FULL STACK JAVA

DATABASE Covered:

1. Oracle
2. MySQL
3. MongoDB

ID's Covered:

1. Eclipse
2. Spring Tool Suite (STS)
3. IntelliJ Idea
4. Netbeans

SERVER Covered:

1. Tomcat
2. Weblogic
3. JBOSS and Wildfly
4. Glassfish

Real Time Tools Covered:

1. Maven
2. Log4j

HIBERNATE

1. Introduction
    1. Enterprise
    2. Enterprise Application
    3. Enterprise Application Layer
        1. User Interface Layer
        2. Business Processing Layer
        3. Data Storage and Access Layer
    4. Data Persistency
    5. Data Persistency through Serialization and Deserialization
    6. Data Persistency through JDBC
    7. Data Persistency through ORM
        1. Paradigm Mismatches
            1. Granularity Mismatch
            2. Subtypes Mismatch
            3. Associations Mismatch

        4. Identity Mismatch
    2. EJBs Vs Hibernate
    3. JPA Vs Hibernate
8. Hibernate History
9. Hibernate Features
10. Hibernate Arch.

2. Steps to Prepare Hibernate Application
    1. Persistence Class / POJO class
    2. Mapping File
    3. Hibernate Configuration File
    4. Client Application

3. Hibernate Applications
    1. Hibernate Application with Main Class as Client.
    2. Hibernate Application with GUI Application as Client.
    3. Hibernate Application with Servlet as Client.
    4. Hibernate Application with JSP Page as Client.
    5. Hibernate Application with Struts Application as Client.
    6. Hibernate Application with MYSQL DB
    7. Hibernate Application with Multiple DBs [Oracle DB and MySQL DB]
    8. Hibernate Basic Annotations [Without Mapping File]
    9. Hibernate Application without Configuration File
    10. Hibernate Application with Composite Keys.
4. Hibernate Persistence Object Lifecycle
    1. Transient State
    2. Persistent State
    3. Detached State
    4. Removed State
5. Hibernate Tools
    1. Schema Export
    2. Schema Update
    3. Code Generation
6. Primary Key Generation Algorithms [XMl and Annotations]
    1. Assign
    2. Increment
    3. Sequence
    4. Identity
    5. Hilo
    6. Seq-Hilo
    7. Native
    8. UUID
    9. Foreign
    10. GUID

- 11. Select

7. Transaction Management
    1. ACID Properties
        1. Atomicity
        2. Consistency
        3. Isolation
        4. Durability
    2. Transaction Management in JDBC
        1. Atomicity Achievement in JDBC
        2. Isolation Problems
    3. Transaction Management in Hibernate

8. Hibernate Query Language [HQL]
    1. HQL Elements 1. Clauses
        1. 'From' Clause
        4. 'Order by' Clause
        2. 'Select' Clause
        5. 'Group by' Clause
        3. 'Where' Clause
        6. 'Having' Clause
    2. Aggregate Functions
        1. count(-)
        4. max(-)
        2. sum(-)
        5. avg(-)
        3. min(-)
    3. Generic Expressions
        1. Arithmetic Operators in Generic Expressions
        2. Comparison Operations in Generic Expressions
        3. Scalar Functions in Generic Expressions
            1. In
            4. is null
            2. Between
            5. is not null
            3. Like
    4. Associations and Joins
    5. Parameters
        1. Positional parameters
        2. Names Parameters
    6. Subqueries
        1. Pagination
        2. HQL with Updations

9. Native SQL
    1. Scalar SQL Queries
    2. Stored Procedures and Functions
10. Criteria API
11. Hibernate Filters
12. Hibernate Mappings
    1. Basic 'OR' Mapping
    2. Component Mapping
    3. Inheritance Mapping
        1. Table per Class Hierarchy
        2. Table per Subclass
        3. Table per Concrete Class
    4. Associations Mapping
        1. One-To-One Association
        2. One-To-Many Association
        3. Many-To-One Association
        4. Many-To-Many Association

