

**Induction Training**

Table of Contents

Duration - 40 Days

**Training Approach:**

**Approach of Delivery**

* Session Duration: 10 am - 6 pm
* Session review with small Question Answer (15 Min) Morning
* Concept Delivery: Approximately 30% of the session time
* Demo on the concepts: 30% of the session time
* Lab Practice: 40 % of the session time

**Approach of Assessment**

* Quiz to be asked during the concept discussion
* Session Review with Question answer every morning
* Post Test after every module viz. Core Java, Data Modeling, Azure, Dev Ops, etc.

**Approach for Hands-on**

* Labs/Do it during the sessions
* Extra project/practical/exercises every evening approximately 2-3 Hours Daily
* Project/Case study after modules and at the end of the training

**Criteria for Assessment**

* Technical Knowledge
* Analytical Ability
* Punctuality
* Responsiveness
* Corporate Etiquette
* Team Work
* Communication

**Table of Contents**

|  |  |
| --- | --- |
| **Training Topics** | **Duration**  **(Days)** |
| OOAD UML, RDBMS, Core Java (Basics) | 4 |
| Core Java (Advanced), Hibernate, Java EE, Spring | 8 |
| Oracle 12c SQL & PL/SQL  (Basic/Intermediate concepts) | 2 |
| Unix & Shell Scripting | 1 |
| HTML5 & CSS3 | 1 |
| JavaScript (ECMAScript 2015) | 3 |
| NodeJS | 1 |
| ReactJS | 2 |
| Microsoft Azure (PaaS) | 3 |
| Artificial Intelligence + Machine Learning | 2 |
| Agile SCRUM + DevOps/CICD | 5 |
| Office Etiquettes | 1 |
| Capital Markets | 4 |
| ITIL V3 Foundation | 1 |
| Software Testing Basics & Testing Tools | 2 |

**OOAD UML**

**OOAD**

* Object Oriented Analysis and Design
  + Introduction to Object Oriented Analysis & Design (OOAD)
  + Object Oriented Methodology
  + Object Oriented Analysis & its primary tasks
  + Object oriented Design
  + Object oriented Modeling & its benefits
* Problem Statement
* Problem Specifications
* Identifying Object Classes
* Classes and Objects
* Prepare a Data Dictionary
* Identifying Association
* Identifying Attributes
* Refining with Inheritance
* Testing Access Path
* Iterating Object Modeling

**Fundamentals of UML**

* Introduction to UML
* Goals of UML
* Objects, Classes and the UML

**UML Diagrams**

* Class and Object Diagrams
* Use Case
* Interaction Diagrams
* Activity Diagram
* Associations (Composition and Aggregation)
* Relationships and Multiplicity

**RDBMS**

**Introduction to RDBMS**

* Overview of Database Models
* ER Diagram
* Normalization

**Core Java (Basic)**

**Introduction to Java**

* Features of Java
* JDK, JRE and JVM
* OOPs
* Class, Object, Attribute, method
* Access Modifiers – Private, Public
* Constructors
* Data types and Operators
* Arrays
* Control Flow statements

**Class Design**

* Class and Objects
* Instance Variables
* Methods and Constructors

**Polymorphism**

* Method Overloading
* Constructor Overloading
* Use of “this” keyword
* Variable Argument Method (Varargs) in Java
* Varargs versus Overloading

**Inheritance**

* Inheritance Basics
* Use of super keyword
* Overriding
* Runtime Polymorphism

**Advanced Class Features**

* Static variables, Static block and methods
* Static Import

**Abstraction**

* Abstract classes and methods
* Final classes and methods
* Interfaces

**Encapsulation**

* Packages and Access Specifiers

**Core Java (Advance)**

**Exception Handling**

* Exception and Errors
* Checked and Unchecked Exception
* Handling exceptions using try, catch and finally
* Java7 multi catch
* Use of throw and throws
* Automatic Resource Management
* Create Custom Exceptions

**java.lang**

* Wrapper classes
* Autoboxing and Unboxing
* Object class(equals and hashcode)
* Understanding toString method
* String, String Buffer and String Builder

