



SPRING TESTING

Session 6

OBJECTIVES



- Explain testing of Spring applications
- Explain Agile software testing process
- Describe how to test Spring applications using Junit4
- Explain how to create a Spring MVC controller unit test

TESTING IN SPRING FRAMEWORK (1-2)



- Testing plays a crucial role in ensuring that a software application is fully functional in any customer environment.
- Testing process falls in following two categories:
 - ❑ Manual
 - ❑ Automated

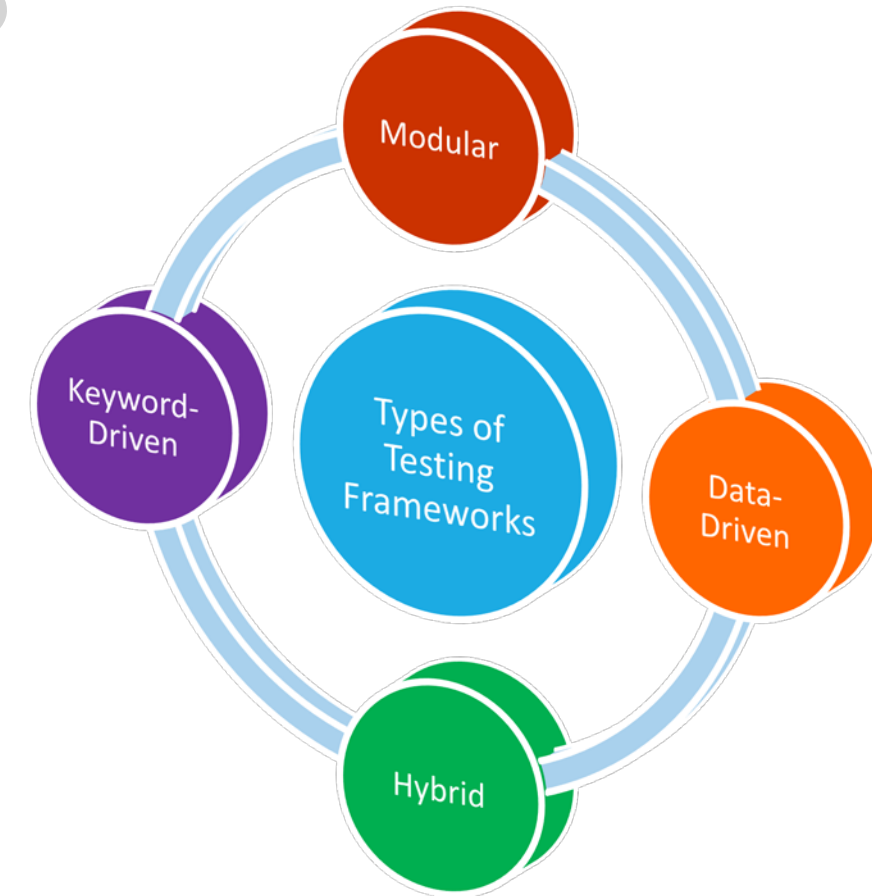
Functional	Performance	Security	Static
<ul style="list-style-type: none">• Core• GUI• API	<ul style="list-style-type: none">• Capacity• Load• Stress• Volume	<ul style="list-style-type: none">• Application vulnerabilities• Authentication• Authorization• White box security	<ul style="list-style-type: none">• Code review• Data verification• Design review• Document review

Types of Testing

TESTING IN SPRING FRAMEWORK (2-2)



- In today's world, testing phase is implemented practically by using a **testing framework**.
- A testing framework is an environment that is independent of the application being tested and can be easily maintained and customized.



Types of Testing

AGILE SOFTWARE TESTING METHODOLOGY (1-2)



- Agile methodology refers to the process in which requirements and solutions evolve through collaboration between cross-functional teams.
- These cross-functional teams include developers, testers, product managers, sales and marketing team, and the software documentation team.
- Agile methodology is based on 12 principles.

