# Top Spring Interview Questions

1. List down the important features of Spring Framework.

|  |  |
| --- | --- |
| **Features of Spring Framework** | |
| **Feature** | **Description** |
| *Lightweight* | Spring Framework is lightweight with respect to size and transparency. |
| *Inversion Of Control* | Loose coupling is achieved using Inversion of Control where objects give their own dependencies instead of creating or looking for a dependent object in Spring. |
| *Aspect Oriented Programming* | Spring Framework separates the application business logic from system services,  supports Aspect Oriented Programming and enables cohesive development. |
| *Container* | Spring Framework creates and manages the lifecycle and configuration of application objects. |
| *MVC Framework* | Spring Framework is an MVC web application framework which is configurable via interfaces and accommodates multiple view technologies. |
| *Transaction Management* | For transaction management,Spring framework provides a generic abstraction layer. |
| *JDBC Exception Handling* | The JDBC abstraction layer of the Spring Framework offers an exception hierarchy, which simplifies the error handling strategy. |

2. Name the different modules of the Spring framework.

The Spring framework has the following modules:

* JDBC module
* ORM module
* OXM module
* Java Messaging Service(JMS) module
* Transaction module
* Web module
* Web-Servlet module
* Web-Struts module
* Web-Portlet module

3. What is a Spring configuration file?

A Spring configuration file is essentially an XML file that contains information on classes and describes how they are configured and introduced to each other.

4. Explain Dependency Injection in the context of Spring framework.

Dependency Injection is a design pattern that allows users to remove hard-coded dependencies and ensure that the application is loosely coupled, extendable and maintainable. The dependency injection design pattern is used to move the dependency resolution from compile to runtime.

5. Which are the two types of Dependency injections?

* Constructor-based dependency injection
* Setter-based dependency injection

6. How do you implement dependency injection in Spring applications?

There are three ways to configure dependency injection in Spring: XML based configuration, Java based configuration and Annotation based configuration.

7. List some of the important annotations in annotation-based Spring configuration.

The important annotations are:

* @Required
* @Autowired
* @Qualifier
* @Resource
* @PostConstruct
* @PreDestroy

8. In the context of Spring framework, explain aspect-oriented programming.

Aspect Oriented Programming basically breaks down program logic into smaller chunks called “concerns”. The functions across multiple points of an application are called cross-cutting concerns and these operate independent of the application’s core business logic. Some of the important Aspects in the context of Spring framework are; logging, auditing, caching and declarative transaction.

9. Explain Bean in Spring.

Beans are objects that form the backbone of a Spring application. They are managed by the Spring IoC container. In other words, a bean is an object that is instantiated, assembled, and managed by a Spring IoC container.

10. List the different Scopes of Spring bean.

There are five Scopes defined in Spring beans. They are:

* singleton
* prototype
* request
* session
* global-session

11. What is the role of the Spring bean configuration file?

The Spring bean configuration file defines all the beans that will be initialized by Spring Context. When an instance of Spring ApplicationContext is created, it reads the spring bean xml file and initialize all of them. Once the context is initialized, it can be used to get different bean instances.

12. Explain the Spring bean lifecycle.

Spring Beans are initialized, along with all the dependencies by Spring Container. When the “context” is destroyed, it also destroys all the corresponding initialized beans. IN rare instances, beans need some level of validation before they can be used. The Spring framework also provides support for post-initialization and pre-destroy methods in spring beans.

13. What are the different types of Spring bean autowiring?

* autowirebyName
* autowirebyType
* autowire by constructor
* autowire with @Autowired and @Qualifier annotations

14. Explain the role of DispatcherServlet and ContextLoaderListener.

DispatcherServlet is basically the front controller in the Spring MVC application as it loads the spring bean configuration file and initializes all the beans that have been configured. If annotations are enabled, it also scans the packages to configure any bean annotated with @Component, @Controller, @Repository or @Service annotations.

ContextLoaderListener, on the other hand, is the listener to start up and shut down the WebApplicationContext in Spring root. Some of its important functions includes tying up the lifecycle of ApplicationContext to the lifecycle of the ServletContext and automating the creation of ApplicationContext.

15. Explain the role of InternalResourceViewResolver and MultipartResolver.

InternalResourceViewResolver is one of the implementations of the ViewResolver interface that allows you to view the page directory and suffix locationsthrough the bean properties.

MultipartResolver, on the other hand is the interface that is used for uploadingfiles.  –CommonsMultipartResolver andStandardServletMultipartResolver are two implementations provided that are provided by the Spring framework for file uploading.

16. How do you create ApplicationContext in a standalone Java program?

We can do this in three ways:

* **AnnotationConfigApplicationContext:** If we are using annotations for configuration, then we can use this to initialize the container and get the bean objects.
* **ClassPathXmlApplicationContext:** If we have spring bean configuration xml file in the Java application, then we can use this class to load the file and retrieve the container object.
* **FileSystemXmlApplicationContext:** This is quite similar to ClassPathXmlApplicationContext except for the fact that the xml configuration file can be loaded from any location in the file system.

17. Explain the uses of the JDBC template in Spring.

Spring simplifies database access handling with the Spring JDBC Template.

The Spring JDBC Template has many advantages compared to the standard JDBC:

* The Spring JDBC template automatically cleans up the resources; like releasing database connections.
* The Spring JDBC template converts the standard JDBC SQLExceptions into RuntimeExceptions. This ensures faster response time to identify and eliminate errors.

18. What kinds of transaction management support does Spring offer?

* **Programmatic Transaction Management**: for operations with lesser transaction, and
* **Declarative Transaction Management**: for larger number of transactions.

19. Explain the difference between Concern and Cross-cutting concern in Spring AOP.

Simply put, Concern is the desired behavior in a module of an application. It is the core functionality the programmer wants to implement.

Cross-cutting concern, on the other hand, is the Concern that is applicable across the entire application. Examples of Cross-cutting concern would be security, data transfer, logging etc.

20. Explain Advice, in the context of Spring.

Advice is an implementation of Aspect. It is inserted into an application at Join Points. There are different types of Advice including “around,” “before” and “after”.

21. What is a JoinPoint, in the context of Spring?

A JoinPoint is an opportunity within the code to which we can apply an Aspect. In Spring programming, a Join Point always represents a method execution.

22. What kind of JoinPoints does Spring support?

The Spring framework supports method executionJoinPoints.

23. What is a Pointcut?

Pointcut is a predicate that matches join points. A point cut defines at what JoinPoints an advice should be applied.

24. What is a Target Object?

Target Object is a proxy object that is advised by one or more aspects.

25. What is Weaving?

It is the process of linking the Aspect with other applications.

*Got a question for us? Please mention it in the comments section and we will get back to you.*