LTI
Let's Solve

Celebrating 20 Years

Java Course Contents

A Larsen & Toubro
Group Company

Objectives of this Course:

- To understand the role of Database in our applications.
- To understand the role of Java technology in modern application development.
- To understand how to develop web applications that are scalable, maintainable.
- To understand the architecture and design of web applications.
- To understand Servlet Technology for generation of Dynamic response to create Simple Web Applications.
- To understand how to separate the application concerns based on functionality.
- To understand the role of Spring framework for managing business processes.
- To understand effective and clean division between controllers, models and view using Spring MVC.
- To understand the Data Persistence using Hibernate Framework.
- To create awareness of Quality Unit Tests, Repeatable Automated Process and Security Concerns.
- To understand the Modern Development techniques using JS Frameworks like jQuery and Angular.

Prerequisites:

- Knowledge of Basic Programming Techniques, Basic Database Functionalities and Basic SDLC.

Course Outline

Day 1 - 3:
Soft Skill Training

Boot Camp Phase – I

Day 4 – 7:
DBMS – DBMS Concepts and SQL

- Introduction to Databases.
- Database Models.
  - Relational Model
- Data Design and Normalization
- Structured Query Language and its categories
  - DDL – DML – DQL – DCL – TCL
- Oracle built in Functions
- SELECT statement varieties with clauses
  - WHERE clause
  - GROUP BY clause
  - HAVING clause
  - ORDER BY clause
- Retrieving data from multiple tables
- Joining the tables – Join variants
  - Equi and Non-Equi Joins
  - Self-Join
  - Cartesian Product
  - Outer Join with modern syntax
- Subqueries
  - Nested Subqueries
  - Co-related Subqueries

DBMS – PL / SQL

- PL/SQL Program Structure.
- Procedural programming using PL/SQL
  - Data Types and Scalar / Anchor Declarations
  - SELECT INTO statement
  - Use of CURSORS in PL/SQL
- Scenario Based Learning
- Modularized programming in PL/SQL
  - Anonymous v/s Named PL/SQL blocks
  - Stored Procedures
  - Functions
  - Triggers
- Exception Handling In PL/SQL

Day 8 – 15:

Core Java – Language Fundamentals

- Different programming paradigms.
- Transformation from OO to PO Paradigm.
- Java Architecture
- A JAVA Program – General Strategy
- Source Code – Byte Code – Path - Class path
- Java – Execution Level Essentials - JDK – JRE – JVM
- Class Loaders and JVM Architecture
- Java versions and high level differences

Core Java – Object Oriented Programming

- A Look at OOP.
- Objects, Classes and Interfaces
- Influential Reuse Mechanisms in OO Languages
  - Composition – Inheritance
- Extending classes - Inheritance Hierarchy
- super and this
- Abstract classes and overriding methods
- Java Modifiers – Access and Non-Access
- Polymorphism and Polymorphic Behavior
  - Static Binding v/s Dynamic Binding
- Creating and Using Packages and Interfaces
- Introduction to Java 8 Lambda expressions
- Introduction to Java 8 Date & Time API

Core Java – Exception Handling

- What is Exception? Java’s exception handling mechanism.
- try – catch block
- finally block
- Exception Propagation and throw keyword
- throw v/s throws
- Exception handling with method overriding
- Custom Exceptions

Core Java – Threads and Concurrency

- Multithreaded programming / Lifecycle of Thread.
- Thread Management
- Multithreaded programming
- Extending Thread class – Implementing Runnable Interface
- Synchronization and Thread Safety

Core Java – Collections and Generics

- Collections framework
  - ArrayList
  - LinkedList
  - HashMap

        - HashTable
        - HashSet
    - Other classes and interfaces in Collection Framework
    - Comparable v/s Comparator
    - Generic Collection
    - Introduction to Java 8 Collection Streams

Core Java – Annotations and Reflections

- Annotations in Java
    - Levels of annotations
- Built-In Annotations – Custom Annotations
- Reflection API
- Modifying the run time behavior of a class

Unit Testing Using JUnit

- What is test and test case
- What is unit testing
- Why we need unit testing
- How to do unit testing by manual
- Problem with manual unit testing
- Introduction about JUnit
- Advantage of JUnit
- Unit test cases using JUnit

