



### COVERAGE

- Introduction to Spring:
- Spring Core (Basic Concepts)
- Spring Core (Advanced Concepts)
- Spring Core (3.0 Annotations)
- Spring AOP
- Spring Transaction
- Spring JDBC
- Spring MVC
- Spring ORM
- Introduction To Hibernate
- Hibernate Annotations vs JPA Annotations:
- Hibernate Mappings
- Hibernate Query Languages and Transactions and Caching
- Spring Hibernate Integrations
- Hibernate uses in Project:

### SYLLABUS IN DETAILS

---

#### Intended Audience:

- This training is designed for developers/fresher's interested in learning Spring framework

#### Prerequisites:

- Basic understanding of JDBC
- Knowledge on Servlet/JSP
- Basic knowledge on any application server(Tomcat)

#### Introduction to Spring:

- What is Spring,
- How Spring fits into the Enterprise world,
- Spring Modules

#### Spring Core (Basic Concepts):

- What is a Core Container
- Introduction to IOC
- Types of DI
- Setter DI vs Constructor DI
- Resolving Constructor Confusion
- Collection DI

- Bean Inheritance
- Collection Merging
- Inner Beans
- Using IDRef
- Bean Aliasing
- Bean Scopes
- Inner Beans
- Null String
- Bean Auto wiring
- Nested Bean Factories

#### Spring Core (Advanced Concepts):

- P – Namespace
- Dependency Check
- Depends On
- Factory Beans
- Static Factory Method
- Instance Factory Method
- Aware Interfaces
- Bean Lifecycle
- Method Replacement
- Lookup Method Injection

- Property Editors
- Bean Factory Postprocessor

#### Spring Core (3.0 Annotations):

- Spring VS Java Config Project
- annotations
- @Required
- @Autowired
- @PostConstruct
- @PreDestroy
- @Qualifier
- @Resource
- @Component
- @Service
- @Controller
- @Named
- Various Annotation based Post Processors
- (AutowiredAnnotationBeanPostProcessor,
- CommonAnnotationBeanPostProcessor,
- PersistenceAnnotationBeanPostProcessor,
- RequiredAnnotationBeanPostProcessor)

#### Spring AOP :

- AOP Concepts
- Programmatic VS Declarative AOP
- Programmatic AOP
- Types of Advices
- Types of Pointcuts
- Working with proxies
- Declarative AOP
- Using AOP 2.0 Config element
- OGNL Expressions
- Aspect J AOP

#### Spring Transaction :

- Aop 2.0 Configuration driven Transaction Management
- Aspect J annotation based Transaction Management

#### Spring JDBC:

- What is DAO pattern?
- Ways to implement Spring DAO
- Choosing an approach for JDBC database access

- JDBC Template
- Executing Statements
- Running Queries
- SQL Parameters

#### Spring MVC :

- Spring 3.0 features
- Introduction to Spring MVC
- Handler Mapping
- Controllers
- Validations
- Handler Interceptors
- Views
- Form tags

#### Spring ORM :

- Integrating with Hibernate
- Creating and Injecting Hibernate Session Factory
- Managing Transaction

#### Introduction To Hibernate :

- Need for Hibernate
- Hibernate and ORM (Object-Relation Mapping)
- POJOs (Plain Old Java Objects) and the Data Layer
- Hibernate Over Entity Beans
- Understanding Hibernate Architecture
- Configuration
- SessionFactory
- Session
- Query
- Criteria
- Hibernate Configuration
- Hibernate Mappings
- Persistent Classes
- Working with Hibernate to perform basic CRUD Operations
- Configuring Mappings Using Annotations.

#### Hibernate Annotations vs JPA Annotations:

- Getting started with Hibernate 3.X
- Using JPA annotations and XML configuration

- Hibernate Domain models
- Hibernate Persistence Context
- Mapping persistence Classes
- Hibernate Inheritance

#### Hibernate Mappings:

- Component Mapping
- Inheritance Mappings
- Table Per Class Hierarchy
- Table Per Sub Class
- Table Per Concrete Class
- Association Mappings
- One -to-One
- One -to-Many
- Many -to-One
- Many -to-Many

#### Hibernate Query Languages and Transactions and Caching:

- Using queries: HQL, criteria API, native (SQL)
- Hibernate Transaction
- Mixing Hibernate and JDBC
- Hibernate second level caching

#### Spring Hibernate Integrations:

- Spring Hibernate Integration
- Data source creation
- Hibernate DAO implementation using Spring Framework
- Spring – JDBC Transaction
- Spring AOP – Integration

#### Spring Hibernate Integrations:

- Spring Hibernate Integration
- Data source creation
- Hibernate DAO implementation using Spring Framework
- Spring – JDBC Transaction
- Spring AOP – Integration

#### Hibernate uses in Project:

- Design Web Application using hibernate
- Hibernate in web application(case studies)
- Project Guidance