Title: Why is an autowired controller always null in junit5 tests?
Post Body:

I'm trying to add some unit tests (using **JUnit5**) to my application. But trying to **autowire** a **controller** raises an **assertion error** because the controller is null.

Test class:

```
package com.mydomain.preview.web; import static org.assertj.core.api.Assertions.assertThat;  import com.mydomain.preview.web.r
```

Controller class:

```
package com.mydomain.preview.web.rest;  import org.springframework.stereotype.Controller; import org.springframework.web.bind.
```

pom.xml (irrelevant sections omitted for brevity):

```
    <dependency>          <groupId>org.springframework.boot</groupId>          <artifactId>spring-boot-starter-test</artifactId>
```

I followed this guide: https://spring.io/guides/gs/testing-web/

The error I'm getting is: `java.lang.AssertionError: Expecting actual not to be null`. Same error is raised for `mvn test, ./mvnw test` and running the test from IntelliJ IDEA IDE.

```
SpringBootApplication Class:  @SpringBootApplication @EnableConfigurationProperties({LiquibaseProperties.class, ApplicationPro
```

Accepted Answer: None
Highest Rated Answer:

I assume that @SpringBootTest doesn't find the classes that need to be tested. Try adding `@SpringBootTest(classes = {TestController.class})`