



Course Overview

This instructor-led training provide candidates to the development technologies using the Java. Java is a modern, object-oriented language created by Sun Microsystems. The major advantages of Java includes its portability and use on the Internet.

This hands-on course provides participants with the knowledge and experience required to develop and deploy robust Java applications. Comprehensive hands on exercises are integrated throughout to reinforce learning and develop real competency. Under this course

Audience Profile

This course is intended for programmers who are evaluating Java, and provides various resources where they can obtain specific skills for creating applications, components, and services built using Java skills & programming.

At Course Completion

After completing this course, students will have a fundamental understanding of how to:

- Design, create, build, and debug Java applications and applets.
- Apply algorithmic thinking to solve programming problems.
- Implement syntax rules in Java programs.
- Explain variables and data types used in program development.
- Apply arithmetic operations for displaying numeric output.
- Write and apply decision structures for determining different operations.
- Write and apply loop structures to perform repetitive tasks.
- Write user-defined methods.
- Identify and implement arrays, array lists, and multidimensional arrays.
- Write Java programs using object-oriented programming techniques including classes, objects, methods, instance variables, composition, inheritance, and polymorphism.
- Write programs using graphical user interface (GUI) components and Java's Event Handling Model

J2SE (JAVA STANDARD EDITION)

Environment: JAVA 1.8

Course Duration: 40 Hours

Course Outline

- **Basics of Java**

- Java - What, Where and Why?
- History and Features of Java
- Internals of Java Program
- Difference between JDK, JRE and JVM
- Internal Details of JVM
- Variable and Data Type
- Unicode System
- Naming Convention

Lab Activities:

- Hello World
- If else, For loop, While loop
- Print Alphabets
- Print Multiplication Table
- Find Odd or Even
- Fahrenheit to Celsius

- **Advantage of OOPs**

- Object and Class
- Method Overloading
- Constructor
- static variable, method and block
- Member Inner class
- Anonymous Inner class
- Local Inner class
- static nested class
- Nested Interface
- this keyword
- Inheritance (IS-A)
- Aggregation and Composition(HAS-A)
- Method Overriding
- super keyword
- Instance Initializer block
- final keyword
- Runtime Polymorphism
- static and Dynamic binding
- Abstract class and Interface
- Downcasting with instance of operator
- Package and Access Modifiers
- Encapsulation
- Object class
- Object Cloning
- Java Array
- Call By Value and Call By Reference

Lab Activities:

- Java Methods
- Static Block
- Static Method
- Multiple classes
- Java constructor tutorial
- Calculate Circle Area and Perimeter using Java Example
- Calculate Rectangle Area and Perimeter using Java Example
- Even Odd Number Example
- Find Largest and Smallest Number in an Array Example

- **String: What and Why?**

- Immutable String
- String Comparison
- String Concatenation
- Substring
- Methods of String class
- StringBuffer class
- StringBuilder class
- Creating Immutable class
- toString method
- StringTokenizer class

Lab Activities:

- Creating String by java string literal
- Concatenation of String
- String compare by equals() method
- String compare by == operator
- String compare by compareTo() method
- Substring
- Java String toUpperCase and toLowerCase
- String charAt
- String Buffer append
- String Buffer replace
- String Buffer delete
- String Buffer reverse
- String Builder append
- String Builder replace
- String Builder delete
- String Builder reverse

- **Exception Handling**

- Exception Handling: What and Why?
- try and catch block
- Multiple catch block
- Nested try
- finally block
- throw keyword
- Exception Propagation
- throws keyword
- Exception Handling with Method Overriding
- Custom Exception
- Nested Classes

Lab Activities:

- Java exception handling tutorial
- Test Multiple Catch Block
- Nested try and catch block
- finally block
- Test Throw
- Exception Propagation
- Throws
- Custom Exception

• Multithreading

- Multithreading: What and Why?
- Life Cycle of a Thread
- Creating Thread
- Thread Scheduler
- Sleeping a thread, Joining a thread
- Thread Priority
- Daemon Thread
- Thread Pooling , Thread Group
- ShutdownHook

• Synchronization

- Synchronization: What and Why?
- synchronized method
- synchronized block
- static synchronization
- Deadlock
- Inter-thread Communication
- Interrupting Thread
- Input and output

Lab Activities:

- Create By extending Thread
- Creating By implementing the Runnable interface
- using Use Sleep
- Can we start a thread twice
- What if we call run () method directly instead start () method?
- The join
- *currentThread*
- Naming a Thread
- Priority
- Daemon Thread, Thread pool, Thread Group
- Shutdown Hook
- Multitasking
- Finalize
- Memory
- Thread synchronized
- Deadlock Example
- Inter-thread communication

- **FileOutputStream & FileInputStream**
 - ByteArrayOutputStream
 - SequenceInputStream
 - BufferedOutputStream & BufferedInputStream
 - FileWriter&FileReader
 - CharArrayWriter
 - Input from keyboard by InputStreamReader
 - Input from keyboard by Console
 - Input from keyboard by Scanner
 - PrintStream class
 - PrintWriter class
 - Compressing and Uncompressing File
 - Reading and Writing data simultaneously
 - DataInputStream and DataOutputStream
 - StreamTokenizer class
 - Serialization

Lab Activities:

- Accessing a data of a file
- File Output Stream, File Input Stream
- Reading the data of current java
- Byte Array Output Stream
- Sequence Input Stream
- Buffered Output Stream
- Buffered input Stream
- File Writer and File Reader
- Merging data of two different files into one file

- **Socket Programming**
 - URL class
 - Displaying data of a web page
 - InetAddress class
 - DatagramSocket and DatagramPacket
 - Two way communication

Lab Activities:

- IP Address

- **Life Cycle of Applet**
 - Graphics in Applet
 - Displaying image in Applet
 - Animation in Applet
 - EventHandling in Applet
 - Painting in Applet
 - Parameter in Applet
 - Applet Communication

- **AWT Controls**
 - Event Handling by 3 ways
 - Event classes and Listener Interfaces
 - Adapter classes
 - Creating Games and Applications

Lab Activities:

- Using action listener
- Implementing keyboard listeners

• Border Layout

- GridLayout, FlowLayout, BorderLayout, CardLayout

• Basics of Swing

- JButton class, JRadioButton class, JTextArea class, JComboBox class, JTable class, JColorChooser class, JProgressBar class, JSlider class
- Graphics in swing
- Displaying Image
- Menubar
- Open Dialog Box

Lab Activities:

- Creating GUI screens with graphics
- Using images on a Swing Application

• Collection Framework

- ArrayList class, LinkedList class, HashSet class, LinkedHashSet class, TreeSet class, PriorityQueue class, ArrayDeque class, HashMap class, LinkedHashMap class, TreeMap class, Hashtable class
- ListIterator interface, Map interface
- Comparable and Comparator

Lab Activities:

- Reverse number
- Add Matrices
- Transpose Matrix
- Multiply Matrices
- Bubble sort
 - JButton class, JRadioButton class, JTextArea class, JComboBox class, JTable class, JColorChooser class, JProgressBar class, JSlider class
 - Digital Watch
 - Displaying image
 - Edit menu code for Notepad
 - Open Dialog Box
 - Notepad
 - Border Layout, Grid Layout, Flow Layout, Card Layout

• JDBC

- Steps to connect to the database
- DriverManager
- Connection interface, Statement interface, ResultSet interface
- PreparedStatement
- ResultSetMetaData
- DatabaseMetaData

Lab Activities:

- Making a login page and validating data with help of database
- Insert data in table, Show records, Update table data