# AutoTest Studio Getting started tutorials 6: Logs and Reports

Logs and reports are one of the important criteria for judging whether an automated testing framework is excellent. AutoTest Studio also has a complete log module and report module. Through these logs and reports, you can evaluate and analyze the details of the automated test task operation.

**Log**

AutoTest Studio has four levels of logs, namely DEBUG, INFO, WARN, ERROR, these four levels of logs, users can flexibly apply according to their actual situation, and can decide which levels of logs to display according to their needs. In practical applications, we can output all logs when developing and debugging test cases, but when the task is running, selectively output some logs to reduce the amount of log output.

Take a piece of test code as an example to introduce how to output the log and configure the displayed log level.

from autotest import \*

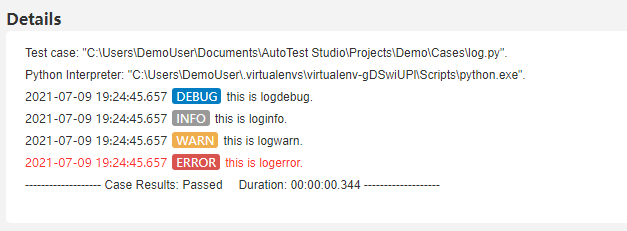
LogDebug("this is logdebug.")

LogInfo("this is loginfo.")

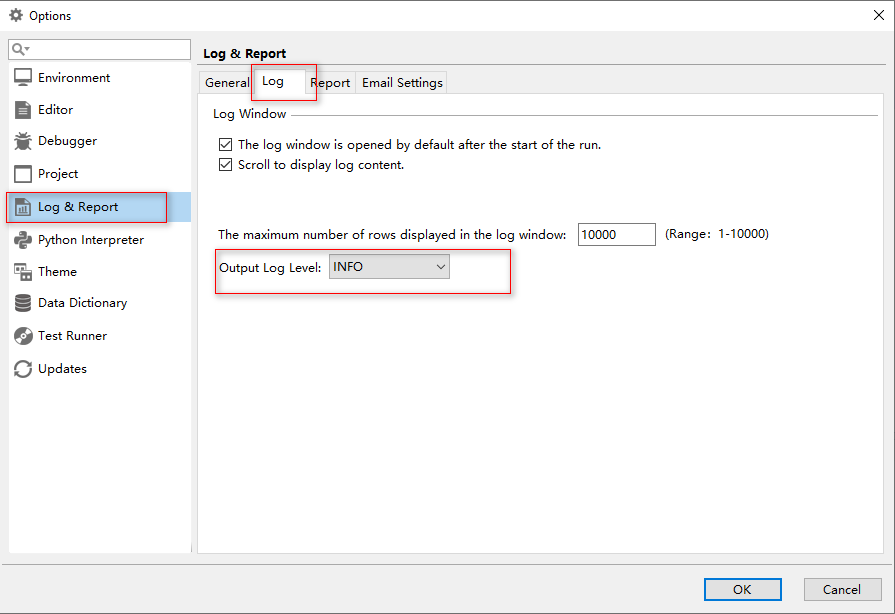
LogWarn("this is logwarn.")

LogError("this is logerror.")

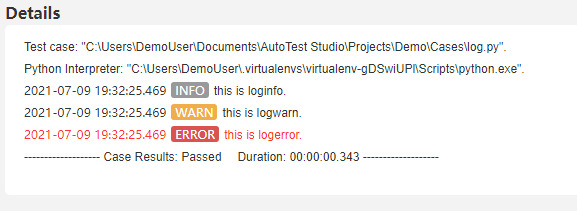
After normal operation, AutoTest Studio will display four levels of logs by default.



In the configuration manager, we can customize the output log level. For example, we only want to output INFO and above logs. The configuration is shown in the figure below.



