

Spring All possible Questions

Q.1 What is Spring?

Ans 1 It is a lightweight, loosely coupled and integrated framework for developing enterprise applications in java.

Q.2 What are the advantages of Spring framework?

- | | |
|---------------------------|-------------------------|
| 1. Predesigned Templates. | 5. fast development |
| 2. loose Coupling | 6. Powerful Abstraction |
| 3. Easy to test | 7. Declarative Support |
| 4. Lightweight | |

Q.3 What is Spring Core Container?

Ans The Spring Core Container contains core beans, context and Expression Language modules.

Q.4 What are the Modules of Spring framework? and Test NG

Ans 1 Test → This layer provides support of testing with JUnit.
Spring Core container →

The spring core container contains, core beans, context and Expression Language modules.

2. Core and Beans → These modules provides IOC and dependency injection features.

3. Context → This module supports internationalization, EJB, JMS Basic Remoting.

BACK

AOP → Aspect and Instrumentation. → This module provides aspect oriented programming implementation, when you can use, advices, etc. to decorate the code.

Data Access integration → This group comprises of JDBC, ORM, ORM and JMS and Transaction modules. These modules basically provides support to interact with database.

Q.5 What is IOC and Dependency injection.

Ans IOC stands for inversion of control basically spring IOC container is the core of spring framework, it creates objects, configure and assemble their dependencies. Manage the entire life cycle. The container uses the dependency injection (DI) to manage the components that makes up the Application.

Q.6 What is the role of IOC Container in Spring?

IOC container is responsible to:

1. Create the instance.
2. Configure the instance.
3. Assemble the dependencies between the objects.

There are two types of IOC container are -

1. Beanfactory → The XML Beanfactory is the implementation class for the Bean factory interface. To use the Beanfactory, we need to create the instance of XML Beanfactory.

2. Using Application Context - class is the implementation class of Application interface. we need to instantiate the classpathXML Application context class to use the Application context.

Q. Explain the Spring Configuration file?

Ans An XML file contains all the information about classes along with their configuration process and how these classes have been interacted with other classes.

Q7 What is Dependency injection?

Ans Dependency injection is the design pattern that removes the dependency from the programming code so that it can be easy to manage and test the application. Dependency injection makes our programming code loosely coupled.

Q8 What are the Two ways to perform dependency injection in Spring framework.

Ans The Spring framework provides two ways to inject dependency.

- By Constructor.
- By setter method.

Q9 What is difference between Beanfactory and ApplicationContext?

Ans Bean factory is the basic container where Application is the advance container. Application context extends the Bean factory interface. Application context provides more facilities than Bean factory such as integration with Spring AOP.

Q10 What is the difference between constructor injection and Setter injection.

Ans Constructor Injection

Setter injection.

1. No partial injection

Partial injection.

2. Doesn't override the setter property

overrides the constructor property if both are defined.

3. Creates new instances if any modification occurs.

Don't create instance if you change the property value.

4. Better for too many properties.

Better for few properties.

Q.11 What is POJO ?

Ans POJO stands for plain old java object and is used for increasing the Readability and reusability of a program. POJO has gained the most acceptance because they are Easy to write and understand.

Q.12 What is autowiring in Spring ?

Ans Autowiring enables the programmer to inject the bean automatically. We don't need to write explicit injection logic.

design pattern or

Q.13 What are the different Bean scopes in Spring ?

Ans There are 5 Beans scope in Spring.

1. Singleton \rightarrow The Bean instance will be only once and same instance will be returned by the IOC container. It is default scope.
2. Prototype \rightarrow The Beans instance will be created each time when requested.
3. Request \rightarrow The Bean instance will be created per HTTP request.
4. Session \rightarrow The Bean instance will be created per HTTP Session.
5. Global session \rightarrow The Bean instance will be created per HTTP global session. It can be used in portlet context only.

Q.14 In which scenario, you will use Singleton and Prototype scope ?

Ans Singleton scope should be used with EJB stateless session bean and prototype with EJB stateless session bean.

Q.15 What are the transaction management support provided by Spring ?

Ans The Spring framework provides two types of transaction Management Support:

1. Programatic Transaction Management: should be used for few Transaction operations.

2. Declarative Transaction Management: should be used for many Transaction operations.

Q.15 What is @Response Body?

A. @ Response body is a spring annotations which Binds to method return value to the web response body.

Q.16 What is different Between Spring Boot and Spring framework?

A.16.

Spring Boot

1. In spring Boot there are default configuration that allow faster Bootstrapping.

2. Spring Boot on the other hand can get an application working with just one dependency.

3. Spring Boot framework is used to develop the REST API's

Spring framework.

In spring framework, you have to build configuration manually.

Spring framework requires a number of dependencies to create a web app.

Spring framework is a java EE framework that is used to build applications.

Q.17 What is front controller class of spring MVC?

A.17 The dispatcher servlet class works as a front controller in spring MVC.

Q.18 What does @ controller annotation?

A.18 The @ controller annotation marks the class as controller class. It is applied on the class.

Q.19 What does @ Request Mapping annotation?

Ans. The @ Request mapping annotation maps the request with the method. it is applied on method.

Q.20 What does the view Resolver class?

Ans. The view Resolver class resolves the view Component to be invoked for the request. It defines prefix and suffix properties to resolve the view component.

Q.21 Which view resolver class is widely used?

Ans. The org.springframework.web.servlet.view.InternalResourceViewResolver class is widely used.

Q.22 Does spring MVC provide validation Support?

Ans. Yes spring MVC provide validation Support.

Q.23 What do you understand by Spring cloud?

Ans. Spring cloud is an integrate software used to integrate with External System. It allows a microservices framework to Build applications that perform restricted amount of data processing.

Q.24 What are the Microservices?

Ans. Microservice is an architectural approach or style that is used to build applications. Microservice architecture provide a rapid and frequent and reliable delivery of large and Complex application.

Q. What are the three Commonly used tool for Microservices?

Ans. Following are the three Commonly used tool for Microservices?

- Wilmock
- Docker
- Hystrix

1. @Document → @Documentation is used to identify a domain object, which is persisted to MongoDB.
2. @Id → This specifies the primary key of an entity.
We are using path variable to send only Id. We used Request Param we receive.
3. @Indexed → This annotation marks the field as indexed in MongoDB.
4. @RestController → The Restcontroller allows to handle all REST APIs such as GET, Post, delete, PUT requests.
5. @RequestMapping → Request Mapping annotation is used to map HTTP request to handler methods of MVC and rest controllers.
6. @Autowired → The Autowired annotation marks a Constructor, ~~Setter method~~ Setter method, properties and Config() Method as to be Autowired that is injecting beans (objects) at runtime by spring dependency injection.
7. @CrossOrigin → cross origin only allows ~~cross~~ Cross-origin HTTP requests from a single origin.
8. @EnableSpringBootApplication → Annotation is used to mark a configuration class that declares one or more @Bean methods and also triggers autoconfiguration and Component Scanning.
9. @EnableEurekaClient → Annotation makes your Spring boot application act as a Eureka client. To register the spring boot application into the Eureka server we need to add the following configuration in our application.
10. @LoadBalanced → annotation will make an instance of created rest template load balanced.