**java.util**

* Generics
* Collections Framework
* Legacy classes Vector and Hashtable
* List, Set
* Map
* Iterator, ListIterator and Enumeration
* Comparator vs Comparable

**Lambda Expressions**

* Why Java 8?
* Behavior Parameterization
* What is a lambda?
* Functional interfaces: where to use lambda expressions?
* Method references: first-class functions in Java 8

**Streams**

* Collection Processing
* Stream operations and patterns
* Stream Optimization

**Collectors**

* Grouping and partitioning
* Collection Operations
* Arithmetic collectors
* Advanced Queries

**JDBC**

* JDBC overview
* Types of drivers
* java.sql interfaces Driver, Connection, Statement
* Loading a driver and establishing a connection using DriverManager
* Perform CRUD operations using JDBC interfaces
* Prepared Statement for precompiled queries

**Hibernate**

**Introduction to ORM framework**

* JDBC Vs Hibernate
* Features of Hibernate

**Hibernate Architecture Overview**

* Configuration and Session Factory
* Session
* Transaction
* Environment Setup
* Getting started with Hibernate quickly

**Database Connection and Schema Generation**

* Mapping JavaBeans to tables
* Mapping properties to columns
* Building a Configuration Programmatically / Using XML configuration file
* Creating a HBM document from the java bean class
* Creating a java bean class from a HBM document

**Persisting objects using Hibernate**

**CRUD Examples**

* Persisting data
* Loading data into an object
* get Vs load
* Deleting, updating & finding objects

**Lifecycle**

* Transient State
* Persistent state
* Detached state

**Setting up all types of Mapping**

* Mapping composite keys
* Mapping a java class
* Mapping Collections
* Mapping Association
* Component Mapping
* Inheritance Mapping

**Hibernate Annotations**

* Using annotation for all kinds of mapping
* Use of @Entity, @Embedded and other annotations
* Mapping a simple class, Association, Inheritance

**Hibernate Query Language**

* Using Query interface for single/multi select statements
* Named and positional parameters
* Named and Native Queries
* Using SQL Queries
* Working with HQL Joins
* Using Criteria for select statements

**JAVA EE**

**Java EE Introduction**

* Java EE Overview
* MVC

**Servlet and JSP**

* Web Application Basics
* Servlets Basics
* Servlet API
* HTML Forms
* HTTP: request- response ,GET, POST
* Servlet life cycle
* Parameters Vs attributes
* Overview of jsp
* Introduction to jstl

**Spring**

**Spring Introduction**

* Shortcomings of Java EE and the Need for Loose Coupling
* Managing Beans, The Spring Container, Inversion of Control
* The Factory Pattern
* Configuration Metadata - XML, @Component, Auto-Detecting Beans
* Dependencies and Dependency Injection (DI) with the BeanFactory
* Setter Injection

**Dependency Injection**

* Using the Application Context
* Constructor Injection
* PropertyEditors
* Factory Methods
* Crucial Namespaces ‘p’ and ’c’
* Configuring Collections
* Bean Definition Inheritance and Collection Merging
* Expression Languages, SpEL

**The Spring Container and API**

* The Spring Managed Bean Lifecycle
* Key interfaces, Annotations and BeanPostprocessors
* Autowiring Dependencies

**Other Metadata Configurations**

* Annotation Configuration @Autowired, @Required, @Resource
* @Component
* @Value and @Qualifier
* Life Cycle Annotations
* Java Configuration, @Configuration, XML free configuration
* The AnnotationConfigApplicationContext

**Spring and Persistence**

* Spring and JDBC
* Spring and ORM

**Develop Web Applications using the Spring Framework - Spring MVC**

* The WebApplicationContext and the ContextLoaderListener
* Model View Controller
* Front Controller Pattern
* DispatcherServlet Configuration
* Controllers, RequestMapping
* Working with Forms
* Getting at the Request, @RequestParam
* ModelAndView