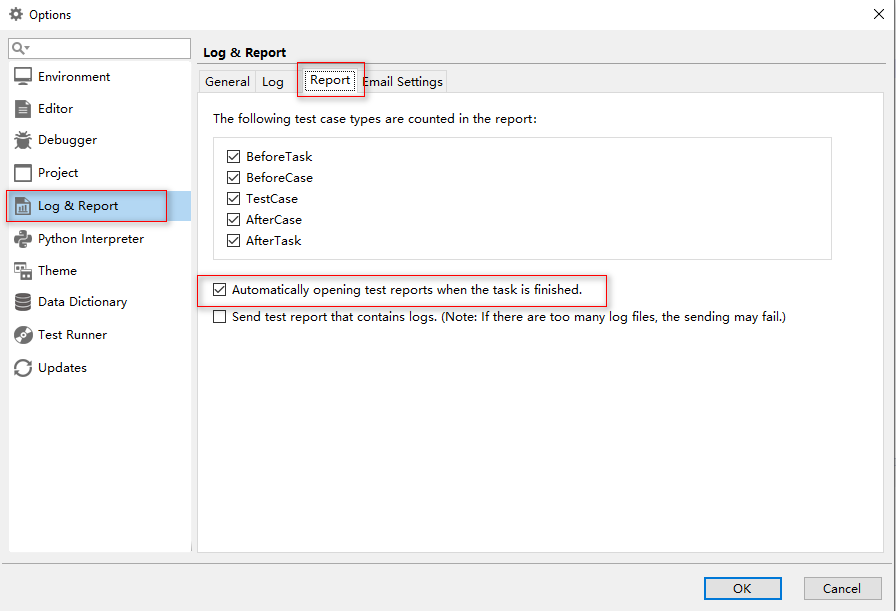
After re-running the code, the output log is as follows, you can see that there is no DEBUG level log.



**Report**

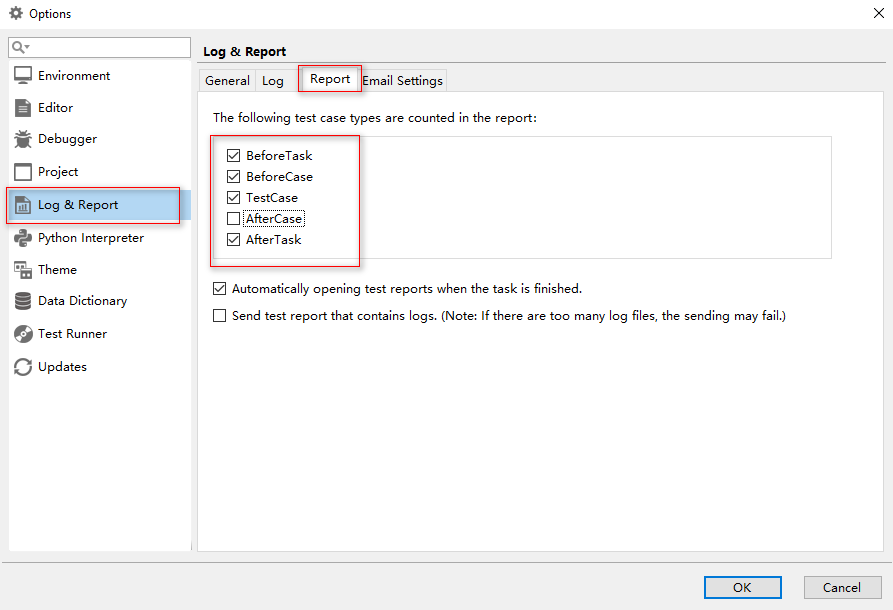
AutoTest Studio takes tasks as the running unit, and each run is a task, even if you only run one test case.

After the task is completed, the test report of this task will be automatically opened by default. Of course, you can also set not to open the test report automatically. The setting method is as follows.

