**🔰 What is Spring Boot?**

**Spring Boot** is part of the **Spring Framework ecosystem**, designed to simplify the setup, development, and deployment of new Spring applications.

**🔧 Key Features:**

| **Feature** | **Description** |
| --- | --- |
| **Auto Configuration** | Automatically configures Spring applications based on the libraries on the classpath. |
| **Starter Dependencies** | Bundled dependencies to reduce boilerplate (e.g., spring-boot-starter-web). |
| **Embedded Server** | Comes with Tomcat/Jetty embedded — no need to deploy to external servers. |
| **Production Ready** | Health checks, metrics, logging via **Spring Boot Actuator**. |
| **Spring Initializr** | Tool to bootstrap Spring Boot projects quickly (https://start.spring.io). |

**📦 Common Spring Boot Starters**

| **Starter Name** | **Purpose** |
| --- | --- |
| spring-boot-starter-web | REST APIs / MVC applications |
| spring-boot-starter-data-jpa | Database and JPA integration |
| spring-boot-starter-security | Spring Security for auth/authz |
| spring-boot-starter-test | Testing (JUnit, Mockito) |
| spring-boot-starter-actuator | Monitoring & health checks |

**⚙️ How It Works – Basic Example**

**✅ 1. Create a Main Class**

java

CopyEdit

@SpringBootApplication

public class MyApp {

public static void main(String[] args) {

SpringApplication.run(MyApp.class, args);

}

}

* @SpringBootApplication is a convenience annotation combining:
  + @Configuration
  + @EnableAutoConfiguration
  + @ComponentScan

**✅ 2. Sample REST Controller**

java

CopyEdit

@RestController

public class HelloController {

@GetMapping("/hello")

public String sayHello() {

return "Hello from Spring Boot!";

}

}

When you run the application, you can access:  
http://localhost:8080/hello

**🧪 How to Run a Spring Boot App**

1. **Using Maven**

bash

CopyEdit

mvn spring-boot:run

1. **Or package as a JAR**

bash

CopyEdit

mvn clean package

java -jar target/myapp-0.0.1-SNAPSHOT.jar

**✅ Benefits of Using Spring Boot**

* Rapid development
* No XML configurations
* Built-in server
* Large ecosystem
* Production-ready tools

**🎯 When to Use Spring Boot**

* REST APIs
* Microservices
* Web applications
* Enterprise-grade backend systems