**Day 1 & 2**

**Campus To Corporate**

**Day 3**

**Module 1**

**RDBMS**

* RDBMS Introduction
* Data
* Database
* Database Models
* Client Server Architecture
* SQL Commands
* SQL and its categorization – DDL, DML, DQL, DCL, TCL

**Day 4**

**Module 2**

**OOP/ Basics of Java**

* Introduction to Java Programming Language.
* JVM Architecture
* Data Types and Variables in Java
* Operators in Java
* Decision Making Statements
* Working with Arrays in Java
* Introduction to OOPS Concepts
* Features of OOPs
* Working with Classes and Objects
* Inheritance in Java
* Abstract Class and Interface in Java
* Encapsulation in Java

**Day 5**

**Exception Handling**

* What is Exception and its types
* How to handle exception?
* Multiple catch blocks
* Finally block
* Difference between throw and throws
* Exception Handling with Method Overriding
* Custom Exception Handling

**Day 6**

**MultiThreading/JDBC**

* What is multithreading and advantages.
* What is Thread?
* Life Cycle of a Thread.
* Thread Classes and Methods
* “Runnable” Interface
* Creating Thread
* Working with multiple Threads
* Java Synchronization
* Interthread Communication
* Dead Lock
* Java 8 Concurrency using Executors
* Introduction to JDBC API
* JDBC Drivers
* JDBC Architecture

**Day 7**

**Collections Framework**

* What are Data Structures?
* Collection Overview
* Lambda Expression and Functional Interface
* Interface in Collection Framework
* List Interface and Its implementation
* Iterator, ListIterator and SliptIterator
* Working with Maps Interface
* Sets Interface and Its class Implementation

**Day 8**

**JDBC Continued/Java 8/11/17 features**

* JDBC Life Cycle
* CRUD Operations using Statements, PreparedStatements and CallableStatements
* ResultSet, Scrollable and Updateable ResultSet types
* Handling SQL Exception
* Primary Key Generators
* Batch Operations
* Connection Pooling
* Java 8/11/17 Features
* Method References
* Default Methods
* Try-with-Resources
* Private Interface Methods
* Local Variable Type in Lambda Expressions
* Sealed Classes and Constraints

**Day 9**

**Module 3**

**Servlet Fundamentals/JSP/Session object APIs**

* Servlet & HTTP Servlet Theory
* Lifecycle of Servlet
* Servlet Request and Response
* Servlet Request Dispatcher
* Servlet Config
* Session Tracking
* Filters
* JSP Scripting Elements
* JSP Implicit Objects
* JSP Page Directive
* JSP Taglib

**Day 10**

**Unit Testing Framework/JSTL**

* JSTL Core Tag
* JSTL Function Tag and Formatting Tag
* Introduction to JUnit
* Junit test : @Test, assertEquals(), assertNotEquals()
* @BeforeAll , AfterALL Annotations
* The assertArrayEquals() method

Doubt Clearing Session Can be planned at mutual agreement of day & date

Day - 11

Module 4

**JPA**

* Review of ORM problem
* Examine industry attempts to solve problem
* Resolve the relationship between Hibernate and JPA
* What is JPA?
* Overview of key concepts and terms
* Persistence units, Entities, and EntityManagers
* EntityManagerFactories, JPA QL, and the JPA Criteria queries
* Set up the environment to work with JPA and a persistence provider
* Create a persistence.xml file
* Define a persistence provider
* Define a persistence unit using JDBC connection strings and pooling
* Define a persistence unit using data sets

Day – 12

* Utilize Hibernate as a persistence provider
* Utilize EclipseLink as a persistence provider
* Choose the best configuration style
* Review of Java annotations
* Review of Dependency Injection
* Creating an Entity
* Associating Entity with EntityManager
* Performing CRUD with an EntityManager
* Understanding Entity states
* Performing basic Primary Key querying

**Spring – Day 13**

**Module 5**

* Spring Framework definition
* Spring Framework design principals
* Spring interfaces
* Bean Factory
* Application Context
* Factory methods
* Dependency Injection advantages
* Setter Injection
* Constructor Injection
* Bean Life Cycle
* Inner Beans
* References to other beans
* Scopes in Spring
* Autowiring advantages and modes
* Spring With Collections
* Spring AOP Advantages
* AOP Configuration
* AOP Terms
* AOP Types of Advices

**Day 14**

**Spring JDBC/ Spring MVC**

* JDBC Templates
* Row Mapper
* BeanProperty Row Mapper
* Bean Row Mapper
* Resultset Extractor
* MVC Architecture in Spring
* About Dispatcher Servlet
* Types of Handler Mappings
* Types of Controller
* Types of View Resolver
* Spring MVC Annotations
* Spring MVC with CRUD operations

**Day 15**

**Spring/ WebServices**

* Introduction to WebServices
* Postman Overview
* Resources and Collections URIS
* HTTP Methods : GET, POST, DELETE, PUT
* HTTP Headers
* Create new Spring Boot Project with Spring Tool Suite
* Create RESTful Web Service EndPoints
* Handle HTTP Get and Post Requests
* Handle HTTP Put and Delete Requests
* Exception Handling
* Dependency Injection
* Adding Security

**Day 16**

**Module 6**

**SpringBoot**

What is Spring Boot and What it Does?