Advance Java - Java Database Connectivity

- JDBC Introduction - JDBC architecture - JDBC driver types.
- JDBC API and various interfaces
    - Connection Interface
    - Statement Interface
    - ResultSet Interface
- Steps to establish a JDBC Connection
- Handling parameterized queries using PreparedStatement Interface
- Embedding PL/SQL objects using CallableStatement interface
- Transaction Management in JDBC

Introduction to Agile (Scrum)

Core Java – Mini Project

Interviews by internal SMEs

Boot Camp Phase – II

Day 16 - 20:

Web Technologies - Web Concepts

- Introduction to the Internet and the World Wide Web
- Understanding the concept of Protocols
- Why Web Standards?

Web Technologies – HTML

- Overview of Hypertext Mark-up Language (HTML) and Cascading Style Sheet (CSS)
- Understanding & using HTML
- HTML headings
- HTML Paragraphs
- HTML Line Breaks & Rules
- Font tags
- Hyperlinks
- The Image tag and the src Attribute
- List tags
- Tables
- Forms

Web Technologies - HTML 5

- HTML5 Intro
- HTML5 New Elements
- HTML5 Semantics
- HTML5 Style Guide
- HTML5 Graphics
- HTML5 Canvas
- HTML5 Google Maps
- HTML5 Video
- HTML5 Audio

Web Technologies – Cascading Style Sheet (CSS)

- Introduction to CSS
- Understanding & using CSS
- CSS Syntax
- CSS classes
- CSS IDs
- CSS Margins
- CSS Text Properties
- Font Properties
- CSS links
- CSS Backgrounds
- CSS Border
- Lists

Web Technologies - CSS 3

- CSS3 Introduction

- CSS3 Modules
- Selectors
- Box Model
- Backgrounds and Borders
- Text Effects
- 2D/3D Transformations
- Animations
- Multiple Column Layout
- User Interface
- Borders
  - border-radius
  - box-shadow
  - border-image
- CSS3 Backgrounds
  - background-size
  - background-origin
- CSS3 Text Effects
  - text-shadow
  - word-wrap

Web Technologies – JavaScript

- What is JavaScript?
- Setting up Variables in JavaScript
- Javascript Conditional Statements
- JavaScript Loops
- Arrays
- Javascript Events and Functions
- JavaScript Form Validation

Web Technologies – jQuery

- Introduction To jQuery
- Selection and DOM Traversal
- Working with JavaScript Events

Advanced Web Technologies – Angular 7

- Angular - Introduction
- Understanding Single Page Applications (SPA)
- AngularJS 1.x vs Angular recent versions
- Introduction to TypeScript
  - Role of typescript in Angular
- Developing a simple Angular application
- Writing custom components
- Understanding One-way data binding
- Understanding Two-way data binding
- Angular forms
- Introduction to Angular Routing and DI (Dependency Injection)

Day 21 - 22:

Advance Java - Servlet

- Describe web applications
- Define Model-View-Controller (MVC) architecture
- Describe Java Servlet technology
- Describe the web container behavior
- Develop a simple HTTP Servlet
- Configure and deploy a Servlet
- Design a controller component
- Describe MVC Architecture from Servlet perspective.
- Create an HTML form and discuss how HTML form data is sent in an HTTP request

Advance Java - Servlet Technology in Detail

- Sharing Application Resources Using the Servlet Context
    - Describe the purpose and features of the Servlet Context
- Developing Web Applications Using Session Management
    - Describe the purpose of session management
    - Design a web application that uses session management
    - Develop Servlets using session management
    - Describe the cookies implementation of session management
- Describe the URL-rewriting implementation of session management

Using Filters in Web Applications

- Describe the web container request cycle
- Describe the Filter API
- Develop a filter class
- Configure a filter in the web.xml file
- Understanding authentication mechanisms
    - HTTP Basic authentication
    - HTTP Digest authentication
    - HTTP Client authentication
- Defining authentication mechanisms for web applications

Day 23 - 24:

Developing Java Server Pages

- Describe JSP page technology
- Write JSP code using scripting elements
- Write JSP code using the page directive
- Write JSP code using standard tags
- Effective use of JSP
- Understanding MVC architecture
- Role of Servlet, JSP and JavaBean in MVC architecture
- Introduction to JSTL
- Implementing MVC architecture using Servlet, JSP and JSTL