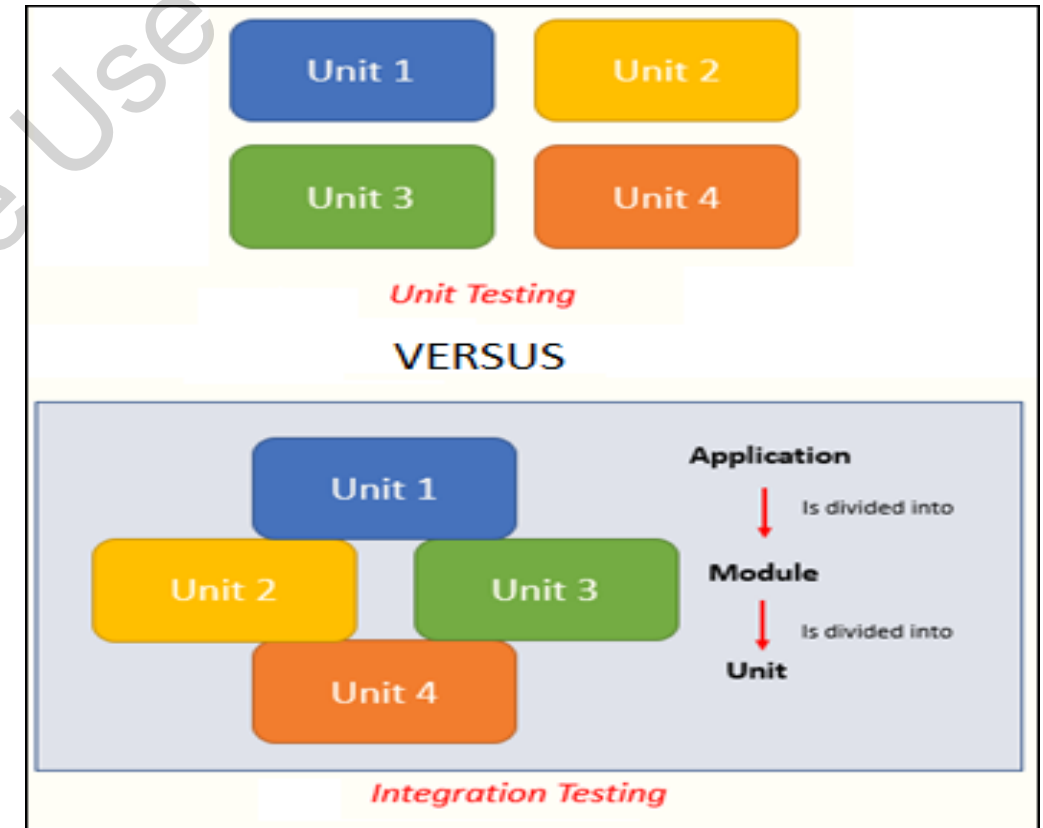
AGILE SOFTWARE TESTING METHODOLOGY (2-2)



- When testing is done within the Agile methodology framework, it includes following testing techniques:

- ☐ Unit testing

- ☐ Integration testing



Unit Testing Vs Integration Testing

UNIT TESTING OF SPRING-BASED JAVA APPLICATIONS (1-2)



- Unit testing:
 - Process of testing each individual method, class, and module is considered as units of an application
 - Commonly used unit testing framework is JUnit, (Java unit testing framework)
 - Testing process includes testing the both the state and behavior of a unit, which ensures a complete testing of the unit being tested.

UNIT TESTING OF SPRING-BASED JAVA APPLICATIONS (2-2)



- Stub is a dummy object that simulates a real object and stores pre-defined hard-coded data.
- Mock is an object that imitates an actual application object.



Tests whether the called method returns a correct value

Stubs used to verify an object's state



Tests whether the correct method was called

Mock objects used to verify an object's behavior

State and Behavior Testing

MOCKITO FRAMEWORK



- Mockito framework is an open source framework to create and configure mock objects for unit testing
- It can be downloaded from:

