

This set of Java Spring Multiple Choice Questions & Answers (MCQs) focuses on “Property Editors and TaskExecutors in Spring”.

1. Which of the following is used to convert property values to text Values?

- a) property Editor
- b) property setter
- c) property getter
- d) none of the mentioned

[View Answer](#)

Answer: a

Explanation: A property editor is a feature of the JavaBeans API for converting property values to and from text values.

advertisement

2. Which property editor is used to specify a URL String for a property of the URL type?

- a) java.net.\*
- b) java.io.\*
- c) java.net.URL
- d) none of the mentioned

[View Answer](#)

Answer: c

Explanation: Spring will automatically convert the URL string into a URL object and inject it into your property.

3. Which interface is used to create your own property editors?

- a) CustomEditorConfigurer
- b) RegisterCustomEditor
- c) PropertyEditorConfigurer
- d) None of the mentioned

[View Answer](#)

Answer: a

Explanation: The CustomEditorConfigurer is implemented as a bean factory post processor for you to register your custom property editors before any of the beans get instantiated.

4. Which of the following property editors are registered by Spring?

- a) CustomNumberEditor
- b) FileEditor
- c) CustomDateEditor
- d) CustomNetEditor

[View Answer](#)

Answer: b

Explanation: ClassEditor, FileEditor, LocaleEditor, and URLEditor are pre registered by Spring, so you don't need to register them again.

5. What are the ways to create custom Property Editors?

- a) implement PropertyEditor interface
- b) extend PropertyEditorSupport Class
- c) none of the mentioned
- d) all of the mentioned

[View Answer](#)

Answer: d

Explanation: You can write custom property editors by implementing the `java.beans.PropertyEditor` interface or extending the convenient support class `java.beans.PropertyEditorSupport`.

6. Method which converts property into a string value

- a) `getAsText`
- b) `setAsText`
- c) `regText`
- d) None of the mentioned

[View Answer](#)

Answer: a

Explanation: The `getAsText()` method converts a property into a string value.

advertisement

7. Method that converts string into a property value

- a) `getAsText`
- b) `setAsText`
- c) `regText`
- d) None of the mentioned

[View Answer](#)

Answer: b

Explanation: The `setAsText()` method converts a string back into a property.

8. The property value is retrieved by:-

- a) `getValue` method
- b) `setValue` method
- c) none of the mentioned
- d) all of the mentioned

[View Answer](#)

Answer: d

Explanation: The property value is retrieved and set by calling the `getValue()` and `setValue()` methods.

9. For a property editor to be searched correctly, it must be located in the same package as the target class, and the name must be Editor's name

- a) True
- b) False

[View Answer](#)

Answer: b

Explanation: The name must be target class name with Editor as its suffix.

10. Which package is used for periodic work

- a) java.lang.Thread
- b) java.util.TimerTask
- c) java.util.Timer
- d) java.util.concurrent

[View Answer](#)

Answer: b

Explanation: Java 1.3 saw the introduction of java.util.TimerTask to support doing some sort of work periodically.

11. Which subinterface provides functionality for managing Threads and their events

- a) ExecutorService
- b) ThreadService
- c) All of the mentioned
- d) None of the mentioned

[View Answer](#)

Answer: a

Explanation: ExecutorService, a subinterface, provides more functionality for managing threads and providing support for raising events to the threads, such as shutdown().

12. Method provided by ExectuorService which returns a Future < T >

- a) submit
- b) publish
- c) addService
- d) registerService

[View Answer](#)

Answer: a

Explanation: ExecutorService, a subinterface, provides more functionality for managing threads and providing support for raising events to the threads, such as shutdown().

advertisement

13. Which method provided by ExecutorService is used to check whether job is finished or cancelled

- a) Future.isDone()
- b) Future.isCancelled()
- c) None of the mentioned
- d) All of the mentioned

View Answer

Answer: d

Explanation: You can call Future.isDone() or Future.isCancelled() to determine whether the job is finished or cancelled, respectively.

14. The Quartz integration and message driven POJO container doesn't needs TaskExecutor Services

- a) True
- b) False

View Answer

Answer: b

Explanation: The TaskExecutor interface is used quite a bit internally in the Spring framework.

15. Which of the following class's instance is used by TimerTaskExecutor for managing jobs

- a) java.util.Timer
- b) java.util.Date
- c) java.util.HashMap
- d) none of the mentioned

View Answer

Answer: a

Explanation: TimerTaskExecutor uses a java.util.Timer instance and manages jobs (java.util.concurrent.Callable or java.lang.Runnable instances) for you by running them on the Timer.

### Sanfoundry Global Education & Learning Series – Java Spring.

To practice all areas of Java Spring, [here is complete set of 1000+ Multiple Choice Questions and Answers \(https://www.sanfoundry.com/1000-spring-questions-answers/\)](https://www.sanfoundry.com/1000-spring-questions-answers/).

« Prev Page - Spring Questions and Answers – Bean Post Processors (<https://www.sanfoundry.com/spring-questions-answers-bean-post-processors/>)

» Next Page - Spring Questions and Answers – AspectJ Annotation (<https://www.sanfoundry.com/spring-questions-answers-aspectj-annotation/>)

advertisement

### Deep Dive @ Sanfoundry:

1. **Java Programming Examples on Networking** (<https://www.sanfoundry.com/java-programming-examples-networking/>)
2. **Java Programming Examples on String Handling** (<https://www.sanfoundry.com/java-programming-examples-string-handling/>)
3. **Java Programming Examples on Classes** (<https://www.sanfoundry.com/java-programming-examples-classes/>)