**What is Testing**

Software testing is **the process of evaluating and verifying that a software product or application does what it is supposed to do**. The benefits of testing include preventing bugs, reducing development costs and improving performance.

**Types of Testing:**

**1.Manual Testing**

**2.Automation Testing**

In manual testing, a **human performs the tests step by** step, without test scripts. In automated testing, tests are executed automatically via test automation frameworks, along with other tools and software.

Unit Tests vs. Integration Tests

[Integration testing](https://www.javatpoint.com/integration-testing), we can test multiple components at once.

These components can be anything in our code, such as functions, classes and module that we have written.

But there is a limitation of the integration testing; what if an integration test doesn't give the expected result. In this situation, it will be very hard to recognize which part of the system is falling.

That's why we consider unit testing to get to know the exact problem in the tested code.

Unit testing is a smaller test, it checks a single component that it is working in right way or not. Using the unit test, we can separate what necessities to be fixed in our system.

**What is Unit Testing?**

Unit Testing is the first level of software testing where the smallest testable parts of a software are tested. This is used to validate that each unit of the software performs as designed.

**What is Python automation testing?**

Automated testing is **the execution of your test plan** (the parts of your application you want to test, the order in which you want to test them, and the expected responses) by a script instead of a human. Python already comes with a set of tools and libraries to help you create automated tests for your application.

**Test Automation Tools with This Ready-To-Use Python Environment**

* pytest.
* robot.
* selenium.
* selenium-firefox.
* selenium-helpers.
* setuptools 51.2.

Which automation tool is written in Python?

**Unittest or PyUnit** is the standard test automation framework for unit testing that comes with Python

[Python](https://www.javatpoint.com/python-tutorial) provides the **unittest module** to test the unit of source code.

* **test case:**  
  A test case is a set of conditions which is used to determine whether a system under test works correctly.
* **test suite:**  
  Test suite is a collection of testcases that are used to test a software program to show that it has some specified set of behaviours by executing the aggregated tests together.
* **test runner:**  
  A test runner is a component which set up the execution of tests and provides the outcome to the user.

**import unittest**

**class SimpleTest(unittest.TestCase):**

**# Returns True or False.**

**def test(self):**

**self.assertTrue(True)**

**if \_\_name\_\_ == '\_\_main\_\_':**

**unittest.main()**

Here, in the output the “.” on the first line of output means that a test passed.  
“-v” option is added in the command line while running the tests to obtain more detailed test results.

**Outcomes Possible:**  
There are three types of possible test outcomes:

* OK – This means that all the tests are passed.
* FAIL – This means that the test did not pass and an AssertionError exception is raised.
* ERROR – This means that the test raises an exception other than AssertionError.

<https://www.fleekitsolutions.com/api-automation-using-python/>

<https://www.wwt.com/article/rest-api-integration-testing-using-python>