Spring is a framework for developing Java applications used by millions of programmers. It helps to create high-performance, easy-to-test, code-reusable applications…

Spring is lightweight and transparent (lightweight: small in size, the basic version is only about 2MB; transparent: works transparently to programmers).

Spring is an open-source code, developed, shared, and has a very large user community.

Spring Framework is built on two main design principles: Dependency Injection and Aspect-Oriented Programming.

The core features of Spring can be used to develop Java Desktop, mobile applications, Java Web. Spring's main goal is to make it easier to develop J2EE applications based on the POJO (Plain Old Java Object) usage pattern. This layer provides support for testing with JUnit and TestNG. Includes spring core, beans, context, and expression language (EL) modules. Spring core, bean provides IOC and Dependency Injection features. Spring Context supports multi-language (internationalization), Java EE features such as EJB, JMX. Expression Language is extended from Expression Language in JSP. It provides support for setting/getting values, improved methods for accessing collections, indexes, logical operators, etc. These modules support the implementation of Aspect-Oriented Programming, support integration with AspectJ. This group includes JDBC, ORM, OXM, JMS, and Transaction modules. These modules provide the ability to communicate with the database. Also known as Spring MVC This group includes Web, Web-Servlet… supporting the creation of web applications.

Spring Boot is an open source Java-based framework used to create a micro Service. It is developed by Pivotal Team and is used to build stand-alone and production ready spring applications. This chapter will give you an introduction to Spring Boot and familiarizes you with its basic concepts.

Micro Service is an architecture that allows the developers to develop and deploy services independently. Each service running has its own process and this achieves the lightweight model to support business applications.

Spring Boot provides a good platform for Java developers to develop a stand-alone and production-grade spring application that you can **just run**. You can get started with minimum configurations without the need for an entire Spring configuration setup.

Spring Boot automatically configures your application based on the dependencies you have added to the project by using annotation. For example, if MySQL database is on your classpath, but you have not configured any database connection, then Spring Boot auto-configures an in-memory database.

Handling dependency management is a difficult task for big projects. Spring Boot resolves this problem by providing a set of dependencies for developers convenience.

Spring Boot application scans all the beans and package declarations when the application initializes. You need to add the annotation for your class file to scan your components added in your project.