

FULL STACK



Automation Testing

FULL STACK

Unit Testing with JUnit



A Day in the Life of an Automation Test Engineer

Now, Marcelo has completed his technical project phase, he decided to do unit testing of his project for further concern and outputs.

To complete his project, Marcelo have to analyze and understand how the unit testing works with the help of software testing, what is the purpose of software test, what are the test terminologies work in unit testing, and some of the features and methods.

To achieve the above, he will learn a few concepts in this lesson that can help him to find a solution for the scenario.



Learning Objectives

By the end of this lesson, you will be able to:

- 👁 Understand the purpose of software testing and its key points
- 👁 Analyze and understand why the software tests are helpful, and much needed concern in the organization
- 👁 Acknowledge the test terminologies
- 👁 Illustrate the different approaches in tests terminologies



FULL STACK

Purpose of Software Test

Purpose of Software Test

These are the following statements for why software testing is required:



- A software test is a piece of software, which executes another piece of software.
- It validates if that code results in the expected state or executes the expected sequence of events.
- With software tests users ensure that certain parts of users software work as expected.

Purpose of Software Test



- These tests are typically executed automatically via the build system, so, these help the developer to avoid breaking existing code during development activities.
- JUnit is a Java unit testing framework that is one of the best test methods for regression testing.



Purpose of Software Test



- Testing is an important step in the development process.
- If the testing is done correctly, it can help detect early flaws in code which may be more difficult to find in later testing stages.
- The primary purpose of software testing is to detect flow at any step of the development process.

FULL STACK

Why are Software Tests Helpful?

Why are Software Tests Helpful ?



- Software unit tests help the developer verify that the logic of a piece of the program is correct.
- Running tests automatically helps to identify software regressions introduced by changes in the source code.
- Having a high tests coverage of the user's code allows you to continue developing features without having to perform lots of manual tests.
- The goal of software testing is to find errors, gaps, or missing requirements in comparison to the actual requirements.

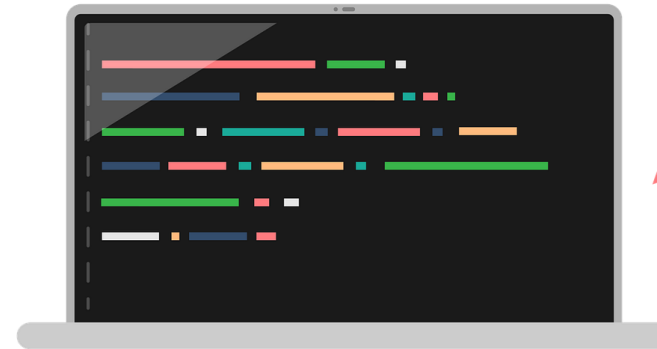
Why are Software Tests Helpful ?

These are some important aspects of software test which are necessary for every organization:

To gain customer confidence.

...

To check software adaptability



...

