



# **VMware**

**EDU-1202**

**Spring Certified Professional 2022**

**QUESTION & ANSWERS**

## Question: 1

Domain: Boot Actuator

What will be the correct syntax to be used in .properties file to enable the shutdown endpoint

- A. endpoint.shutdown.enable=true
- B. management.endpoint.shutdown.enabled=true right
- C. management.endpoint.enabled.shutdown=true
- D. endpoint.shutdown.disabled=false

---

**Answer: B**

---

### Explanation/Reference:

Option A is incorrect, because the syntax for enabling endpoint used in .properties file is management.endpoint.<id>.enabled=true

Option B is correct, because the syntax for enabling endpoint used in .properties file is management.endpoint.<id>.enabled=true and

using it with replacing <id> with shutdown will enable this endpoint.

Option C is incorrect, because the syntax for enabling endpoint used in .properties file is management.endpoint.<id>.enabled=true

Option D is incorrect, because the syntax for enabling endpoint used in .properties file is management.endpoint.<id>.enabled=true

<https://docs.spring.io/spring-boot/docs/current/reference/html/production-ready-features.html>

<https://docs.spring.io/spring-boot/docs/current/actuator-api/htmlsingle/>

## Question: 2

Domain: Boot Testing

Is the following statement correct?

@WebMvcTest annotation will create an ApplicationContext that contains web components and all their dependencies.

- A. True
- B. False

---

**Answer: B**

---

### Explanation/Reference:

Option A is incorrect, because @WebMvcTest will create an ApplicationContext that contains only web components, other

components needed should be mocked and injected.

Option B is correct, because @WebMvcTest will not create an ApplicationContext that contains web components and all their

dependencies. Rather it will create an ApplicationContext that contains only web components, other components needed should be

mocked and injected.

<https://docs.spring.io/spring->

[boot/docs/current/api/org/springframework/boot/test/autoconfigure/web/servlet/WebMvcTest.html](https://docs.spring.io/spring-)

<https://spring.io/guides/gs/testing-web/>

### Question: 3

Domain: Boot Autoconfig

Which of the following annotations applied separately enables auto-configuration? Select all that apply

- A. @SpringBootApplication
- B. @ContextConfiguration
- C. @EnableAutoConfiguration
- D. @ComponentScan

---

**Answer: A,C**

---

### Explanation/Reference:

Option A is correct, because @SpringBootApplication is composed by three annotations - @Configuration, @EnableAutoConfiguration and @ComponentScan and the @EnableAutoConfiguration is responsible for enabling autoconfiguration in Spring Boot.

Option B is incorrect, because @ContextConfiguration defines class-level metadata that is used to determine how to load and

configure an ApplicationContext for integration tests.

Option C is correct, because @EnableAutoConfiguration enables auto-configuration of the Spring Application Context, attempting to

guess and configure beans that you are likely to need.

Option D is incorrect, because @ComponentScan configures component scanning directives for use with @Configuration classes.

<https://docs.spring.io/spring-framework/docs/current/javadoc-api/org/springframework/test/context/ContextConfiguration.html>

<https://docs.spring.io/spring->

[boot/docs/current/api/org/springframework/boot/autoconfigure/EnableAutoConfiguration.html](https://docs.spring.io/spring-)

<https://docs.spring.io/spring->

[boot/docs/current/api/org/springframework/boot/autoconfigure/SpringBootApplication.html](https://docs.spring.io/spring-)

<https://docs.spring.io/spring-framework/docs/current/javadoc->

[api/org/springframework/context/annotation/ComponentScan.html](https://docs.spring.io/spring-framework/docs/current/javadoc-api/org/springframework/context/annotation/ComponentScan.html)

### Question: 4

Domain: MVC

Which of the following statements are correct regarding @RequestMapping annotation? Select all that apply

- A. It can be used on classes annotated with @Controller
- B. @RequestMapping will only map to GET requests unless explicitly specified different.
- C. It has attribute "method" which can be used to customize requests methods that this method will map to.

D. We can specify the request parameters for the method using the attribute "requestParams"

---

**Answer: A,C**

---

### Explanation/Reference:

Option A is correct, because `@RequestMapping` can be used both at the class and at the method level. Used on classes annotated with `@Controller` all methods in that class annotated with `@RequestMapping` or any of its specializations would be using relative paths, meaning that the value provided in the `@RequestMapping` annotation will be used as a prefix for the URI of every request.

Option B is incorrect, because `@RequestMapping` maps to all Http methods unless specified different in its "method" attribute.

Option C is correct, because `@RequestMapping` has a "method" attribute which takes an array of type `RequestMethod`, to specify to which Http methods this method will map

Option D is incorrect, because we can specify the request parameters for the method using the attribute "params", not "requestParams".

<https://docs.spring.io/spring-framework/docs/current/javadoc-api/org/springframework/web/bind/annotation/RequestMapping.html>

### Question: 5

Domain: Boot Testing

Which of the following statements are correct, regarding `@SpringBootTest` annotation? Select all that apply

- A. It contains `@ExtendWith(SpringExtension.class)`
- B. It contains `@RunWith(SpringRunner.class)`
- C. It enables Spring Boot specific features on top of Spring Test
- D. It's used on test methods

---

**Answer: A,C**

---

### Explanation/Reference:

Option A is correct, because `@SpringBootTest` is meta-annotated with `@ExtendWith(SpringExtension.class)`

Option B is incorrect, because `@SpringBootTest` is not meta-annotated with `@RunWith(SpringRunner.class)`

Option C is correct, because `@SpringBootTest` enables Spring Boot specific features on top of Spring Test

Option D is incorrect, because `@SpringBootTest` is a class level annotation

<https://docs.spring.io/spring-boot/docs/current/api/org/springframework/boot/test/context/SpringBootTest.html>

<https://spring.io/guides/gs/testing-web/>

### Question: 6