

# Spring Professional Exam Tutorial v5.0

## Question 38

## Question 38 - When do you want to use @SpringBootTest annotation?

You should use `@SpringBootTest` annotation whenever writing JUnit Integration Test for product that is using Spring Boot.

Spring Boot approach to Integration Testing simplifies it by eliminating requirement of application deployment or establishing connection to other infrastructure.

`@SpringBootTest` annotation enables Spring Boot specific features on top of Spring Test that are useful for testing, like:

- ▶ Automated Context creation through `SpringApplication` class
- ▶ Web Environment for Testing - Mocked or Embedded
- ▶ Mocked Bean Injection via `@MockBean` annotation
- ▶ Spy Injection via `@SpyBean` annotation
- ▶ Ability to customize created context with `@TestConfiguration` annotated classes
- ▶ Auto configurations for:
  - ▶ MVC Testing
  - ▶ JSON Testing
  - ▶ JPA Tests
  - ▶ JDBC Tests
  - ▶ Mongo Db Tests
  - ▶ and much more...

## Question 38 - When do you want to use @SpringBootTest annotation?

To use @SpringBootTest annotation, you will need to add @RunWith(SpringRunner.class) annotation on top of your test class first, this is required **only for JUnit 4**, for JUnit 5 @ExtendWith(SpringExtension.class) annotation is already contained in @SpringBootTest annotation.

Next you need to use @SpringBootTest annotation:

```
@RunWith(SpringRunner.class)
@SpringBootTest
public class HelloControllerTest {

    @Test
    public void shouldPerformSomeActionCorrectly() throws Exception {
        ...
    }
}
```

