1.5 Spring Boot vs Spring Framework

Feature	Spring Framework	Spring Boot
1. Starter Dependencies	You have to ***manually add*** individual ***dependencies*** (e.g., Spring MVC, Jackson, etc.)	Spring Boot provides ***starter POMs*** like ***spring-boot-starter-web*** that ***auto-import*** required ***dependencies***.
2. Auto Configuration	Requires ***manual configuration*** of ***beans***, ***data sources***, ***view resolvers***, etc.	Uses ***@EnableAutoConfiguration*** to automatically configure based on ***classpath contents***.
3. Externalized Configuration	**Configuration** mostly through ***applicationContext.xml*** or ***Java Config***.	Supports ***application.properties*** or ***application.yml*** for ***external config*** (e.g., **port**, **DB URL**).
4. Embedded Servers	You need to deploy ***WAR*** to an ***external server*** like **Tomcat**.	Comes with ***embedded servers*** (**Tomcat**, **Jetty**, etc.) → Just run the **jar** with ***main()*** method.
5. Built-in Metrics & Health Checks	**No built-in** support; you need to **add** **Actuator**-like **tools** manually.	**Built-in** support via ***Spring Boot Actuator*** → Get health checks, metrics, and endpoints easily.