**Advanced techniques**

* Spring form tags and Model Binding, @ModelAttribute

**Oracle 12c SQL & PL/SQL**

**SQL**

* Introduction
* Environment Setup
* Getting Started
* DDL Statements
* Create Table
* Alter Table
* Drop Table
* Truncate Table
* Constraints
* Not Null Constraint
* Unique Constraint
* Primary Key Constraint
* Check Constraint
* Foreign Key Constraint
* Managing Constraints
* Insert Statement
* Update Statement

**PL/SQL**

* Overview
* Environment Setup
* Basic Syntax
* Data Types
* Variables
* Constants and Literals
* Operators
* Conditions
* Loops
* Strings
* Arrays
* Procedures

**Unix & Shell Scripting**

* Introduction to Shell
* Types of Shell
* Basic syntax
* Conventions
* Variables
* Decision making
* Looping
* Strings
* Functions
* File Handling
* Advanced Techniques
* Working with AWK

**HTML5/CSS3**

* History of HMTL5
* The HMTL5 vision
* WHATWG and W3C specifications
* What is part of HMTL5?
* HMTL5 Roadmap
* Detecting Features using Modernizr
* HMTL5Page Structure
* HMTL5 DOCTYPE
* HMTL5Markup Structural elements
* Semantic Elements
* HMTL5Form Elements
* Overview of the HTML5 APIs
* Introduction to CSS3
* CSS 3 Fonts & Selectors
* CSS3 Animations
* CSS3 Media Queries

**JavaScript (ECMAScript 2015)**

**Introduction to JavaScript**

* What JavaScript Is?
* What JavaScript Is Not?
* What JavaScript Is Used For?
* JavaScript and Its Place in the Web.

**JavaScript Building Blocks: Data Types, Literals, and Variables**

* Data Types
* Variables
* let, const and Block Scoping

**Operators**

* About JavaScript Operators and Expressions
* Types of Operators
* Data Type Conversion
* Wrapper Objects (String, Number, Function, Boolean)

**Under Certain Conditions (Just Overview, if required)**

* Control Structures, Blocks, and Compound Statements
* Conditionals
* Loops

**Functions**

* What Is a Function?
* Function Improvements

**Functions Continued…**

* Function Overloading Assignment
* Closures
* Object Literal Syntax
* Module Pattern
* Revealing Module Pattern
* Destructuring Objects
* Destructuring Functions - Multiple returns and named defaults

**String Type**

* Template Strings (Literal)
* Creating HTML fragments with Template Literals
* Tagged Template Literals
* Sanitizing User Data with Tagged Templates
* New String Methods

**Array & Collections**

* Creating Array
* Array Shorthand
* Spread Operator & Rest Revisited
* New Array Methods
* The for of loop
* Destructuring Arrays
* Swapping Variables with Destructuring
* Array Extensions
* ArrayBuffers and Typed Arrays
* Map and WeakMap
* Set and WeakSet

**Objects**

* What Are Objects?
* Object Literal Syntax
* Classes and User-Defined Functions
* Manipulating Objects
* ES6 Classes
* Extending Objects with Prototypes
* ES6 Modules
* New Data Types and Extensions

**Iterators & Generators**

* Iterators
* Generators

**Async Await Flow**

* Understand Promises
* Async Await - Native Promises
* Async Await - Custom Promises
* All About Async + Await
* Async + Await Error Handling
* Waiting on Multiple Promises

**NodeJS**

* History of Node.js
* Non-Blocking, I/O
* Node.js Benefits
* Node.js Place in the Tech Stack
* Evolution of Node.js Frameworks
* Installing Node.js
* Nodejs Building Blocks
* Core built-in modules
* Creating Node Modules
* Modularizing JavaScirpt code
* Using require() to modularize application code
* Using npm for third-party modules
* Node.js and the web
  + Building a web server
  + Handling web requests
  + Returning HTML
* Building simple web applications using Express.js

**ReactJS**

**Introduction to React**

* Why React?
* Advantages of React
* React Development Style

**Get Started with React**

* Setting up a React App
* Running a React App
* React Initialization Process
* React Virtual DOM
* React Ecosystem

**Webpack Configuration**

* What is Webpack?
* Configuring Webpack
* Understanding Webpack Loaders
* Using Webpack plugins
* Hot Reloading
* Understanding Babel
* Adding Scripts to package.json