<http://mockito.org/>

OR

<https://code.google.com/p/mockito/>

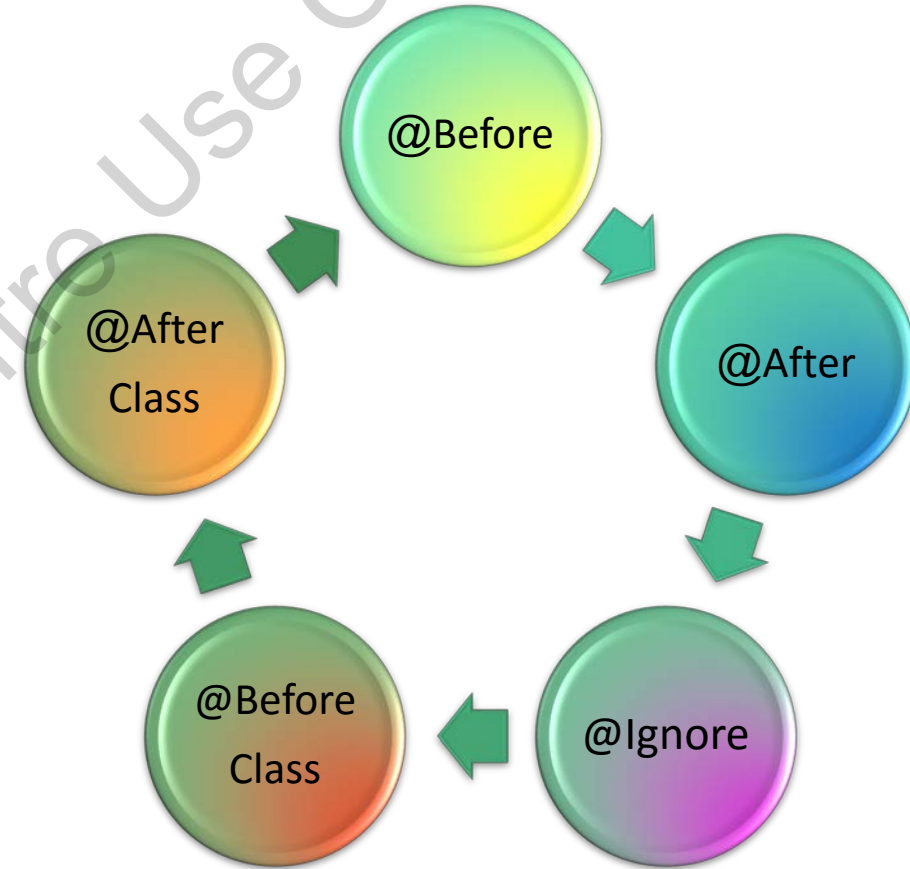
@Mock	@Spy	@InjectMocks	@RunWith (MockitoJUnitRunner.class)
Creates a mock object for an annotated field	Creates spies for the objects or the files it annotates	Represents instantiated private field on which injection will be performed	Creates mock and spy objects for all the fields annotated with @Mock or @Spy annotation

Mockito JAR Annotations

TESTING IN JUNIT4 – ANNOTATIONS



- JUnit4 is the commonly used unit testing framework for Java applications and uses annotations to create automated test cases for Java applications.
- JUnit4 uses the `@Test` annotation to annotate the methods that need to be tested.