To accelerate software development

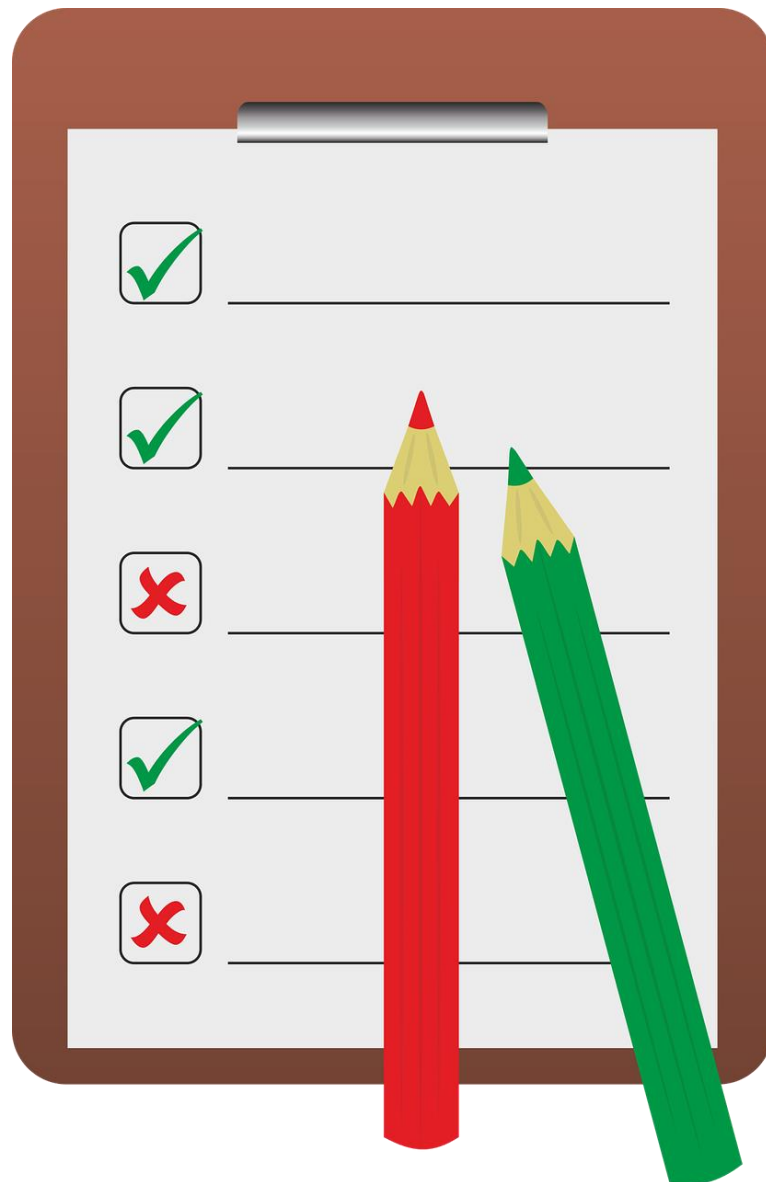
To avoid extra costs, avoid risks, and to optimize business

FULL STACK

Testing Terminologies

Testing Terminologies

Below are some methods to use the testing terminologies:



Code or application under test

Test fixture

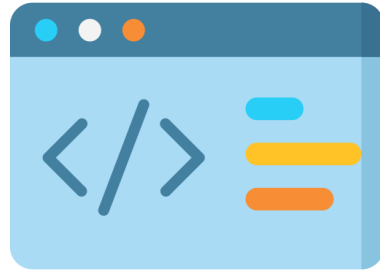
Unit tests and unit testing

Integration and performance tests

Testing frameworks for Java



Testing Terminologies



Code (or application) under tests: The code being tested is sometimes referred to as the code under test. This is referred to as the application under test while testing an application.



Test fixture: A test fixture is a fixed state in code that is used as test input. This is also known as a test pre-condition.

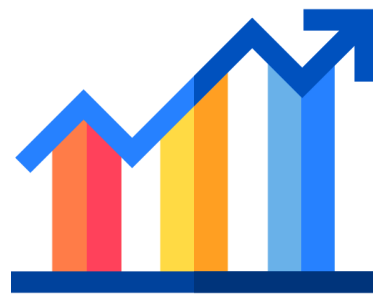


Unit tests and unit testing: A unit test is a piece of code developed by a developer that runs a specified functionality in the code under test and asserts a given behavior or condition.

Testing Terminologies



Integration tests: An integration test is designed to evaluate the behavior of a component or the integration of a group of components. The phrase functional test is sometimes used similarly with the integration test.



Performance tests: Performance tests are used to frequently analyze software components. Their goal is to verify that the code under test performs quickly enough even while under heavy demand.



Testing frameworks for Java: There are several testing frameworks available for Java. The most popular ones are JUnit and TestNG. This description focuses on JUnit. It covers both JUnit 4 and JUnit 5.

Key Takeaways

- A software test is a piece of software, which executes another piece of software.
- With software tests, users ensure that certain parts of their software work as expected.
- Test terminologies refer to individual units or components of the software that are tested.
- Code under test, test fixtures, unit testing, integration testing, testing the framework, all lay under the methods of test terminologies.