**JSX and React Components**

* Introduction to JSX
* Render Elements
* Understanding Component
* Creating a Component
* Container vs. Presentation Components
* Props and State
* Component Lifecycle

**Data Flow**

* Introduction to One Way Data Flow in React
* Props and State
* Passing Data to Child Component
* Handling State of Component

**Event and Actions**

* Events in React
* Data Binding
* Events and Actions
* Handling Events
* Communication Among Components

**React Forms and Controls**

* Introduction to React Forms
* Creating Forms
* Forms Validations

**Ajax Calls and Error Handling**

**Microsoft Azure (PAAS)**

* Cloud Computing Overview
* Introduction to Cloud Computing
* Overview of Traditional DC and why cloud?
* Characteristics of Cloud
* Cloud Service Models
* Responsibility Model
* Microsoft Azure Fundamentals
* Azure Projects and Subscriptions
* Azure Resource Manager
* Billing model in Azure
* Managing Azure using GUI Portal
* Compute Services with images, instance types, internal root disks, firewall groups etc
* Managing Compute Service using Azure Portal
* Azure Networking Overview
* Introduction to Azure virtual networks and various interconnects
* Storage Services with Block level and Object level storage
* Backup/DR & HA Solutions
* Database as a Service with Relational and non-relational DB offerings
* App Services for ops less teams
* Planning App deployment in Azure App Services
* Implementing and managing web apps in Azure

**Artificial Intelligence + Machine Learning**

* AI Overview
* AI Intelligent Systems
* AI Research Areas
* AI Agents & Environments
* AI Popular Search Algorithms
* AI NLP
* AI Issues
* What is ML
* What is need of ML
* How Machine Learning Works
* Examples
* Application of Machine Learning Algorithms
* Types of Learning
* Example of Machine Learning with Python

**Agile/SCRUM + DevOps/CICD**

* Agile Project Lifecycle
  + Essence of Agile Simulation
  + Agile Values and Principles
  + Minimum Viable Product
  + Release Planning
* Scrum Framework
  + Product Backlog
  + Sprint (sprint durations and its consequences)
  + User Stories Authoring
  + Different Agile Estimations Techniques
  + Sprint Backlog & Sprint Planning
  + Daily Standup Meeting with a simulation
  + Burn-down/Burn-up Charts & Project Reporting
  + Sprint Review & Retrospective
  + Scrum Master roles and responsibilities
  + Product Owner roles and responsibilities
  + Cross Functional Team’s roles and responsibilities
* DevOps Fundamentals
* DevOps Terminologies
* What is Continuous Integration
* What is Continuous Deployment
* What is Continuous Delivery
* DevOps Toolset
* Getting started with Version Control System and Operations
* Configuration Management with Ansible
* Ansible setup and configuration
* Ansible Terminologies
* Ansible Modules
* Ansible Ad-hoc requests
* Introduction to YAML
* Automation with Ansible Playbooks
* Containerization with Docker
* Docker setup and configuration
* Difference between VM and Containers
* Docker Container and Operations
* Docker Image and Operations
* Docker Network and Operations
* Docker Volume and Operations
* Jenkins for building CI/CD pipelines
* What is CI
* Jenkins Setup and configuration
* What are pipelines
* Automated Pipelines for various tools with Jenkins
* Configuring End to End Deployment Pipeline in Jenkins by integrating Jenkins, GIT, Ansible and Docker

**Office Etiquettes**

**First Impression and Image Management**

* Understand why first impressions matter
* Personal Grooming and hygiene
* Image makers & breakers
* ABC of Image management
* How to dress to make an impact
* Effective Body Language
* Perfect handshake

**Business Etiquette**

* Understand Business Protocols
* Professional Manners & Workplace courtesy
* Meeting Etiquette
* Dining Etiquette
* Cubicle Etiquette

**Email Etiquette**

* After the completion of the session, the participants will be able to
* Understand how to structure clear and concise email communication to ensure the main points are not lost
* Avoid the most common mistakes people make in business writing
* Importance of appropriate subject line, salutation and body of the email
* Describe what should not be discussed via email
* Determine who should and should not receive copies of email
* Know what to do when email misfire and reach the wrong people

