Top 40 Spring Interview Questions and Answers (Updated for 2018)

Introduction

The traditional J2EE application contains an excessive amount of code that does not have much importance in the application. Such an approach distracts the developers from focusing on the application’s business domain. Also, the J2EE applications are tough to take through the testing phase. Because of the problems, a tool based code generation approach was adopted; however, it was a failure. With such experiences in mind, frameworks were introduced. Frameworks provide great flexibility during the runtime. Spring is a very powerful and lightweight application development framework.

Spring Framework is extensively used for-

- Web Application Development
- RESTful Web Services
- Securing Web Applications
- Interaction with DataBase
- Testing Purposes
- Used for Standalone Projects
- Integrate Social Media Login (OAuth)

15
Shares

40 Latest Interview Questions and Answers on Spring, Spring MVC, and Spring Boot

1. What is Tight Coupling?

When a class (ClassA) is dependent on another class’s object (ClassB), then we say ClassA is "tightly" Coupled with ClassB. Spring helps us to create classes in a way that Tight Coupling can be removed and Loose Coupling can be done.

2. What is Loose Coupling?

Loose Coupling removes the dependency of an object (ClassB) on a class (ClassA). Loose Coupling is approached by creating an interface and a setter & getter method, or by using a constructor which takes the interface object.

3. What are Beans in Spring?

When a class is annotated or decorated using the @Component, such a class is called a Bean in Spring. Beans are maintained by Application Context.

Did you know? *How is C Programming still popular?*

4. Explain Bean creation process?

The process of Bean creation has the following phases

(i) Starts with a class (c1) which has the annotation @Component.

(ii) Checks if the component annotated class (c1) is dependent.

(iii) If yes, then Spring will create a bean for that class (c2) too.

(iv) A connection or autowiring will occur between the two classes (c1 and c2) using @Autowired annotation and also through the constructor (c2) or the default case setClass Function (interface the Interface).

5. What is the importance of the annotation @Primary

This annotation is used on a class that needs to be taken by spring on a primary basis. For instance, if ClassX is @Component annotated and is dependent on both Class1 and Class2 (both @Component annotated) then the compiler would report an error. To show the primary class between Class1 and Class2 we use @Primary.

6. What is Dependency Injection?

Dependency Injection is where Spring searches for beans; once the appropriate bean is found, it autowires the bean to the dependent class. Dependency Injection is the process where Spring framework looks for the beans and identifies the dependencies, and creates the instances of beans and autowires them.

7. Explain Inversion of Control (IOC).

In Tight Coupling the dependent class takes the responsibility of creating its dependency. Whereas, in Loose Coupling, we use @Autowired annotation over the dependency class (or reference) and Spring takes control of creating the instance and injects the dependency.

8. What are the roles of an IOC (Inversion of Control) Container?

IOC Container does the following things-

(i) Find Beans

(ii) Identify their dependencies and wire the dependencies

(iii) Manage Lifecycle of the Bean (creation, processing, and destruction)

Discover the *Difference Between Java And Javascript*

9. What is Application Context?

It is an advanced version of IOC Container. It provides all the functionalities of Bean Factory and also provides things like AOP, Internationalization capabilities, web application context (request, session, etc).

10. Explain the process of creating an ApplicationContext in Spring.

The ApplicationContext can be defined in two ways (i) using XML, (ii) using @Configuration. Once the configuration is done in any of the ways defined above, the ApplicationContext is created using new ClassPathXmlApplicationContext. The ClassPathXmlApplicationContext looks for the XML files, using this is one of the two ways. The other way is to use AnnotationConfigApplicationContext.

11. Explain Component Scan.

Component Scan is one method of asking Spring to detect Spring-managed components, the input for this search is the packages. Two methods are available to define a Component Scan-

(i) Java Configuration; wherein, we use the @Component annotation to which we specify all the packages, for which Spring does the search.

