# OOPS using JAVA

**Course Content:**

| Modules | Topics Coverage |
| --- | --- |
| Module 1 | **Java Fundamentals**: Introduction, Application areas of Java, Platform Independency in Java, Java Features, Difference between JDK, JRE, JVM, JVM Architecture, Data types, variable, operators, keywords and control Statements, Java coding standards. |
| | **OOPS using Java:** Class, Object, 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; |
| | Array: Introduction, 1-D, 2-D Array, Jagged Array; |
| | Classes: Types of classes (abstract class, final class, Inner classes, Static class, Singleton class); Inheritance, Polymorphism, Encapsulation, Interfaces; |
| | Methods (types and Overloading/Overriding); Wrapper Classes. |
| | **Exception Handling:** Understanding Exceptions, Exception Hierarchy, try and catch blocks, Nested try statements, throw, throws, finally, Creating Custom Exceptions, Checked & Unchecked Exceptions. |
| | **String Classes:** String, StringBuffer, StringBuilder, StringJoiner, StringTokenizer classes and its methods. |
| Module 2 | **Java Packages:** Definition and Purpose of Packages: Organizing Classes and Interfaces, Avoiding Name Conflicts; Creating Java Packages; Importing Java Packages; Static Imports; Access Modifiers and Packages; Access Control and Packages; Abstracting Program Logic to Packages; Creating Executable Main Class; Running the Executable Class Inside a Package. |
| | **Collection Framework:** Introduction to Collection Framework, Collection Hierarchy. List: ArrayList, LinkedList; Set: TreeSet, HashSet; Map: HashMap, TreeMap; Comparable and Comparator Interface. |
| | **Multithreading:** Multithreading in Java, Thread LifeCycle, Thread Synchronization (Synchronized method, Synchronized block, Static synchronization). |
| | **JDBC:** Database connectivity, Types of Drivers for connection, Connection Example. CRUD operations using Database using Statement/ PreparedStatement/CallableStatement interface. JDBC – joining of tables and applying criteria, Configuring various types of drivers for Java Database Connectivity. |
| | **Advanced features:** Lambda Expressions, Method References, Functional Interfaces, Stream API, default methods and static methods in interface, private methods in interfaces, Optional class, Collectors class, ForEach() method, Local-Variable Type Inference (var), Optional.orElseThrow() method |