

Using Variables

The following table shows variables that can be used in settings values.



We recommend you avoid using spaces, +, /, or any other special characters when setting variables or values for configuration settings, as some API calls may require properly encoded URLs.

Variable	Description	Example
analysis_type	Outputs a comma separated list of enabled analysis types (e.g., Static, Generation and Execution)	<code>\${analysis_type}</code>
env_var	Outputs the value of the environmental variable specified after the colon.	<code>\${env_var:HOME}</code>
config_name	Outputs the name of executed Test Configuration.	<code>\${config_name}</code>
dtp_project	Outputs the name of DTP project specified in the settings file using <code>dtp.project</code> option.	<code>\${dtp_project}</code>
project_module	Outputs the name of the tested project's module. If more than one module is provided as an input, the first tested module name is output followed by an ellipsis (...). The variable can be configured in the settings file with the <code>project.module</code> option.	<code>\${module_name}</code>
host_name	Outputs the name of the host.	<code>\${host_name}</code>
user_name	Outputs the name of the current user.	<code>\${user_name}</code>
os	Outputs the name of the operating system.	<code>\${os}</code>
arch	Outputs the name of the operating system architecture	<code>\${arch}</code>
exec_env	Outputs information about the execution environment. By default, this variable outputs a combination of the <code>\${os}</code> and <code>\${arch}</code> variables. It can be customized by configuring the <code>exec.env</code> option in the <code>.properties</code> file (see Report Settings).	<code>\${exec_env}</code>
scontrol_branch	Outputs the source control branch name for the tested project. If more than one branch name is detected, the first branch name is output followed by an ellipsis (...). The variable can be configured in the settings file with the <code>scontrol.branch</code> option.	<code>\${scontrol_branch}</code>
tool_name	Outputs the name of the tool (i.e., Jtest, C++test, dotTEST).	<code>\${tool_name}</code>
jvm_prop	Outputs the value of the Java vm property specified after the colon.	<code>\${jvm_prop:os.name}</code>