

# Core Java

Duration: 24 Hours

## Java Basics

- Language and Platform Features
- Program Life Cycle
- The Java SE Development Kit (JDK)

## Class and Object Basics

- The Object Model and Object-Oriented Programming
- Classes, References, and Instantiation
- Adding Data to a Class Definition
- Adding Methods (Behavior)

## More on Classes and Objects

- Accessing data, the "this" variable
- Encapsulation and Access Control, public and private Access
- Constructors and Initialization
- static Members of a Class
- Scopes, Blocks, References to Objects

## Flow of Control

- Branching: if, if-else, switch
- Iteration: while, do-while, for, break, continue

## Strings and Arrays

- String,
- StringBuffer,
- StringBuilder
- Arrays,
- Primitive Arrays,
- Arrays of Reference Types

## Packages

- Package Overview - Using Packages to Organize Code
- import statements
- Creating Packages, package Statement, Required Directory Structure
- Finding Classes, Packages and Classpath

## Inheritance

- Using Composition to Deal With Complexity
- Composition/HAS-A, Delegation
- Using Inheritance and Polymorphism to share commonality
- IS-A, extends, Inheriting Features, Overriding Methods, Using Polymorphism
- Class Object
- Abstract Classes

## Interfaces

- Using Interfaces to Define Types
- Interfaces VS Abstract Classes
- Nested Interface

## Exceptions

- Exceptions and the Exception Hierarchy
- try and catch
- Handling Exceptions
- Program Flow with Exceptions
- Finally

## Thread Handling

- Delegation-Based Stream Model
- Life Cycle of a Thread
- Creating Thread
- Thread Scheduler
- Sleeping a thread
- Start a thread twice
- Calling run() method
- Joining a thread
- Naming a thread
- Thread Priority
- Thread Pool
- Thread Group
- Shutdown Hook
- Performing multiple task

### Java Collections

- The Collections Framework and its API
- Collections
- Collection, Set, List, Map, Iterator
- Autoboxing
- Collections of Object (non-generic)
- Using ArrayList, HashSet, and HashMap
- for-each Loop
- Processing Items With an Iterator

### File Handling I/O

- Delegation-Based Stream Model
- InputStream and OutputStream
- Media-Based Streams
- Filtering Streams
- Readers and Writers
- File Class
- Modeling Files and Directories
- File Streams
- Random-Access Files