

## **1. \*Spring Framework:**

- Spring Framework es un framework de desarrollo de aplicaciones empresariales para la plataforma Java. Proporciona una infraestructura completa para el desarrollo de aplicaciones Java empresariales, abordando aspectos como la inversión de control (IoC), la inyección de dependencias, la transacción, la seguridad, entre otros.
- Spring Framework se basa en principios de diseño como la modularidad, la reutilización y la extensibilidad, lo que facilita el desarrollo de aplicaciones escalables y mantenibles.

## **2. \*Spring Boot:**

- Spring Boot es una extensión del framework Spring que se enfoca en simplificar el proceso de configuración y desarrollo de aplicaciones Spring.
- Spring Boot incorpora opiniones preconfiguradas y convenciones para simplificar la configuración inicial de la aplicación. Proporciona una forma rápida y sencilla de crear aplicaciones Spring listas para la producción con mínima configuración.
- Spring Boot también incluye un servidor integrado (por ejemplo, Tomcat, Jetty) para simplificar la implementación y ejecución de aplicaciones.

## **3. \*Spring MVC:**

- Spring MVC (Modelo-Vista-Controlador) es un marco de trabajo dentro de Spring para el desarrollo de aplicaciones web basadas en el patrón de diseño Modelo-Vista-Controlador.
- Spring MVC proporciona una estructura para desarrollar aplicaciones web de forma modular, separando las preocupaciones de la lógica de negocio (modelo), la presentación (vista) y el control de flujo (controlador).
- Con Spring MVC, puedes crear controladores para manejar las solicitudes de los clientes, definir vistas para mostrar la respuesta al usuario y manejar la lógica de negocio dentro de los modelos.

	<b>Spring.</b>	<b>Spring Boot.</b>
Uso	Para la construcción o el desarrollo de aplicaciones, el marco de primavera se considera el marco Java EE más utilizado.	Para desarrollar REST APIs El marco de arranque de primavera es ampliamente utilizado.
Ventaja	Los desarrolladores se han vuelto más productivos a medida que este marco ha hecho que el desarrollo de Java EE sea simple.	Es el marco más fácil para desarrollar aplicaciones web a medida que se reduce la longitud de código cuando se utiliza este marco.
Características clave o primaria	Inyección de dependencia	Autoconfiguración: En función del requisito, las clases se configuran automáticamente por Spring Boot.
Desarrollo de aplicaciones	Este marco ayuda a simplificar las cosas permitiéndonos construir aplicaciones acopladas vagamente.	Las aplicaciones independientes se pueden construir utilizando este marco de arranque de primavera con configuraciones mínimas o menos.
Código de calderas	Para hacer una tarea mínima, el desarrollador necesita escribir una gran cantidad de código o, más específicamente, código de caldera.	El código de la caldera se reduce en cierta medida si utilizamos Spring Boot.
Servidor	El servidor necesita ser configurado explícitamente para el propósito de prueba del Proyecto de Primavera.	Servidores empotrados como Jetty, Tomcat, etc., son ofrecidos por el marco de la Bota de Primavera.
Base de datos	Este marco no proporciona apoyo a la base de datos en memoria.	Ofrece varios plugins para trabajar con una base de datos integrada y en memoria como H2.
Dependencias	En pom.xml, las dependencias de nuestro proyecto de Primavera son definidas manualmente por los desarrolladores.	Hay un concepto de inicio en el archivo pom.xml de nuestra Bota de Primavera que se utiliza para el cuidado internamente de la descarga de JAR de dependencia basado en la necesidad de arranque de primavera.
Necesidad del descriptor de despliegue	Se requiere descriptor de implementación para ejecutar aplicaciones de Primavera.	No se requiere descriptor de implementación para Spring Boot.
Configuraciones	En el marco de la primavera, las configuraciones deben construirse manualmente.	Para Spring Boot, hay configuraciones predeterminadas para las que se permite una bootstrapping más rápida.

Pruebas	Debido a una gran cantidad de código fuente, las pruebas en primavera son más difíciles que las pruebas en Spring Boot.	Debido a la cantidad reducida de código, la prueba es más fácil en Spring Boot.
Configuración XML	En primavera, se requiere configuración XML.	En Spring Boot, la configuración XML no es necesaria.
Plugins	El marco de Primavera no proporciona ningún plugin.	Para la herramienta de construcción Maven y Gradle, los plugins son proporcionados por la Bota de Primavera. Los plugins proporcionan varias características, incluyendo el embalaje de los frascos ejecutables.
Herramientas CLI	Para el desarrollo y ensayo de aplicaciones, Spring Framework no proporciona ninguna herramienta CLI.	Spring Boot proporciona la herramienta CLI para el desarrollo y prueba de aplicaciones.
Autización HTTP	La autenticación HTTP permite la autenticación de seguridad. Se necesitan varias configuraciones y dependencias para permitir la seguridad. Las dependencias estándar de primavera-seguridad-config y spring-security-web deben establecer la seguridad en la aplicación.	Sólo se necesita definir la dependencia de la seguridad de primavera-boot-starter-starter porque todas las dependencias pertinentes se añadirán automáticamente al camino de clase.

	<b>Spring Boot</b>	<b>Spring MVC</b>
Definición	Spring Boot se considera un módulo del marco de primavera para empaquetar la aplicación basada en Spring con valores predeterminados sensatos.	Spring MVC se considera el marco web basado en el controlador de modelos en el marco de primavera.
Uso	Para construir un marco impulsado por la primavera, se proporcionan configuraciones predeterminadas.	Para la construcción de aplicaciones web, las características listas para usar son proporcionadas por él.
Configuraciones	Los desarrolladores no necesitan construir las configuraciones manualmente.	En Spring MVC, las configuraciones deben construirse manualmente.
Dependencias	Se evita el código de la caldera y las dependencias están envueltas en una sola unidad.	Cada dependencia se especifica por separado por ella.
Descriptor de implementación	No se requiere un descriptor de despliegue en el caso de Spring Boot.	Se requiere un descriptor de despliegue en el caso de Spring MVC.
Hora requerida	Es muy eficiente de usar. El tiempo de desarrollo se reduce con este marco, aumentando así la productividad de los ingenieros.	En el caso de Spring MVC, se necesita mucho más tiempo para lograr lo mismo.
Traces	El potente procesamiento por lotes es proporcionado por él.	El potente procesamiento por lotes no es proporcionado por él.
Capas y componentes	Hay cuatro tipos de capas de presentación: capa de presentación, capa de negocio, capa de persistencia y la capa de base de datos.	Hay cuatro componentes en MVC tales como Modelo, Vista, Controlador, Controlador de Front, o Clase DispatcherServlet.
Desarrollo	Junto con las páginas web dinámicas y los servicios web RESTFUL, Spring Boot también puede desarrollar otras aplicaciones.	Las páginas web dinámicas y los servicios web RESTFUL se desarrollan utilizando Spring MVC.