The Spring container is responsible for creating the beans in your application and coordinating the relationships between those objects via DI. But it’s your responsibility as a Developer to tell Spring which beans to create and how to wire them together.

* When it comes to bean wiring specification, Spring is incredibly flexible, offering three primary wiring mechanisms:
* Explicit configuration in XML
* Explicit configuration in Java
* Implicit bean discovery and automatic wiring

The choice is largely a matter of personal taste, and you’re welcome to choose the approach that feels best for you.

* It’s great that you have many choices about how to wire beans in Spring, but at some point you must select one.
* Spring’s configuration styles are mix-and-match, so you could choose XML to wire up some beans, Use Spring’s Java-based configuration (JavaConfig) for other beans, and let other beans be automatically discovered by Spring.
* Recommendation is to lean on automatic configuration as much as you can. The less configuration you have to do explicitly, the better. When you must explicitly configure beans (such as when you’re configuring beans for which you don’t maintain the source code), I’d favor the type safe and more powerful JavaConfig over XML. Finally, fall back on XML only in situations where there’s convenient XML namespace you want to use that has no equivalent in JavaConfig.