Concept Summary

Spring framework

Spring framework can be thought of as a smart contractor for your Java application. It does the heavy lifting of managing the complex infrastructure concerns that would otherwise be overwhelming. Technically, we can say Spring is a lightweight and extensible framework that acts as a container handling the creation, configuration, and lifecycle of Java objects, usually referred to as beans, enabling you to overcome tight coupling using Dependency Injection (DI).
This helps developers focus on core business logic.

Inversion of Control (IoC)

Usually, when class implementations overlap or are related in a way, they end up being tightly coupled, where the creation of one requires that it also be created during the same instance. However, an IoC container provides a way to create, configure, manage, and control the entire lifecycle of objects (beans). Using Dependency Injection, the Spring container (IoC) manages the instantiation, configuration and use of the object based on input and design.

REST

REST mean Representational State Transfer, which has to do with how the state of objects and data is maintained. This includes their creation, retrieval, update and deletion, a common term known as CRUD. REST follows certain principles that lead to the development of a RESTful API that allows for inter-application communication when managing resource objects. Some of the principles are HTTP standards that allow for the transfer and storage of data between applications.