Java

Course Outline

INTRODUCTION TO JAVA

- Why Java was Developed
- Application Areas of Java
- History of Java
- Platform Independency in Java
- USP of Java: Java Features
- Sun-Oracle Deal
- Different Java Platforms
- Difference between JDK, JRE, JVM
- Java Versions
- JVM Architecture
- Installing Java on Windows
- Understanding Path Variable: Why Set Path

CREATING FIRST JAVA PROGRAM

- Understanding Text Editors to Write Programs
- How to Compile Java File
- Byte Code and Class File
- How to Run Class File

JAVA LANGUAGE FUNDAMENTALS

- Identifiers
- Keywords
- Variables
- Literals
- Data Types
- Operators
- Comments
- Looping Statements
- Condition Statements
- Type Casting

OOP IMPLEMENTATION (PIE)

- Why OOP
- OOP Concepts with Real-Life Examples
- Class & Its Syntax
- Object & Its Syntax
- Reference Variable
- Constructors
- Instance (Non-Static) & Static Variables
- Instance (Non-Static) & Static Methods
- this Keyword and Its Usages
- Object & Static Initializers (Anonymous Blocks)

- Understanding '+' Operator
- Inheritance & Its Syntax
- Types of Inheritance
- Object Class as Root of Java Class Hierarchy
- Variable Hiding
- Method Hiding
- Method Overriding
- Method Overloading
- super Keyword and Its Usages
- final Keyword and Its Usages
- Constructor Chaining
- Upcasting and Downcasting
- Static & Dynamic Binding
- Run-Time Polymorphism
- abstract Keyword (Abstract Classes and Methods)
- Understanding Interfaces
- Implementation of Encapsulation
- Association with Implementation

PACKAGES

- Understanding Packages
- Setting Class Path
- Reading Input from Keyboard
- Access Modifiers

NESTED TYPES

- Non-Static Nested Class
- Local Class
- Anonymous Class
- Nested Interface

ARRAYS

- General Definition of Array
- Advantages from Array
- Arrays in Java
- 1-D Arrays
- 2-D Arrays
- Jagged Arrays
- Array of Reference Type
- Operations on Arrays

COMMAND LINE ARGUMENTS AND WRAPPER CLASSES

- How to Read Command Line Arguments
- Wrapper Classes
- Parsing of Numeric Strings
- String Representation of Primitives

EXCEPTION HANDLING

- Types of Runtime Errors
- Understanding Exceptions
- Exception Class Hierarchy
- Try & Catch Blocks
- Patterns of Catch Block
- Nested Try Statements
- throw, throws, and finally
- Creating Custom Exceptions
- Checked & Unchecked Exceptions
- Assertion

WORKING WITH STRINGS

- What is String
- String Class
- Creating String Object
- Operations on String
- String Buffer Class and Its Methods
- Difference Between String and StringBuffer Class
- String Builder Class and Its Methods
- Difference Between StringBuffer and StringBuilder

SWING

- Introduction to AWT
- Introduction to Swing Components
- Look and Feel of Swing Components
- MVC Architecture of Swing Components
- Working with Image
- Advanced Swing Components
- JOptionPane, JTree, JTable, JTabbedPane
- JFileChooser, JColorChooser
- Menu Components: JMenu, JMenuItem, JMenuBar

MULTITHREADED PROGRAMMING

- Multitasking: Why Concurrent Execution
- Multiprocessing vs Multithreading
- Main Thread (Default Java Thread)
- Creating Child Threads and Understanding Context Switching
- Thread States
- Thread Group
- Thread Synchronization: Methods and Blocks
- Inter-Thread Communication
- Daemon Threads
- Deadlock

I/O STREAMS

- What is I/O

- Why Need Streams
- Byte Streams and Character Streams
- Read/Write Operations with File
- Scanner Class
- Object Serialization & Deserialization
- transient Keyword
- File Class and Its Methods

SOCKET PROGRAMMING

- Understanding Fundamentals of a Network
- Socket and ServerSocket Classes
- InetAddress Class
- DatagramSocket and DatagramPacket Classes
- URL, URLConnection, HttpURLConnection Classes

REFLECTION

- Understanding the Need of Reflection
- Getting Information About Class's Modifiers, Fields, Methods, Constructors, and Super Classes
- Finding Out Constant and Method Declaration Belong to an Interface
- Creating an Instance of the Class Whose Name is Not Known Until Runtime
- Getting and Setting Values of an Object's Field if Field Name is Unknown Until Runtime
- Invoking a Method on an Object if the Method is Unknown Until Runtime
- Invoking Private Methods

EXTENDED & UTILITY CONCEPTS

- Generics
- Lambda Expression
- Annotations
- Object Cloning
- varargs
- Static-Import
- Enum
- Static, Default, and Private Methods of Interface
- var Type
- Java Modules

COLLECTIONS FRAMEWORK

- What is Collection?
- What is Framework?
- Collections Framework
- Core Interfaces: Collection, List, Queue, Deque
- Set, NavigableSet, SortedSet
- Map, NavigableMap, SortedMap
- Core Classes:
    - ArrayList, LinkedList, PriorityQueue, ArrayDeque

  - HashSet, LinkedHashSet, TreeSet
  - HashMap, IdentityHashMap, WeakHashMap, LinkedHashMap, TreeMap
- Accessing a Collection via an Iterator
- Accessing List via ListIterator
- Accessing a Collection via For-Each Loop
- Working with User-Defined Objects
- The Comparator and Comparable Interfaces
- The Legacy Classes and Interfaces:
  - Enumeration, Vector, Stack
  - Hashtable, Properties

DATE & TIME API

- java.util.Date
- java.util.Calendar
- java.sql.Date
- JODA API
- java.time.LocalDate
- java.time.LocalTime
- java.time.LocalDateTime

SYSTEM PROPERTIES & INTERNATIONALIZATION

- Understanding Locale
- Resource Bundle
- Usage of Properties File
- Fetching Text from Resource Bundle
- Displaying the Text in Hindi
- Display