



# **ALTALUNE TECHNOLOGY**

## **CORE JAVA**

### **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& it's Syntax
- ❖ Object& it's Syntax
- ❖ Reference Variable
- ❖ Constructors
- ❖ Instance(Non-Static)& Static Variables
- ❖ Instance(Non-Static) & Static Methods



# **ALTALUNE TECHNOLOGY**

- ❖ this Keyword and it's usages
- ❖ Object & Static Initializers(Anonymous Blocks)
- ❖ Understanding '+' Operator
- ❖ Inheritance& it's Syntax
- ❖ Types of Inheritance
- ❖ Object Class as Root of Java Class Hierarchy
- ❖ Variable Hiding
- ❖ Method Hiding
- ❖ Method Overriding
- ❖ Method Overloading
- ❖ Super keyword and it's usages
- ❖ Final keyword and it's 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**

- ❖ Static Nested Class
- ❖ 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



# **ALTALUNE TECHNOLOGY**

## **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 it's Methods
- ❖ Difference between String and String Buffer class
- ❖ String Builder Class and it's Methods
- ❖ Difference between String Buffer and String Builder

## **SWING**

- ❖ Introduction to AWT
- ❖ Introduction to Swing Components
- ❖ Look And Feel of Swing Components
- ❖ MVC Architecture of Swing Components
- ❖ Working with Image
- ❖ Advance Swing Components
- ❖ JOptionPane, JTree, JTable, JTabbedPane
- ❖ JFileChooser, JcolorChooser
- ❖ Menu Components
- ❖ JMenu
- ❖ JMenuItem
- ❖ JMenuBar

## **MULTITHREADED PROGRAMMING**

- ❖ Multitasking: Why Concurrent Execution
- ❖ Multiprocessing v/s Multithreading



# **ALTALUNE TECHNOLOGY**

- ❖ 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
- ❖ Vargs
- ❖ Static-import
- ❖ Enum
- ❖ Static, Default and Private Methods of Interface
- ❖ Var Type



# **ALTALUNE TECHNOLOGY**

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.LocalDateTime
- ❖ java.time.LocalTime

## **SYSTEM PROPERTIES & INTERNATIONALIZATION (I18N)**

- ❖ Understanding Locale
- ❖ Resource Bundle
- ❖ Usage of properties file
- ❖ Fetching text from Resource Bundle
- ❖ Displaying the text in HINDI
- ❖ Displaying date in Hindi

## **INTRODUCTION TO SQL (PROJECT BASED)**

### **DATABASE PROGRAMMING USING JDBC**

- ❖ Need Of JDBC
- ❖ JDBC Drivers
- ❖ Statement, Prepared Statement, Callable Statement

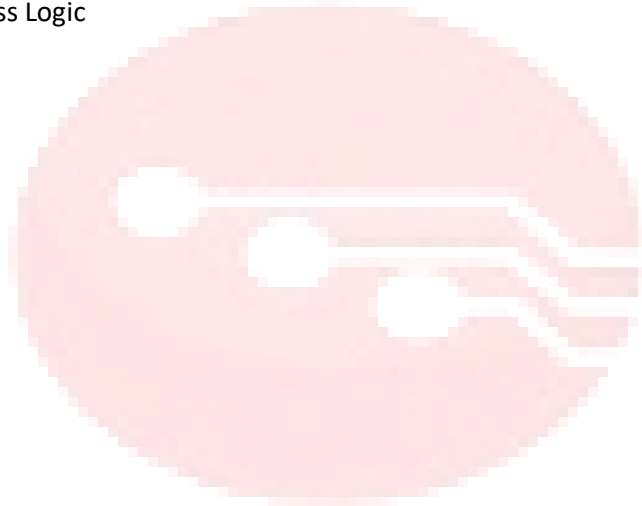


# **ALTALUNE TECHNOLOGY**

- ❖ Scrollable and Updatable Result Set
- ❖ Batch Updates
- ❖ Transaction
- ❖ Metadata

## **PROJECT CLASSES**

- ❖ Front End Coding
- ❖ Form Designing
- ❖ Back End Coding
- ❖ Database Designing
- ❖ Connecting forms to database
- ❖ Writing Business Logic



ALTALUNE  
TECHNOLOGY