**Telephone Etiquette**

* Use the correct protocols and techniques for speaking on the telephone & conference calls
* Demonstrate the best way to start the call
* Explain what the right way is to handle customer calls
* Practice how to end the call in a positive manner
* Answer telephone promptly and professionally

**Interpersonal Communication**

* Gain Effective interpersonal skills with Subordinates and Peer colleagues
* Practice the art of effective Listening (Hearing Vs Listening)
* Right Questioning
* State what non-verbal communication is and how it can enhance interpersonal relationships

**Public Speaking & Confidence Building**

* Getting deeper understanding of the basic concepts of Public speaking
* How to communicate effectively using verbal and nonverbal techniques
* How to speak with passion and energy
* How to handle audience
* How to remove stage fear

**Assertive Communication**

* Participants will be able to
* State the difference between Assertive & Aggressive Communication
* Identify ways of sharing one’s opinions constructively
* How seeing the other side without giving up can improve skills in influencing other people
* Practice Assertive Skills in difficult situations

**Capital Markets**

**Introduction to financial markets**

**Purpose of Financial Markets**

* Economic life cycle
* Implication for different entities
* Flow of money between the entities
* Money Supply
* Inflation & Interest Rates
* Business Cycles

**Overview of Financial Markets**

**Concept of Asset Class**

* Equity
* Debt
* Forex
* Commodities

**Use of each Asset Class (in brief)**

**Concept of Primary Market and Secondary Market**

**Difference between basic asset class and derived asset class**

**Primary market**

* Methods of issue of securities
* Players
* Regulations

**Secondary market**

* Players
* Securities
* Methods of trading
* Regulations

**Types of financial markets**

**OTC**

* Securities traded
* Rationale
* Disadvantages
* Regulations

**Exchange traded**

* Securities
* Regulations
* Advantages & disadvantages

**Money Markets Instruments**

* Introduction to Money Markets
* Money Markets Instruments
* Money Markets Participants
* Repurchase (Repo and Reverse Repo) transactions
* Overnight Interbank Loan Markets
* LIBOR (London Interbank Offered Rate)
* Certificate of Deposit
* Commercial Paper

**Introduction to Equity**

**Equity securities**

* What are equity shares?
* Risk and reward
* Capital structure – when would a company issue equity securities versus debt securities?

**Overview of Foreign Exchange Markets**

* Forex market terminologies
* Structure of the Forex market
* Forward Trades
* Hedging/Speculation/Arbitrage
* Forex Risks

**Introduction to Derivatives (5 hours)**

**Derivatives Overview**

* Need for Derivatives
* Products
* Participants and Functions
* Exchange Traded vs. OTC Derivatives

**Market Index**

* Index Construction
* Desirable attributes of an index
* Application of Index

**Forward Contracts**

* Introduction to Forward Contracts
* Limitations of forward contracts

**Futures Contracts**

* Introduction to Futures Contracts
* Futures Terminology
* Standardized Exchange traded contracts

**Introduction to Derivatives**

* Options Contracts
* Introduction to Option Contracts
* Calls and Puts
* Options Terminology
* Options Characteristics

**Commodities**

**What is commodity market**

* Market Players
* Basic Features

**Life Cycle of a Trade**

* Trading & settlement
* Market place
* Participants
* Regulators
* Custodians
* Types of transactions
* Trade enrichment
* Trade validation
* Settlement Instructions
* Corporate actions
* Reconciliations

**ITIL V3 Foundation**

* Service Management as a Practice
* Service Strategy
* Service Design
* Service Transition
* Service Operation
* Incident Management, Problem Management and Change Management
* Introduction to CAB (Change advisory Board)
* Types of Changes
* Risk Management
* Continual Service Improvement

**Software Testing Basics + Testing Tools**

* Overview of TDD
* Overview of BDD
* Different types of Testing
* Testing Automation
* Different Testing tools available
* Overview of Selenium
* Overview of Cucumber
* Overview of JMeter