(ii) XML Configuration- we use <context:component-scan base-package=”com.demo.compscanex”/>

12. How do you perform the same (above question) in Spring Boot?

In Spring Boot the annotation used to perform the scan is @SpringBootApplication. This annotation on a class would automatically initiate the component scan on the package where they are in.

13. Differentiate @Component, @Repository and @Service and @Controller?

Typically a web application is developed in layers like the controller (which is the initial point of client communication), business (where the actual code or logic of the application is written) and DAO (where the database connections and interaction happens). In such an architecture web application, @Component can be used in any of the layers. Whereas, the @Controller is used in the controller/web layer. @Service is used in the business layer and @Repository is used in the DAO layer.

14. List out the different scopes of Bean.

(i) Singleton: throughout the spring context only one instance is created.

(ii) Prototype: a new bean is created whenever requested.

(iii) Request: Every HTTP Request creates a bean.

(iv) Session: A bean for every HTTP Session.

15. List out the types of Dependency Injection.

The types of Dependency Injection-

(i) Setter Injection and (ii) Constructor Injection.

16. What is the difference between the Configuration types XML and Annotation?

These are the two ways of setting up the configuration, and they perform in the say way. Though, when the annotation approach is taken very less amount of code is written and the result would be the same as compared to the XML approach.

17. List out the ways Autowiring is done.

(i) byType

(ii) byName

(iii) Constructor (same as byType, but through constructor)

You may also like: *Top 5 Skills That Make You A Sure Shot Programmer*

18. What is Dirty Read?

When a transaction (t1) is meant to read the changes that are performed by another transaction (t2) and provided transaction t2 is not committed yet; then in such a situation, the transaction t1 is called Dirty Read transaction.

19. List out the new features available in Spring Framework 4.0 and Spring Framework 5.0?

Spring 4.0 is the first to support Java features. Spring 5.0 has the support for Reactive Programming and Kotlin.

20. What is a FrontController?

In FrontController, the Servlet will not get the first request; the first request would go to FrontController and the request is passed on to the right servlet. In other words, DispatcherServlet is the front controller which intercepts all the requests from the client and then dispatches to appropriate controllers.

21. What is a ViewResolver?

ViewResolver enables a web application to select its view (such as JSP) dynamically. ViewResolver gets a name which is appended by /WEB-INF/views and a .jsp. All the display on the content is done in an HTML page.

22. List out all the concepts that are available in the MVC Architecture?

(i) The browser sends a request to DispatcherServlet

(ii) DispatcherServlet knows the HanderMapping and can find the appropriate controllers

(iii) Controllers execute the request and put the data in the model and return back the view name to the DispatcherServlet.

(iv) DispatcherServlet uses the view name and ViewResolver to map to the view.

23. Explain Model Attribute?

The annotation @ModelAttribute is decorated on a method typically present inside a Controller. This will help the method to be available in all other methods available in the controller.

24. What is a Session Attribute?

The annotation @SessionAttributes ("argument") is decorated on class (Controller). The attribute (argument) that is present in Model is available in the session.

25. Explain the @InitBinder?

This annotation is decorated on a method in which a date format is declared, and throughout the class, the defined date format is used. Whenever the binding happens with a date field @InitBinder; annotation says to use the CustomDateEditor, which in return uses the date format mentioned in @InitBinder.

26. Define @ControllerAdvice?

This annotation is used when logic needs to be implemented commonly in multiple classes (Controllers). For instance, if an Exception or its subclasses, or when an exception is raised in the classes, it will be handled by a method annotated with @ExceptionHandler. Whenever an exception occurs in any of the controllers, the exception is handled by the method annotated with @ExceptionHandler.

27. Why Spring Boot?

Spring-based applications have a lot of configuration (boiler-plate code). In Spring MVC, a lot of configuration is required (like component scan, dispatcher servlet, view resolver, etc). Whereas, in Spring Boot the boiler-plate code is not required.

28. Spring vs Spring MVC vs Spring Boot?

Spring: the most important feature of Spring is Dependency Injection or Inversion of Control.

