Duration: 28 Classroom hours and **32** Lab hours

Objective: To introduce the student to Java programming

Prerequisites: Knowledge of programming in any language like C, C++

Evaluation method: Theory exam– 40% weightage
Lab exam – 40% weightage
Internal exam– 20% weightage

List of Books / Other training material

Courseware: No specific courseware for the module, faculty may share some coursematerials.

Reference Book:

1. Java - The Complete Reference by Herbert Schildt / Tata Mcgraw Hill Education
2. Java Server Programming (J2EE 1.7 Edition) Black Book by Dreamtech Software Team
3. Java 8 Programming Black Book by Dreamtech Press
4. Core Java : Fundamentals - Volume 1 Gary Cornell, Cay S. Horstmann/ Pearson
5. Programming in Java by Sachin Malhotra, Saurabh Choudhary / Oxford University Press
6. Core Java : Advanced Features - Volume 2 Gary Cornell, Cay S. Horstmann/ Pearson
7. Beginning Java 2 by Ivor Horton; Wrox Publication

Note:

- **Each session mentioned is of 2 hrs of Theory and 2 hours of Lab duration, unless indicated otherwise. T for Theory & L for Lab.**

Session 1 & 2:
Lecture

- Introduction to JVM Architecture
- Java Basics :Overview of Java, Features of Java, Scope of variables
- Object Oriented Concepts
- JDK and its usage (Java Compiler, Java Runtime, Java Debugger, Java doc)
- Working with Data Types: Structure of a Java Class, Importing Packages
- Wrapper classes (Boolean, Double and Integer)
- Operators (Unary, Binary, Arithmetic, Assignment, Compound, Relational, Logical, Equality) andControl Statements (if, if-else, for, while, switch, do-while, break and continue)

Session 3 & 4: (4T + 6L)

Lecture

- Packages
- Arrays
- Understanding of String Class, String Builder Class
- Methods and Encapsulation: Methods, Access Modifiers, Method Overloading, Passing Data, Creating Constructors, Immutable Classes
- Class Inheritance, Abstract Classes, Interface and Implementation classes.
- Understanding Polymorphism: Object vs Reference, Object Casting, Virtual Methods, Method Overriding

Session 5: (2T +6L)

Lecture

- Exception-Handling: Basics, Role of Exceptions, Types
- Using try and catch, Multiple Catch, Nested try (throw, throws, finally)
- Built-in Exceptions, Runtime Exceptions Checked Exceptions, Errors
- Creating own Exception Subclasses
- Enumerations, Auto boxing
- Java API: java.util, java.lang, java.math
- Generics and Collections - Overview

Session 6 & 7

Lecture

- Functional Programming Overview
- Functional Interfaces
- Explore java.util.function package : Predicate, Map, Consumer, Supplier
- Lambda Expressions
- Impact of Functional programming upon Collection Framework

Assignment:

Create an appropriate data structures to store your employee object and use the java.util.package properties.

Session 8 & 9 (4T + 6L)

Lecture

- Introduction to Streams
- Streams vs. Collections
- java.util.stream.Stream API
- Types of Primitive Streams : IntStream,LongStream,DoubleStream & its API
- Different operations on streams : filter, map, reduce, sort, flatMap, anyMatch, count, boxing.
- Overview of Java Date Time API

Session 10 & 11:

Lecture

- Java Concurrency: Using threads in Java, Life cycle of thread
- Advantages and issues
- Thread class, thread groups
- The Runnable interface
- Synchronization

Session 12:

Lecture: Reflection in Java

- Why Java Reflection
- Basic Reflection API for finding out details of the class name, super classes & interfaces.

Session 13: (2 T)

Lecture: Introduction to Node

- Node.js
- Browser JS vs. Node.js
- Node.js REPL

Session 14: (2 T)

Lecture: Introduction to Spring Frame Work

- What is Spring Framework
- Overview of Spring Architecture
- Spring MVC architecture
- Spring Modules Overview