13. Connection Pooling
    1. Inbuilt Connection Pooling Support in Hibernate.
    2. Third Party Connection Pooling Mechanisms C3P0, Proxool, DBCP.....
    3. Connection Pooling through Weblogic Server JNDI.
14. Cache Mechanisms
    1. I level Cache
    2. II Level Cache

SPRING AND SPRING BOOT

1. Introduction:
    1. Enterprise Appl
    2. Enterprise Application Layers
        1. Presentation Layer
        2. Business Layer
        3. Data Access Layer
    3. System Architectures
        1. 1-Tier Arch.
        2. 2-Tier Arch.
        3. n-Tier Arch
    4. Types of Enterprise Applications.
        1. Web Applications
        2. Distributed Applications

- 5. Modeled Arch.
    - 1. Model-I Arch.
    - 2. Model-II Arch.
- 6. MVC
- 7. Requirement to user Frameworks
- 8. Types of Frameworks
    - 1. Web Frameworks
    - 2. Application Frameworks
- 9. Differences between Spring and Struts, JSF
- 10. Spring History
- 11. Spring Modules.
    - 1. Spring1.x Modules
    - 2. Spring2.x Modules
    - 3. Spring 3.x Modules
    - 4. Spring 4.x Modules
    - 5. Spring 5.x Modules

2. Steps To Prepare Spring Application [Core Module Application]:
    - 1. Download Spring Framework from the Internet.
    - 2. Provide Spring Setup in Eclipse IDE
    - 3. Prepare Bean Class
    - 4. Prepare Bean Configuration File
    - 5. Prepare Test / Client Appl.

3. Core Module
    - 1. Introduction
    - 2. IOC Containers
        - 1. BeanFactory
            - 1. XmlBeanFactory
            - 2. Resources
                - 1. ByteArrayResource
                - 2. FileSystemResource
                - 3. ClassPathResource
                - 4. InputStreamResource
                - 5. UrlResource
                - 6. ServletContextResource
                - 7. PortletContextResource
        - 2. ApplicationContext
            - 1. ClassPathXmlApplicationContext

        2. FileSystemXmlApplicationContext
        3. WebXmlApplicationContext
3. Beans in Spring Framework
    1. Beans Definition
    2. Beans Configuration
        1. XML Based Configuration
        2. Annotation Based Configuration
        3. Java Based Configuration
    3. Bean Scopes
        1. singleton Scope
        2. prototype Scope
        3. request Scope
        4. session Scope Spring Framework.
        5. globalSession Scope
        6. application Scope
        7. webSocket scope
        8. Custom Scopes in
    4. Bean Lifecycle
        1. Bean Loading
        2. Bean Instantiation
            1. By Constructor
            2. By Static Factory Method
            3. By Instance Factory Method
        3. Bean Initialization and Destruction
            1. By Custom initialization and destruction methods.
            2. By InitializingBean and DesposableBean callback interfaces.
            3. By @PostConstruct and @Predestroy Annotations
    5. Beans Inheritance
    6. Nested Beans
    7. BeanPostProcessor
4. Inversion Of Control[IOC]
    1. Dependency Lookup
        1. Dependency Pull
        2. Contextualized Dependency Lookup
    2. Dependency Injection
        1. Constructor Dependency Injection
        2. Setter Method Dependency Injection
    3. Different Types of Elements Injection
        1. User defined data types elements injection.
        2. List types injection
        3. Set types injection
        4. Map Types Injection 5. Properties types Injection
    4. Circular Dependency Injection