Day 25 - 27:
Web Services – REST

Overview of Web Services

- Creating Service Oriented Architectures
- Key Components of Web Services and SOA: HTTP, XML
- WS-I Basic Profile

Web Services for Java

- Creating Web Services with Java EE
- Difference between JAX-WS and JAX-RS

Working with JAX-RS

- Writing RESTful web services
- Defining resources and sub-resources
- Consuming data
- Using Application and ApplicationPath
- Resource locators

Working with Return Types

- Simple parameter types
- Specifying return types
- Using JAX-B
  - Standard conversions
  - Custom conversions
- Using Response objects
- Using JSON
- Working with non-standard types: Binary and Files

Handling Errors

- WebApplicationException
- Exception mappers

Writing Clients

Assessment + Mini Project

Day 28 - 30:

DevOps – Overview of Dev-Ops & Git

- Application Lifecycle Management
- Traditional Responsibility Silos
- What is DevOps
- Developers Role
- Testers Role
- Operations Role
- Continuous Integration
- Continuous Deployment
- The Workflows:
- Continuous Integration
- Continuous Build
- Continuous Delivery & Deployment

DevOps – Git

- Git Overview
- Git Installation
- Git Basics
- Git Merge
- Git Branches
- Git Workflows
- Git Hosting

DevOps - Deployment (Maven)

- Maven Installation
- Maven Basics
- Maven Demo
- Maven Eclipse Demo
- Maven lifecycle
- Maven Repositories
- Multi-Module projects
- Maven Summary
- Create a maven project

DevOps – Jenkins

- Jenkins Installation
- Jenkins Basics
- Jenkins Distributed Builds
- Create Jenkins Job

Day 31 – 34:

Introduction to Hibernate Framework

- Object Persistence
  - Object/Relational Paradigms
  - O/R Mismatch
  - Object Relational Mapping (ORM)
  - Java ORM/Persistent Frameworks
- Hibernate Architecture
  - Hibernate Architecture and API
- Introduction to JPA
  - Hibernate and JPA relationship
  - Why JPA?
- Using Hibernate/JPA
  - Using annotations for persistence mapping
  - Configuration of the persistence context
  - Understanding EntityManagerFactory
  - Working with EntityManager for persisting objects
  - Transactions and flushing
- Persistent Classes
  - POJOs
  - JavaBeans - Basic Mapping
  - Class to Table Mappings - Property Mapping
  - Identifiers – Generators
- Working with Persistent Objects
  - Entity Lifecycle
  - Transient State - Persistent State
  - Persistent Object State
  - Detached State
  - Object Identifiers Revisited
- HQL/JPQL
  - Introduction
  - Difference between JPQL and SQL
  - Handling joins and common other cases
  - Named queries
- Introduction to Inheritance mapping
- Conclusion

Day 35 - 41:

Introduction to Spring Framework

- Spring Overview
    - Spring Philosophy
    - Spring Architecture and Modules
- A First Spring Application
    - Dependency Injection (DI)
        - Spring Containers
        - BeanFactory
        - ApplicationContext
        - Bean Definitions & Property Injection
    - Understanding @Component and it's types
    - Using @Autowired for DI
    - Bean Lifecycle

Database support in Spring

- Spring support for DataSource
    - Configuration
    - Benefits of using DataSource
        - Connection Pooling advantage
- Introduction to JdbcTemplate API
    - Performing insert/update/delete
    - Handling select queries
        - Role of RowMapper
- Hibernate/JPA integration with Spring
    - Benefits of using Spring on top of Hibernate/JPA
    - Steps for integration
    - Transaction support in Spring

Spring MVC

- Spring MVC Architecture
- Spring MVC Configuration and Setup
- DispatcherServlet
- Context Configuration
- Annotation Based Controllers
- Understanding the flow
- Commonly used annotations in Spring MVC
- Spring MVC for REST
    - Understanding support for REST WebServices
    - Writing @RestController
    - Handling incoming/outgoing JSON
    - Implementing GET and POST calls

Integrating Hibernate/JPA with Spring MVC

Integration of Angular with Spring MVC

Assessment

Day 42 - 49:
Project Gladiator
Project Evaluation by SMEs

Day 50:
Brain bench Test Preparation & Test