1. What is Spring?

The Spring Framework is an application framework and inversion of control container for the Java platform. The framework's core features can be used by any Java application.

2. What is Spring Boot?

Spring Boot is an open-source Java-based framework used for programming standalone

3. What is the relation between Spring platform and Spring Boot?

**Spring Boot is basically an extension of the Spring framework, which eliminates the boilerplate configurations required for setting up a Spring application**. It takes an opinionated view of the Spring platform, which paves the way for a faster and more efficient development ecosystem.

Spring Boot is a convention-over-configuration extension for the Spring Java platform intended to help minimize configuration concerns while creating Spring-based applications.

4. What is the relation between Spring platform and Spring framework?

The Spring Framework serves as the core for Java enterprise applications, offering features like dependency injection and transaction management. The broader Spring Platform includes additional projects and tools such as Spring Boot and Spring Cloud, expanding functionality for building modern, cloud-native applications.

5. What is Dependency Injection and how is it done in the Spring platform/framework?

Dependency Injection (DI) in the Spring framework involves managing dependencies between components by having external entities provide them. This is achieved through inversion of control (IoC) containers, which create and inject dependencies into beans based on configuration metadata. Dependency injection can be done via constructor injection or setter injection, and Spring supports various bean scopes for lifecycle management. It promotes loosely coupled and maintainable code by decoupling components and simplifying dependency management.

6. What is Inversion of Control (IoC) and how is it related to Spring?

Inversion of Control (IoC) is a principle where control over object creation and management is shifted to a framework or container. In Spring, the IoC container manages the lifecycle of beans and their dependencies, promoting loose coupling and making the code more modular and maintainable. Dependency Injection (DI) is closely related to IoC, as it's the mechanism through which dependencies are provided to beans by the IoC container. Overall, IoC in Spring enhances configurability, extensibility, and testability of applications.