------------------------------------Q_set3.06

25. Can the default web server in the Spring Boot application be disabled?

Spring Boot provides several ways to configure and customize the web server. By default, Spring Boot uses an embedded web server, which is either Tomcat, Jetty or Undertow, depending on the dependencies included in your project. However, you can configure your application to use an external web server or disable the embedded web server altogether. Application.properties: spring.main.web-application-type=none

26.What are the uses of @RequestMapping and @RestController annotations in Spring Boot?

@RequestMapping--- map HTTP requests to handler methods of MVC and REST controllers. **the value parameter is used to specify the path**

@RestController **----create RESTful web services using Spring MVC**. **the parameter value depicts the component name or bean name, whereas in**

Spring AOP, Spring JDBC, Spring Hibernate Interview Questions

27.What is Spring AOP?

Aspect-Oriented Programming (AOP) is one of the key elements of the Spring Framework. AOP praises Object-Oriented Programming in such a way that it also provides modularity.

28.What is an advice? Explain its types in spring.

The advice is an action that we take either **before** or **after** the method execution. The action is a piece of code that invokes during the program execution.

There are five types of advices in the Spring AOP framework: **before, after, after-returning, after-throwing, and around advice**.

Before Advice: executes before a join point.
After Advice: executes after a join point, is called after advice. We use **@After** annotation to mark an advice as After advice.
Around Advice: executes before and after of a join point
After Throwing Advice: An advice that executes when a join point throws an exception.
After Returning Advice: An advice that executes when a method executes successfully.

29.What is Spring AOP Proxy pattern?

AOP proxy: an object created by the AOP framework in order to implement the aspect contracts (advise method executions and so on). In the Spring Framework, an AOP proxy will be a JDK dynamic proxy or a CGLIB proxy.

30.What are some of the classes for Spring JDBC API?

- ***core*** — the core functionality of JDBC. Some of the important classes under this package include *JdbcTemplate*, *SimpleJdbcInsert*, *SimpleJdbcCall* and *NamedParameterJdbcTemplate*.
- ***datasource*** — utility classes to access a data source. It also has various data source implementations for testing JDBC code outside the Jakarta EE container.
- ***object*** — DB access in an object-oriented manner. It allows running queries and returning the results as a business object. It also maps the query results between the columns and properties of business objects.
- ***support*** — support classes for classes under *core* and *object* packages, e.g., provides the *SQLException* translation functionality

31.How can you fetch records by Spring JdbcTemplate?

1. Insert Record In Db With Spring Boot Jdbctemplate.
2. Insert Record In Db With Spring Boot Namedparameter Jdbctemplate.
3. Fetch Auto Generated Primary Key Value After Insert Spring Jdbc.
4. Working With Springboot Namedparameter Jdbctemplate.
5. Execute Stored Procedure In Spring Jdbc.

32.What is Hibernate ORM Framework?

Hibernate ORM **enables developers to more easily write applications whose data outlives the application process**. As an Object/Relational Mapping (ORM) framework, Hibernate is concerned with data persistence as it applies to relational databases (via JDBC).

33.What are the two ways of accessing Hibernate by using Spring.

1. HibernateTemplate: is a Spring-provided helper class that simplifies the use of Hibernate Session API. It handles opening and closing sessions and transactions, as well as exception handling.

2. The SessionFactory is the core interface to Hibernate. It represents a single database session factory, and it is responsible for creating new Session instances.

34.What is Hibernate Validator Framework?

validate the field values in a Java bean. Bean validation API offers some very useful annotations that can be applied to any bean property for the purpose of maintaining data integrity.

35.What is HibernateTemplate class?

1. `HibernateTemplate` is a helper class that simplifies Hibernate data access code.
2. It automatically converts `HibernateExceptions` into `DataAccessExceptions`.
3. The central method of `HibernateTemplate` is `execute` that accepts `HibernateCallback`.

4. The execute method provides Hibernate session handling. It means neither the HibernateCallback implementation nor the calling code needs to explicitly care about retrieving/closing Hibernate Sessions, or handling Session lifecycle exceptions.

5. The HibernateTemplate is instantiated using SessionFactory either in @Configuration class or in XML configuration.

5. In @Configuration class

Spring MVC Interview Questions

36.What is the Spring MVC framework?

a Java framework that is used to develop web applications. It is built on a Model-View-Controller (MVC) pattern and possesses all the basic features of a spring framework, such as Dependency Injection, Inversion of Control.

37.What are the benefits of Spring MVC framework over other MVC frameworks?

The MVC term signifies that it follows the Model View Controller design pattern. So, **Spring MVC is an integrated version of the Spring framework and Model View Controller**. It has all the basic features of the core Spring framework like Dependency Injection and Inversion of Control.

38.What is DispatcherServlet in Spring MVC? In other words, can you explain the Spring MVC architecture?

Spring DispatcherServlet uses special beans to process requests and render the appropriate views. These beans are part of Spring Framework. You can configure them in the WebApplicationContext.

Spring MVC is **a Java framework that is used to develop web applications**. It is built on a Model-View-Controller (MVC) pattern and possesses all the basic features of a spring framework, such as Dependency Injection, Inversion of Control.

39.What is a View Resolver pattern and explain its significance in Spring MVC?

The ViewResolver **provides a mapping between view names and actual views**. The View interface addresses the preparation of the request and hands the request over to one of the view technologies.

Spring MVC is a Web MVC Framework for building web applications. In generic all MVC frameworks provide a way of working with views. Spring does that via the ViewResolvers, which **enables you to render models in the browser without tying the implementation to specific view technology**.

40.What is the @Controller annotation used for?

Annotated class indicates that it is a controller components, and mainly used at **presentation layer**.