1. What is Spring?

Spring is a popular Java-based framework that helps developers build robust, scalable, and maintainable enterprise applications.
 It takes care of common concerns like managing objects, handling security, connecting to databases, and more—so developers can focus on writing business logic.

2. What is Spring Boot?

Spring Boot is like Spring made easy.

It's built on top of Spring and helps you get started quickly without writing lots of configuration code. It does things like:

- Automatically setting up things (Auto Configuration)

- Providing production-ready features (metrics, health checks, etc.)

- Embedding a web server like Tomcat so you can run your app with just a `main()` method

3. What's the relation between Spring and Spring Boot?

- Spring Boot uses the Spring framework underneath.

- It's not a replacement, but rather a helper that reduces the boilerplate and config work needed to use Spring.

- You still get all the power of Spring, just with much less setup.

4. What's the relation between Spring platform and Spring framework?

- Spring platform is the whole ecosystem—it includes the core Spring framework, plus projects like Spring Boot, Spring Security, Spring Data, etc.

- Spring framework is just one part of the platform—specifically the core part that provides things like Dependency Injection, AOP, and more.

Think of the Spring platform as the full toolbox, and the Spring framework as the most essential tools in that box.

5. What is Dependency Injection (DI), and how does Spring do it?

Dependency Injection is when an object's dependencies (other objects it needs) are provided to it, instead of the object creating them itself.

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

IoC is a design principle where control of object creation and management is handed over to a container (like Spring), rather than being handled manually by your code.

Spring's IoC container takes care of:

- Creating objects (beans)

- Injecting dependencies

- Managing the entire object lifecycle