In this lecture I’m going to go over some of the most useful PyTest command line arguments.

* By default Pytest runs all tests that it finds in the current working directory and sub-directory using the naming conventions for automatic test discovery.
* There are several pytest command line arguments that can be specified to try and be more selective about which tests will be executed.
* You can simply pass in the module name to execute only the unit tests in one particular module.
* You can also simply pass in a directory path to have pytest run only the tests in that directory.
* You can use the “-k” option to specify an evaluation string based on keywords such as: module name, class name, and function name.

You can use the “-m” option to specify that any tests that have a “pytest.mark” decorator that matches the specified expression strin

Additional Useful Command Line

Arguments

* -v : Report in verbose mode .

•

-q : Run in quiet mode (can be helpful when running hundreds or

thousands of tests at once).

* -s : Don’t capture console output (show print statements on the console).

•

—ignore : Ignore the specified path when discovering tests.

* —maxfail : Stop after the specified number of failures.
* Here are some additional command line arguments that can be very useful.
* The -v option specifies that verbose output from pytest should be enabled.
* The -q option specifies the opposite. It specifies that the tests should be run quietly (or minimal output). This can be helpful from a performance perspective when you’re running 100’s or 1000’s of tests.
* The -s option specifies that PyTest should NOT capture the console output.
* The —ignore option allows you to specify a path that should be ignore during test discovery.
* The —maxfail option specifies that PyTest should stop after N number of test failures.