5. Name Spaces
    1. P-Name space
    2. C-Namespace
6. Beans Autowiring of Beans Collaboration
    1. Autowiring and its Modes
        1. no
        2. byName
        3. byType
        4. constructor
    2. Annotation Based Wiring
    3. Auto Discovery or Stereotypes
    4. Java based Autowiring[Java Based Configuration]
7. Method Injection
    1. Lookup Method Injection
    2. Arbitrary Method Replacement
8. Event Handling
    1. ContextRefreshedEvent
    2. ContextStartedEvent
    3. ContextStoppedEvent
    4. ContextClosedEvent
    5. RequestHandledEvent
    6. Custom Events In Spring Framework
9. Bean Validations in Spring Framework
10. Internationalization in Spring Framework
11. Bean Manipulations and Bean Wrappers
12. Property Editors
    1. ByteArrayPropertyEditor
    2. ClassEditor
    3. CustomBooleanEditor
    4. CustomCollectionEditor
    5. CustomNumberEditor
    6. FileEditor [USer defined]
    7. InputStreamEditor
    8. LocaleEditor
    9. PatternEditor
    10. PropertiesEditor
    11. StringTrimmerEditor
    12. URLEditor
    13. Custom Property Editors
13. Profiling
14. Spring Expression Language[SpEL]
    1. SpEL Expressions
    2. SpEL Operators
    3. SpEL Variables

- 4. SpEL Method Invocations
- 5. SpEL Collections

4. Spring JDBC/DAO Module:
    1. Introduction
    2. DAO Definition
    3. Advantages of DAOs
    4. Drawbacks with DAOs
    5. Guidelines to prepare DAOs
    6. Pain JDBC Vs Spring JDBC
    7. JdbcTemplate
    8. NamedParameterJdbcTemplate
        1. Parameter values through Map
        2. Parameter Values through SqlParameterSource
            1. MapSqlParameterSource
            2. BeanPropertySqlParameterSource
    9. SimpleJdbcTemplate
    10. DAO Support Classes
        1. JdbcDaoSupport
        2. NamedParameterJdbcDaoSupport
        3. SimpleJdbcDaoSupport
    11. Spring Batch Updations or Batch Processing
    12. Stored Procedure and Functions in Spring JDBC
        1. Procedures and Functions without CURSOR Types
        2. Procedures and Functions with CURSOR Types
    13. Blob and Clob processing in Spring JDBC
        1. AbstractLobCreatingPreparedStatementCallback
        2. AbstractLobStreamingResultSetExtractor
        3. LobCreator
        4. LobHolder
    14. Connection Pooling in Spring JDBC
        1. Default Connection Pooling Mech.
        2. Third Party Connection Pooling Mechanisms
            1. Apache DBCP
            2. C3P0
            3. Proxool
        3. Application Servers provided Connection Pooling Mechanism
            1. Weblogic12c provided Connection Pooling Mechanism.

5. Spring ORM
    1. Introduction
    2. Hibernate Integration with Spring
        1. Hibernate Introduction
        2. Hibernate Application Development
        3. Spring with Hibernate Integration.

- 3. JPA Integration with Spring
    - 1. JPA Introduction.
    - 2. JPA Application development
    - 3. Spring with JPA Integration.
- 4. iBatis integration with Spring
    - 1. iBatis Introduction.
    - 2. iBatis Application Development.
    - 3. Spring with iBatis Integration.

6. Aspect Oriented Programming [AOP]
    - 1. Introduction
    - 2. AOP Terminology
        - 1. Aspect
        - 2. Advice
        - 3. JoinPoint
        - 4. Pointcut
        - 5. Introduction
        - 6. Target
        - 7. Proxy
        - 8. Weaving
        - 9. Advisor
    - 3. Types of AOPs
        - 1. Proxy Based AOP
        - 2. Declarative Based AOP
        - 3. Annotation Based AOP
    - 4. Advice
        - 1. Before Advice
        - 2. After Advice
        - 3. After-Returning Advice
        - 4. Around Advice
        - 5. After-Throwing Advice
    - 5. Pointcuts
        - 1. Static Pointcut
        - 2. Dynamic Pointcut.
7. Spring Transactions
    - 1. Introduction
    - 2. Transaction Attributes
    - 3. Isolation Levels
    - 4. Programmatic Based Transactions
    - 5. Declarative Based Transactions.
    - 6. Annotation Based Transactions
8. Spring web MVC Module
    - 1. Introduction