* High level Spring Boot features
* Dependency management using Spring Boot starters
* Creating Spring Boot Application
* SpringBootApplication / CommandLineRunner / ApplicationRunner
* Exploring Autoconfiguration
* Exploring Embedded Containers Configuring your Boot containers
* The application properties and application .yml files
* Create a Simple Spring Boot Application using Spring initializer
* REST Overview (Characteristics/Capabilities, URI Templates, REST vs SOAP
* REST and Spring MVC
* Spring support for REST
* @RequestMapping/@PathVariable, @RequestBody, @ResponseBody, HTTP Method

conversion

* URI Templates and @PathVariable
* Writing RESTFul Controllers/@RestController

**Day 17**

**SpringBoot /Security**

* Basic Auto-configuration – DataSource and Pooling
* Configuration Properties
* Spring Boot with JDBC
* Quick Introduction to ORM with JPA
* Benefits of using Spring JPA
* JPA configuration in Spring
* Overview and Capabilities
* Grabbing / @Grab
* Spring Security Overview(Web)
* spring-boot-starter-security-Auto-configuration and Customization

**Day 18**

**Module 7**

**Angular**

* Introduction to Angular
* Why use it?
* What about Angular JS?
* Why TypeScript?
* Introduction to Angular CLI
* Modules and ngModule
* Components and component libraries
* Introduction to pipes, directives, and services
* What are components?
* Example of Angular component
* Example Hello World
* Template and expressions
* Passing data to a component with @Input()
* One-way data bindings
* Expressions
* Two-way data bindings with ngModel
* Template syntax
* Template reference variables
* What are directives for?
* Different kinds of directives
* ngFor
* ngIf
* ngSwitch, ngClass and ngStyle
* How to create a custom directive?
* What are pipes for?
* Examples of common Angular pipes
* Pure and Impure pipes
* How to create our own pipes?

**Day 19**

**Angular**

* How Angular bootstraps an application
* Importance of using angular CLI to keep the project structure updated
* Angular builds and environments
* Difference between components and services
* Dependency injection in Angular
* Example of the component router in action
* Injector Tree
* How to create a service?
* Introduction to RxJS and Reactive Programming
* What is an Observable?
* Writing our own Observable
* RxJs operators
* Examples of RxJs operators: max, merge, filter
* Introduction to Subjects
* How to use the HTTP client?
* HTTP Interceptor
* Async pipe
* What is the component router?
* Example of the component router in action

**Day 20**

**Angular/Unit Testing**

* Child routes and route parameters
* Guards for authentication and user rights
* Lazy-loading
* Form Handling in Angular
* Form Validations and Submission
* Custom form Validator
* Component Communication
* Introduction to the Component Lifecycle
* Lifecycle hook
* Single Page Application
* How Test Angular Application
* Angular Test Bed
* Mocking and Stubbing with Angular
* End to End Test with Protractor

**Day 21**

**Module 7**

**NoSQL DB (Couchbase)**

* Introduction to NoSQL
* Discuss the modern application challenges
* Describe Big Data
* Describe the common strategies for handling Big Data
* Explain CAP Theorem
* Differentiate Hadoop and NoSQL
* Explain different types of NoSQL
* Describe Polyglot Programming
* Introduction to Couchbase
* Describe the history of Couchbase server
* Describe Couchbase Data Platform
* Explain how data is stored in Couchbase server
* Describe Couchbase server architecture [Single Data Node]
* Describe the anatomy of a Couchbase application
* Explain Multi Dimensional Scaling
* Describe Tools to Integrate with Couchbase
* Explain Couchbase SDKs

**Day 22**

**NoSQL DB(Couchbase)**

* N1QL Essentials
* Select documents and limiting results
* Explain Aliasing, concatenating, and selecting by keys
* Create indexes and filter queries
* Describe Querying ranges, ordering results, and explaining queries
* Describe Aggregating, distinct values, and filtering with wildcards
* Select for missing attributes, and group results
* Index and select based on values in JSON arrays
* Use functions in queries
* Join documents

**Doubt Clearing Session**