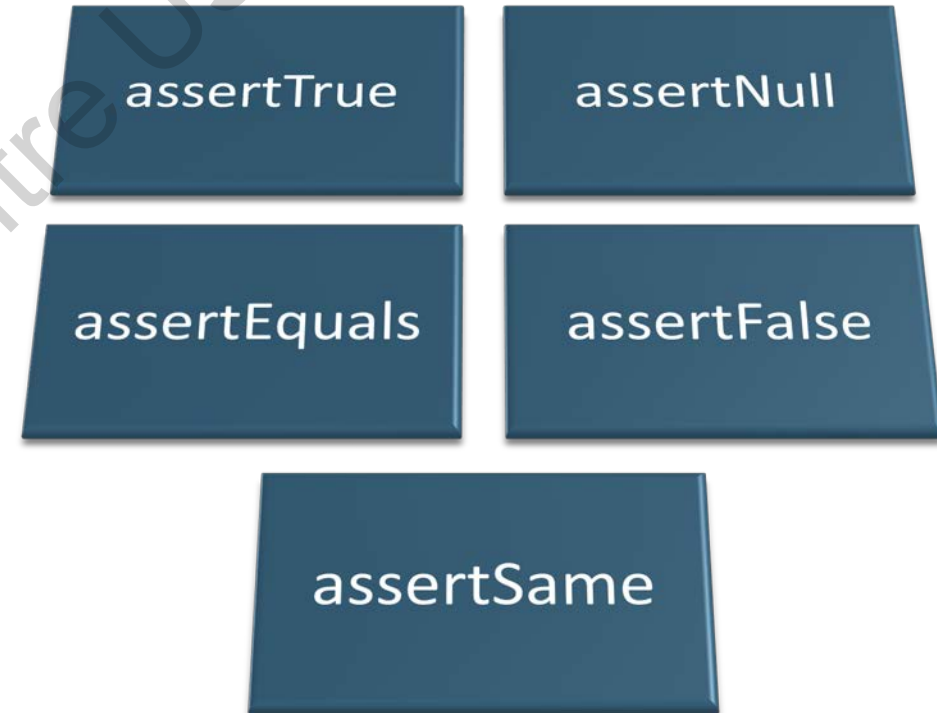


JUnit4 Annotations

TESTING IN JUNIT4 – ASSERT METHODS

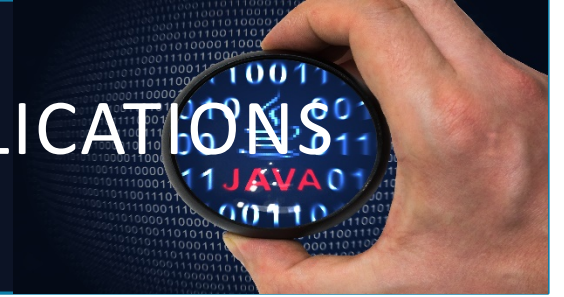


- JUnit4 uses assert methods that enable you to test specific conditions for the Java applications and compares the expected value with the value returned by the Test method.
- The assert methods are static methods that are declared in the org.junit.Assert class.



JUnit4 Assert Methods

INTEGRATION TESTING OF SPRING-BASED JAVA APPLICATIONS



- Guarantees that all individual units or modules interact correctly with each other
- Results in a fully functional and an error-free application
- Tests non-functional requirements such as performance, reliability, and so on, of entire application, with all units working together

UNDERSTANDING SPRING MVC TEST FRAMEWORK



- Spring MVC test framework

- Allows to test Spring MVC applications using the JUnit framework
- Is used to perform a complete integration testing of the Spring MVC applications through the Spring Configuration Loader

org.springframework

It supports unit and integration testing of Spring components and its Artifact ID is spring-test.

site.mockito.org

This is the library of Mockito mocking framework, and its Artifact ID is mockito-all.

JUnit

This is the library of the JUnit framework, and its Artifact ID is 'junit'.

Dependencies

SPRING MVC TEST FRAMEWORK - ANNOTATIONS



- Spring Framework provides following two annotations used to perform unit and integration testing with the TestContext framework:
 - ❑ @ContextConfiguration
 - ❑ @WebAppConfiguration

SPRING MVC TEST FRAMEWORK - MOCKMVC



- MockMvc is a key component of the Spring MVC Test framework.
- It is used to create test cases for Spring applications developed using Spring MVC.

`ContextMockMvcBuilder annotationConfigSetup(Class... configClasses)` - Used when application context is configured using Java configuration

`ContextMockMvcBuilder xmlConfigSetup(String... configLocations)` - Used when application context is configured using XML configuration files

`StandaloneMockMvcBuilder standaloneSetup(Object... controllers)` - Used when the test controller is to be configured manually

`InitializedContextMockMvcBuilder webApplicationContextSetup(WebApplicationContext context)` - Used when the WebApplicationContext object is fully initialized

MockMvcBuilder Static Methods

SPRING MVC TEST FRAMEWORK - `@RunWith(SpringJUnit4ClassRunner.class)`



- This JUnit annotation executes the tests included in a class that is annotated by the `@RunWith` annotation.
- It extends a class annotated by the `@RunWith` annotation by invoking the class which is passed as the parameter.

SUMMARY



- Testing is an essential part of SDLC that ensures a fully functional software application in any customer or external environment.
- Testing can be manual or automated.
- Agile methodology refers to the process in which requirements and solutions evolve through collaboration across cross-functional teams.
- Unit testing is the process of testing each individual method, class, module, which are considered units of an application.
- Integration testing is the testing of all the modules or units combined, to ensure a fully-functional application.
- JUnit4 is the commonly used unit testing framework for Java applications.
- The Spring MVC test framework allows you to test Spring MVC applications using the JUnit framework.