Spring MVC: provides a decoupled approach in developing web applications. Concepts like DispatcherServlet, ModelAndView, ViewResolver makes web application development easy.

Spring Boot: makes the configuration very easy and automatic using a feature called Auto Configuration, in which the DispatcherServlet is done by Spring internally.

29. What is the role of @SpringBootApplication?

This annotation is used to launch up the entire application. Internally, @SpringBootApplication does the following,

@SpringBootConfiguration: same as @Configuration in a Spring Application.

@EnableAutoConfiguration: auto-configures the classes available in the classpath.

@ComponentScan: all the classes available under a package will be scanned when this annotation is applied.

Explore the other Programming courses here!

30. What does an Embedded Server mean in Spring Boot?

To deploy any web application a server like Tomcat is required. In Spring Boot a server (like Tomcat) is available as part of the application in a jar. The concept of Embedded Server makes the deployment of application very easy and independent.

31. Why do we use application.properties?

The file application.properties is used to configure things like database details, log generation, security (username/password), serialization, etc.

32. What is Spring JDBC?

Spring JDBC uses methods like update (query), execute (query) and query (SQL, resultSetExtractor) to interact with the database.

33. What is the difference between JDBC and Spring JDBC?

In JDBC, the checked exceptions need to be written; whereas, in Spring JDBC those exceptions are made into Runtime Exceptions. Which means, exception handling is not manually done in Spring JDBC.

34. What is JPA?

Java Persistence API (JPA) defines the mapping from Java Object to a Database Table. The procedure to map a Java object to a row in a database table is defined in JPA. JPA provides a lot of useful annotations, using which the relationship between classes and tables are defined.

35. What is Hibernate?

Once the mapping is done, Hibernate (a JPA Implementation) will help us create query under the hood and interact with the database.

36. Describe the cases in which the Dependency Injection is done through Constructors and Setters?

When the dependencies are required/mandatory, the Constructor approach is selected. And when the dependencies are optional then the Setters approach is used.

37. What is the importance of POM.XML file?

Project Object Model (POM) is an XML formatted file in which all the configuration for a maven project is defined. The most commonly used tags in POM.XML are <groupid>, <artifactId>, <version>, <packaging> and a few more.

38. What does the @RequestParam annotation do?

This allows the server side to read from data and automatically bind it to a parameter coming into the method.

39. What is Spring Security?

Spring Security provides security services to J2EE applications. Spring Security is implemented using Servlet Filters under the hood. Servlet Filters are used to pre-process or post-process web requests.

40. What is CSRF?

Cross-Site Request Forgery (CSRF) is a security attack where a fraudulent website tricks the user into performing an event on the web application that he/she is logged into. For instance, if the user is logged into the online banking account, this attack tricks the user into transferring the money to an unknown person.

Certification:

It is good to get certified as it not only shows your commitment towards learning the technology, but also the topics that were explored by you while preparing for the certification. A certificate along with good knowledge and work experience is the finest qualification one can have to make a good career.

Conclusion:

Currently, we have Spring Framework 5.0 in trend, which was a major release after Spring 4. Spring Framework 5.0 has features like Reactive Programming Model, Functional Support with Kotlin, Library Support, etc. Reactive Programming is a major achievement by the Spring team. Soon, projects will start implementing the reactive features.

Find other interesting reads here:

Top 30 Interview Questions and Answers on Angular 5

Top 55 Java Interview Questions and Answers for 2018

About The Author

Kiran Ramanadham is a Software Engineer by profession and has been working on frameworks like Angular and Spring for the past few years. Apart from this, as part of his routine job, he likes to write articles, blogs, tutorials on the same technologies. Not only an author and a developer, Kiran is very passionate about teaching and has helped many students and novice programmers by offering classes on technologies like HTML, CSS, Bootstrap, Javascript, JQuery, Typescript, Angular, JSP, Java, Servlets, Hibernate, Spring (MVC & Boot).

Kiran Ramanadham