTokenizer class in Java Set-1 | Set-2
- StringJoiner in Java
- Java String Programs

Arrays in Java

- Introduction to Arrays in Java
- Arrays class in Java
- Multi-Dimensional Array in Java
- How to declare and initialize 2D arrays in Java
- Jagged array in Java
- Final Arrays in Java
- Reflect Arrays in Java
- Difference between util.Arrays and reflect.Arrays
- Java Array Programs

OOPS in Java

- OOPS Concept in Java
- Why Java is not a purely Object-Oriented Language?
- Classes and Objects
- Naming Convention in Java
- Methods in Java
- Access Modifiers in Java
- Constructors in Java
- Four pillars of OOPS in Java
- Inheritance in Java
- Abstraction in Java
- Encapsulation in Java
- Polymorphism in Java
- Interfaces in Java
- EnumSet
- HashSet
- TreeSet
- SortedSet
- LinkedHashSet
- NavigableSet
- ConcurrentSkipListSet
- CopyOnWriteArraySet

Exception Handling in Java

- Exceptions in java
- Types of Exceptions
- Difference between Checked and Unchecked Exceptions
- Try, Catch, Finally, throw, and throws
- Flow control in Try catch block
- Throw vs Throws
- Final vs Finally vs Finalize
- User-defined custom exception
- Chained Exceptions
- Null pointer Exceptions
- Exception handling with method Overriding

Multithreading in Java

- Introduction to Multithreading in Java
- Lifecycle and Stages of a Thread
- Thread Priority in Java
- Main Thread in Java
- Thread class
- Runnable interface
- How to name a thread
- start() method in thread
- Difference between run() and start() Method
- sleep() method
- Daemon thread
- Thread Pool in Java
- Thread Group in Java
- Thread Safety in Java
- ShutdownHook

- This reference in Java

Inheritance in Java

- Introduction to Inheritance in Java
- Inheritance and Constructors
- Multiple Inheritance in Java
- Interfaces and Inheritance
- Difference between Inheritance in C++ and Java

Abstraction in Java

- Introduction to Abstraction in Java
- Abstract Keyword in Java
- Abstract classes in Java
- Abstract class vs Interface in Java
- Control Abstraction in Java
- Difference between Data Hiding and Abstraction

Encapsulation in Java

- Introduction to Encapsulation in Java
- Difference between Encapsulation and Abstraction

Polymorphism in Java

- Introduction to Polymorphism in Java
- Difference between Inheritance and Polymorphism
- Runtime Polymorphism in Java
- Compile-Time vs Runtime Polymorphism

Constructors in Java

- Introduction to Constructors in Java
- Copy Constructor in Java
- Constructor Overloading
- Constructor Chaining
- Private Constructors and Singleton Class

- Multithreading Tutorial

Synchronization in Java

- Java Synchronization
- Importance of Thread synchronization in Java
- Method and Block Synchronization in Java
- Local frameworks vs thread synchronization
- Difference between Atomic, Volatile, and Synchronized in Java
- Deadlock in Multithreading
- Deadlock Prevention and Avoidance
- Difference between Lock and Monitor in Concurrency
- Reentrant Lock

File Handling in Java

- File Class in java
- How to create files in java
- How to read files in java
- How to write on files in java
- How to delete a file in java
- File Permissions
- FileReader
- File Writer
- FileDescriptor class
- RandomAccessFile class

Java Regex

- Introduction to Java Regex
- How to write Regex expressions
- Matcher class
- Pattern class
- Quantifiers
- Character class

Java IO

- Introduction to Java IO

Methods in Java

- Introduction to methods in Java
- Different method calls in Java
- Difference between Static methods and Instance methods in Java
- Abstract methods in Java
- Method Overriding in Java
- Method Overloading in Java
- Method Overloading Vs Method Overriding

Interfaces in Java

- Java Interfaces
- Interfaces and Inheritance in Java
- Difference between Interface and Class in Java
- Functional Interface
- Nested Interface
- Marker Interface
- Comparator Interface

Wrapper Classes in Java

- Need of Wrapper classes in Java
- How to create instances of Wrapper classes
- Character class in Java
- Byte class in Java
- Short class in Java
- Integer class in Java
- Long class in Java
- Float class in Java
- Double class in Java
- Boolean class in Java
- Autoboxing and Unboxing
- Type Conversion in Java

Keywords in Java

- List of all Java Keywords
- Important Keywords in Java
- Reader Class
- Writer Class
- FileInput stream
- File Output stream
- BufferedReader Input Stream
- BufferedReader Output stream
- BufferedReader vs Scanner
- Fast I/O in Java

Java Networking

- Introduction to Java Networking
- TCP architecture
- UDP architecture
- IPV4 vs IPV6
- Connection-oriented vs connectionless protocols
- Socket programming in Java
- Server Socket class
- URL class and methods

Java SE 8 Features

- Lambda Expressions
- Streams API
- New Date/Time API
- Default Methods
- Functional Interfaces
- Method references
- Optional class
- Stream Filter
- Type Annotations
- String Joiner

Java Date & Time

- Date Class in Java
- Methods of the Date class
- Java Current Date and time
- Compare dates in Java

Java JDBC

- Super Keyword
- Final Keyword
- Abstract keyword
- Static Keyword
- This Keyword
- Enum Keyword in Java
- Transient keyword in java
- Volatile keyword in java
- Final, Finally, and Finalize in Java

Access Modifiers in Java

- Introduction to Access Modifiers in Java
- Public vs Protected vs Package vs Private Access Modifier in Java
- Access Modifiers Vs Non-Access Modifiers in Java

Memory Allocation in Java

- Java Memory Management
- How are Java objects stored in memory
- Stack vs Heap memory allocation
- Types of memory areas allocated by JVM
- Garbage Collection in Java
- Heap and Stack memory allocation
- Types of JVM Garbage Collectors in Java
- Memory leaks in Java
- Java Virtual Machine(JVM) Stack Area

Classes of Java

- Classes and Objects
- Understanding classes and objects in Java
- Class vs interface
- Singleton class in java
- Object class in java
- Inner class in java
- Abstract classes in java
- Throwable class in java

- Introduction to Java JDBC
- JDBC Driver
- JDBC Connection
- Types of Statements in JDBC
- JDBC Tutorial

Java Miscellaneous

- Introduction to Reflection API
- Java IO Tutorial
- JavaFX Tutorial
- Java RMI
- How to Run Java RMI application?
- Java 17 New Features

Interview Questions on Java

- Commonly Asked Java Interview Questions Set – 1 | Set – 2
- Interview Questions For Java Professionals
- 10 Most Asked Questions on Java
- Java Multiple Choice Questions

GeeksforGeeks Courses: