

1. **Spring:** It's a comprehensive framework for building Java-based applications, offering a wide range of functionalities like dependency injection, aspect-oriented programming, and more. For instance, it allows developers to focus on business logic instead of boilerplate code for database connections.
2. **Spring Boot:** An extension of the Spring framework, Spring Boot simplifies the process of setting up and developing Spring applications by favoring convention over configuration. It enables quick creation of stand-alone, production-grade applications with minimal setup, like running a web application with just an annotation and a main method.
3. **Spring vs. Spring Boot:** Spring Boot is part of the Spring ecosystem, designed to make the development process faster and easier by building on the foundational features provided by the Spring framework.
4. **Spring Platform vs. Spring Framework:** The Spring platform encompasses all Spring-related technologies, including the core Spring framework, Spring Boot, Spring Data, etc. The Spring framework is a key component that provides the foundational functionalities for other projects within the platform.
5. **Dependency Injection in Spring:** This design pattern is implemented in Spring using annotations like `@Autowired`, allowing the framework to inject dependencies at runtime, ensuring more modular and manageable code.
6. **Inversion of Control (IoC) in Spring:** IoC is a principle where the control of objects is inverted from the program to the Spring IoC container. This container manages object creation, configuration, and assembly, promoting a cleaner, more modular application structure.