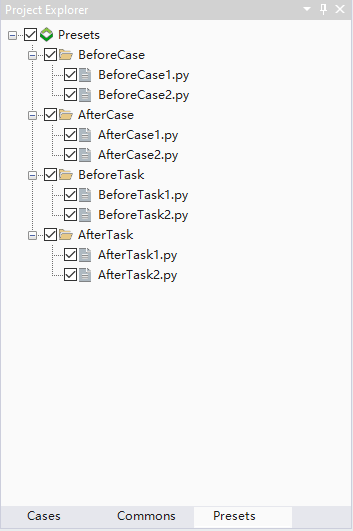


**Set report statistics type**

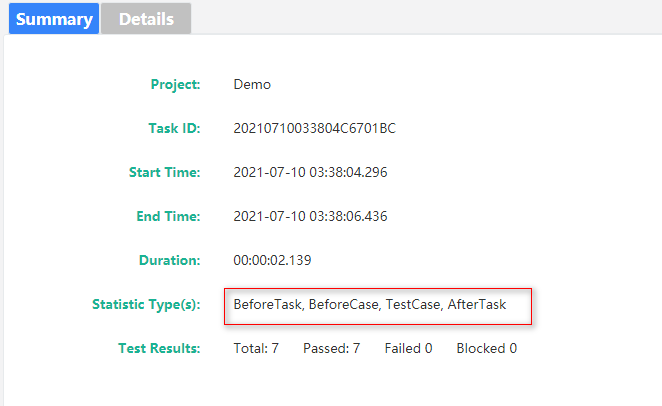
The user can decide the use case type of the report statistics according to the needs, but it must be set before the task is run. The settings are as follows. For comparison, we assume that the AfterCase type is not counted.



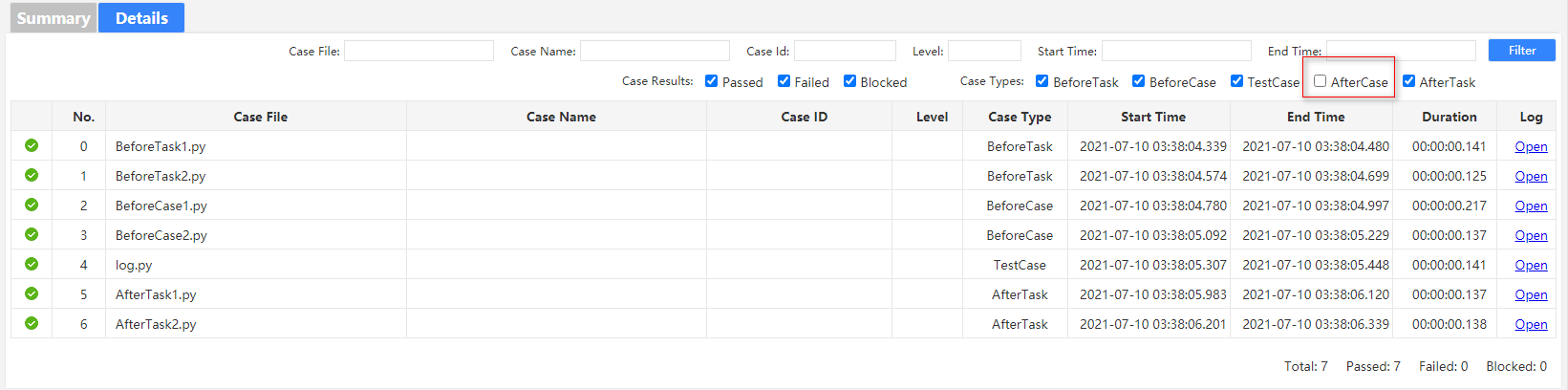
**Select a preset use case in "Presets".**



