Testing Spring Boot Test.

Integration testing in Spring Boot applications is an important step in ensuring that different parts of your application work together correctly. It involves(implica) testing modules in combination to check if they interact with each other as expected. Spring Boot provides several tools and annotations to facilitate integration testing, making the process efficient and straightforward.

**@SpringBootTest:** This annotation is used to load the complete application context for the test, making it ideal for integration tests. It can be used when a test needs to bootstrap the entire container. The annotation can be customized to start a web environment on a random port or the default port.

Testing diferente Layers.

@SpringBootTest

It starts the embedded server, creates a web environment and then enables @Test methods to do integration testing.

By default, @SpringBootTest does not start a server. We need to add attribute

webEnvironment to further refine how your tests run. It has several options:

★ MOCK(Default): Loads a web ApplicationContext and provides a mock web

environment

★ RANDOM\_PORT: Loads a WebServerApplicationContext and provides a real web

environment. The embedded server is started and listen on a random port. This is the

one should be used for the integration test

★ DEFINED\_PORT: Loads a WebServerApplicationContext and provides a real web

environment.

★ NONE: Loads an ApplicationContext by using SpringApplication but does not

provide any web environment