# NKXGEN

### Next Generation Paradigms

**Java**

## Course Content

Mentor : SreedharKosaraju

No of Hours: 45

**Prepared by: NKXGEN Talent Development Team**

# Java (45 Hours)

## ❖ Java Features

- ➢ Features of Java
- ➢ Why Java is Powerful?
- ➢ Platform Neutrality
- ➢ Java Programming Model

## ❖ Applications Overview

- ➢ S/W Applns
- ➢ Web Applns
- ➢ Web vs Windows Applns
- ➢ Java in Web Applns

## ❖ Java Environment

- ➢ Java History
- ➢ Java Framework
- ➢ Java SE Overview
- ➢ Java EE Overview

## ❖ Java Dev Environment

- ➢ Simple Java Program
- ➢ Java IDE's
- ➢ Runtime Environ
- ➢ JVM Architecture
- ➢ Steps to execute program

## ❖ Java Lang-I

- ➢ Data Types
- ➢ Reference Types
- ➢ Stack and Heap
- ➢ Operators
- ➢ Type Casting

## ❖ Java Lang-II

- ➢ Control Structures
- ➢ Reference Types
- ➢ Input & Output
- ➢ Command Line Args
- ➢ Arrays & foreach loop

## ❖ OOP Concepts

- ➢ Class & Objects
- ➢ Why OOP?
- ➢ OOP vs Structured
- ➢ OOP Features

## ❖ OOP Concepts

- ➢ Encapsulation
- ➢ Data Abstraction
- ➢ Writing Classes
- ➢ Creating Objects

➢ Constructor Overloading

❖ **Class Internals**

➢ Instance & Static Members

➢ Instance & Static Methods

➢ Static blocks

➢ Overloading of methods

➢ Passing Objects as Arguments

❖ **Strings**

➢ Strings

➢ String Pool Concept

➢ String Methods

➢ String Buffer & Builder

➢ String Manipulations

❖ **Wrapper Classes**

➢ Wrapper classes

➢ Primitives as Objects

➢ Important Methods

➢ Autoboxing & Unboxing

❖ **Inheritance**

➢ Inheritance basics

➢ Types of Inheritances

➢ Overriding

> ➢ Co-Variant Return Types

## ❖ Inheritance

- ➢ Constructor Calling Order
- ➢ Object Type Casting
- ➢ Up & Down Casting
- ➢ Early & Late Binding
- ➢ Dynamic Method Dispatch

## ❖ Abstract Classes & Interfaces

- ➢ Hirearchical Abstraction
- ➢ Final Classes & Methods
- ➢ Writing Abstract Classes
- ➢ Interfaces
- ➢ Why Interfaces

## ❖ Interfaces

- ➢ Interface Implentations
- ➢ Interface Features
- ➢ Interface Casting
- ➢ Interface Inheritance

## ❖ Polymorphism

- ➢ Polymorphism Features
- ➢ Polymorphism Implementation
- ➢ Polymorphism with Inheritance

- ➤ Polymorphism with Interfaces

## ❖ Exceptions

- ➤ Exceptions
- ➤ Exception Architecture
- ➤ Propagations
- ➤ Exception Hierarchy
- ➤ System Defined Exceptions

## ❖ Exceptions

- ➤ Checked & Unchecked
- ➤ User Defined Exceptions
- ➤ Exception Chaining
- ➤ Multi Catch

## ❖ Nested Classes

- ➤ Top Level Classes
- ➤ Static Nested Classes
- ➤ Non-Static Nested Classes
- ➤ Local Inner Class
- ➤ Anonymous Inner Class

## ❖ Threads

- ➤ Multithreading
- ➤ Thread Necessity
- ➤ Threads as Objects

- ➢ Creating Threads
- ➢ Thread States
- ➢ sleep Methods

## ❖ Threads

- ➢ Thread Synchronization
- ➢ Synchronized blocks
- ➢ Imp Thread Methods
- ➢ Monitors & Queues

## ❖ Threads & Packages

- ➢ Thread Priorities
- ➢ Thread Communication
- ➢ Why Packages
- ➢ Package Features

## ❖ Packages

- ➢ Creating packages
- ➢ Referring packages
- ➢ Built in packages
- ➢ Understanding Scopes

## ❖ Packages

- ➢ Creating Jar files
- ➢ Manifest file

- Using Jars
- Resolving Naming Conflicts

## ❖ Generics

- What is Generics
- Simple Generics
- Sub Typing
- Generic Methods
- Wild Cards in Generics

## ❖ Collections

- Collections vs Arrays
- Why Collections
- Collections Framework
- Classification in Collns
- Generics in Collns

## ❖ Collections

- Interfaces in Collns
- Collections and Map
- Adv Collns
- "Collections" class methods

## ❖ Util Classes

- Util Package
- StringTokenizer

- ➢ Scanner
- ➢ Date Time API
- ➢ Regular Expressions

## ❖ Memory Management & Reflection

- ➢ Garbage Collection
- ➢ Finalizer
- ➢ Cloning
- ➢ Reflection
- ➢ Reflection API

## ❖ I/o Streams

- ➢ IO Streams
- ➢ Character vs Byte
- ➢ Working with Files
- ➢ Working with Buffer Streams
- ➢ Data Streams

## ❖ I/O Streams

- ➢ Builtin Streams
- ➢ Reading from Console
- ➢ Serialization
- ➢ Object Streams
- ➢ Transient variables
- ➢ Deserialization

## ❖ Sockets

- ➢ Sockets & Ports
- ➢ Listeners
- ➢ Internet Addressing
- ➢ Socket API
- ➢ Simple Socket Communication
- ➢ Using Data Streams

## ❖ Sockets

- ➢ Handling multiple requets
- ➢ Multithreaded sockets
- ➢ Multicast sockets
- ➢ Using DataGram Sockets
- ➢ URL Sockets Overview

## ❖ AWT

- ➢ Foundation Classes
- ➢ AWT Components
- ➢ Frame Life Cycle
- ➢ Simple Frames
- ➢ Standard Controls
- ➢ AWT Graphics Object

## ❖ AWT

- ➢ Layout Management

- ➢ Builtin Layouts
- ➢ AWT Events
- ➢ Event Architecture
- ➢ Event Propogation

## ❖ AWT

- ➢ Listener Interfaces
- ➢ Event Classes
- ➢ Delegation Event Model
- ➢ Event Implementations
- ➢ Writing Inner Classes

## ❖ AWT

- ➢ More Event Implementations
- ➢ Adapters
- ➢ Dialogs
- ➢ Menus

## ❖ Applets

- ➢ Applets Features
- ➢ Applets Life Cycle
- ➢ Applets Life Cycle
- ➢ Security in Applets
- ➢ Sandbox Model
- ➢ Applets in HTML
- ➢ Using Applet Viewer

## ❖ Applets

- ➢ Images in Applets
- ➢ DocBase & CodeBase
- ➢ Communication in Applets
- ➢ Applet Context
- ➢ Playing Audio in Applets

## ❖ Swing at a Glance

- ➢ AWT Vs Swing
- ➢ Swing Components
- ➢ Simple Swing Applns
- ➢ Swing Events
- ➢ Swing Applets

## Point of Contacts

**Mr. K Vidya Sagar**
**Admin &Ops Manager**
**NKXGEN PARADIGMS Pvt. Ltd.**
 4th Floor, 71c Arunodaya Comple
,1$^{st}$ Lane, Dwaraka Nagar
Visakhapatnam 530016,
India Email: : enquiry@nkxgen.com

**M: +91 90009 23366**