Appendix

Version 6.0.0-M1

Table of Contents

1. Spring Properties 2

This part of the reference documentation covers topics that apply to multiple modules within the core Spring Framework.

Chapter 1. Spring Properties

`SpringProperties` is a static holder for properties that control certain low-level aspects of the Spring Framework. Users can configure these properties via JVM system properties or programmatically via the `SpringProperties.setProperty(String key, String value)` method. The latter may be necessary if the deployment environment disallows custom JVM system properties. As an alternative, these properties may be configured in a `spring.properties` file in the root of the classpath — for example, deployed within the application's JAR file.

The following table lists all currently supported Spring properties.

Table 1. Supported Spring Properties

Name	Description
`spring.beaninfo.ignore`	Instructs Spring to use the `Introspector.IGNORE_ALL_BEANINFO` mode when calling the JavaBeans `Introspector`. See `CachedIntrospectionResults` for details.
`spring.expression.compiler.mode`	The mode to use when compiling expressions for the Spring Expression Language.
`spring.getenv.ignore`	Instructs Spring to ignore operating system environment variables if a Spring `Environment` property — for example, a placeholder in a configuration String — isn't resolvable otherwise. See `AbstractEnvironment` for details.
`spring.index.ignore`	Instructs Spring to ignore the components index located in `META-INF/spring.components`. See Generating an Index of Candidate Components.
`spring.jdbc.getParameterType.ignore`	Instructs Spring to ignore `java.sql.ParameterMetaData.getParameterType` completely. See the note in Batch Operations with a List of Objects.
`spring.jndi.ignore`	Instructs Spring to ignore a default JNDI environment, as an optimization for scenarios where nothing is ever to be found for such JNDI fallback searches to begin with, avoiding the repeated JNDI lookup overhead. See `JndiLocatorDelegate` for details.
`spring.objenesis.ignore`	Instructs Spring to ignore Objenesis, not even attempting to use it. See `SpringObjenesis` for details.
`spring.test.constructor.autowire.mode`	The default *test constructor autowire mode* to use if `@TestConstructor` is not present on a test class. See Changing the default test constructor autowire mode.

Name	Description
`spring.test.context.cache.maxSize`	The maximum size of the context cache in the *Spring TestContext Framework*. See Context Caching.
`spring.test.enclosing.configuration`	The default *enclosing configuration inheritance mode* to use if `@NestedTestConfiguration` is not present on a test class. See Changing the default enclosing configuration inheritance mode.