- 2. Spring MVC Flow
- 3. Controllers
    - 1. Abstract Controller
    - 2. ParameterizableViewController
    - 3. MultiActionController
    - 4. Command Controllers
        - 1. AbstractCommandController
        - 2. AbstractFormController
        - 3. SimpleFormController
        - 4. AbstractWizardFormController
- 4. Handler Mappings
    - 1. BeanNameUrlHandlerMapping
    - 2. SimpleUrlHandlerMapping
- 5. HandlerInterceptor
- 6. ViewResolvers
    - 1. AbstractCachingViewResolver
    - 2. XmlViewResolver
    - 3. ResourceBundleViewResolver
    - 4. UrlBasedViewResolver
    - 5. InternalResourceViewResolver
    - 6. VelocityViewResolver / FreeMarkerViewResolver
- 7. Spring Exception Handling
- 8. File Uploading and File Downloading
- 9. Internationalization
- 10. Spring MVC with Tiles

9. Spring Web :
    - 1. Introduction
    - 2. Spring Integration with Struts.
    - 3. Spring Integration with JSF.

10. Spring Security
    - 1. Spring Security Introduction
    - 2. Spring Security Features
    - 3. Spring Security XML Based Example
    - 4. Spring Security Java Based Example
11. Spring Boot
    - 1. Introduction
    - 2. Spring Boot Features/ Advantages.
        - 1. Spring Boot Starters.
        - 2. Spring boot Auto configurations
        - 3. Spring Boot Embedded Containers
        - 4. Spring Boot Actuators
        - 5. Spring boot Test

3. Spring boot Core Applications [Core Module].
4. Spring boot JDBC Applications [JDBC Module].
5. Spring boot Hibernate Application [ORM Module]
6. Spring Boot JPA Application [ORM module]
7. Spring Boot Data-JPA Application [ORM Module]
8. Spring Boot Transaction Application [Transaction module]
9. Spring boot Web MVC Application[Web MVC Module]

WEB SERVICES

1. Introduction to Distributed Technology
    1. Why Distributed Technology
    2. Forces/Circumstances in building applications on distributed technology
    3. Evolution and Support of Distributed Technology in Market
    4. Java Support in developing distributed technology applications
2. Role of Data Representation Standards like XML, JSON and YAML and how are they being used in building distributed applications eXtensible Markup Language (XML)
    1. Well-formedness and validity of XML
    2. Why DTD and basics of DTD
    3. Why XSD an alternate to DTD and deep dive on XSD Language and namespace
3. XML Programming
    1. Java api for XML Parsing (jax-p api)
    2. Java Architecture for XML Binding (jax-b api)
4. SIMPLE OBJECT ACCESS PROTOCOL (SOAP) SERVICES
    1. Architecture of SOAP Web Services
    2. Concepts of SOAP Web Services
        - Java Api for Xml Remote Procedurel calls (jaxrpc api) using jaxrpc si implementation

        - Java Api for Xml Web Services (jaxws api)
    1. Jaxws Reference Implementation (jax ws ri)
    2. Apache CXF (SOAP Integration with Spring Framework
    - Developing Provider Program using contact-first and contract-last approach
    - Building Consumer Programs using jaxrpc api and using jaxws api
    - In-Depth coverage of WSDL Document
    - SOAP Web Service Security

5. JAVASCRIPT OBJECT NOTATION
    1. Understand JSON
    2. JSON JAVA API JSON Reader API
    3. JSON PARSING API

4. JSON BINDING API

MICRO SERVICES

1. MICRO SERVICES (USING SPRINGFRAMEWORK & CLOUD)

1. Concept of Monolithic application and Microservice based application development

2. Patterns of Microservices

- **Data Consistency using Saga Pattern**
  - **Choreography-based Saga**
  - **Orchestration-based Saga**
- **API Composer Command Query Responsibility Segregation (CQRS)**

3. Implementing Microservices using Spring Cloud and Netflix

- **Eureka**
- **Ribbon**
- **Feign**
- **Hystrix**