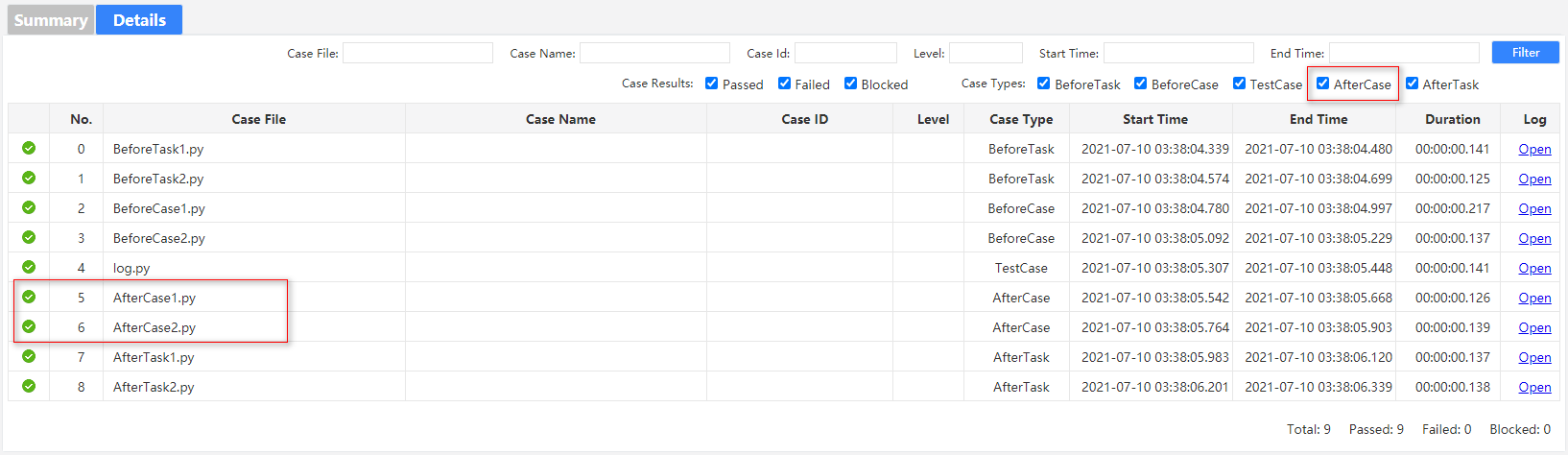
Run the task once, as shown in the figure below, in the statistical report, there is no AfterCase.



In the test log record, the default is the same as the report statistics, and the types without statistics will not be displayed, as shown in the figure below.

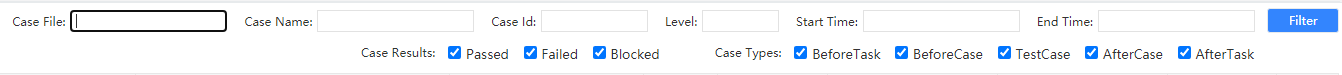


But this does not mean that the log of this type of AfterCase is lost. You can select the type without statistics in the filter and filter again.

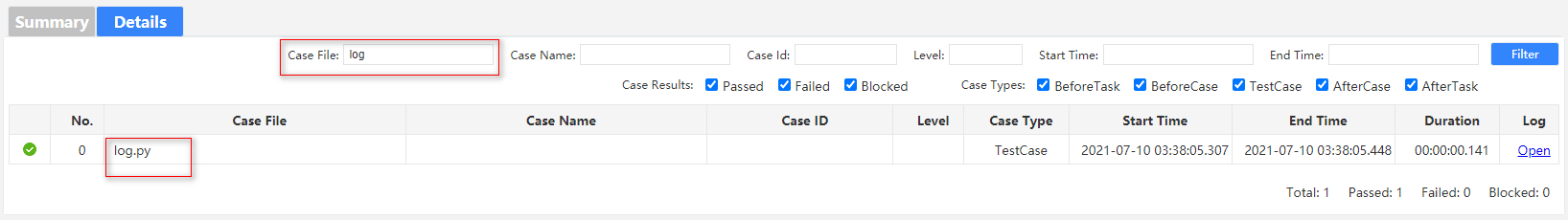


**Log filtering**

In the test report, the user can filter the log records through a variety of conditions. The filter conditions are as follows.



Suppose we want to filter the use cases that contain the "log" keyword in CaseFile.



**Historical report**

In the toolbar "**Tools**" -> "**Reports**", you can open the history report list. With this tool, users can open the task reports that have been run before, or export one or more task reports in the form of a compressed package through the Export button.

