

# CLion 2017.3 RC, build 173.3727.18

<b>No subsystem</b>	Bug	<a href="#">CPP-11115</a>	Access is allowed from event dispatch thread only.
	Bug	<a href="#">CPP-11074</a>	Valgrind parse DIE
	Bug	<a href="#">CPP-11129</a>	Preserve tree selection during Valgrind execution
	Bug	<a href="#">CPP-10999</a>	Valgrind integration: handle clashing parameters
	Cosmetics	<a href="#">CPP-11051</a>	Valgrind integration: Underscore disappears from target name
<b>Code: Highlighting and Inspections</b>	Bug	<a href="#">CPP-9802</a>	Cyrillic symbols break Clang-tidy suggestions
	Bug	<a href="#">CPP-10530</a>	ClangTidy: java.lang.IllegalStateException: Compiler is not found exception after rename .cpp file to .c
<b>Code: Parsing</b>	Bug	<a href="#">CPP-7461</a>	Better recovery from parsing errors in typedefs
<b>Code: Resolving</b>	Bug	<a href="#">CPP-10995</a>	False ambiguous error on tuple initialization
<b>Running</b>	Bug	<a href="#">CPP-11062</a>	Run configurations cease to seem broken after using run icon near main function
	Bug	<a href="#">CPP-11061</a>	Run configuration doesn't show broken icon when it can't be to run
<b>Diff_Merge</b>	Usability	<a href="#">IDEA-171250</a>	New Auto Resolving conflicts must not automatically be applied
<b>Editor. Editing Text</b>	Bug	<a href="#">IDEA-180641</a>	Expand selection works incorrectly on Java String with slash escaping on last position
	Bug	<a href="#">IDEA-179710</a>	Wrong paste from history
	Usability	<a href="#">IDEA-180879</a>	Parameter popup is absolutely incomprehensible
<b>Find, Replace, Find Usages</b>	Bug	<a href="#">IDEA-181024</a>	Replace in path broken
<b>User Interface</b>	Exception	<a href="#">IDEA-181936</a>	Opening 'Find In Path' from 'Find Class' throws exception
<b>Version Control</b>	Exception	<a href="#">IDEA-181922</a>	NPE from VCS
<b>Code Insight</b>	Bug	<a href="#">PY-26243</a>	Unresolved reference in dunder all for not imported modules
	Bug	<a href="#">PY-26801</a>	PyCharm creates __init__.py in every folder to project path on new package creation action
	Bug	<a href="#">PY-25751</a>	If "__all__" is used in "__init__.py" PyCharm doesn't understood type of imported package
	Exception	<a href="#">PY-26006</a>	SOE if function is decorated with the same name decorator
	Exception	<a href="#">PY-26028</a>	Code insights fail - StackOverflowError
	Exception	<a href="#">PY-26847</a>	Pycharm 2017.2 insane logging - freezing
<b>Console</b>	Bug	<a href="#">PY-26603</a>	Exception when stopping Python Console
<b>Debugger</b>	Usability	<a href="#">PY-26883</a>	Show a full name of interpreter when Attaching to process on Windows
<b>Inspections</b>	Bug	<a href="#">PY-8933</a>	Import in try block should also be defined in except block: false positive when name not used
	Usability	<a href="#">PY-3318</a>	Add all words from django settings.py to spell-checker vocabulary
	Exception	<a href="#">PY-26061</a>	Specific code pattern causes inspections to hang
	Exception	<a href="#">PY-25994</a>	Code insight failures
<b>Interpreters Settings</b>	Bug	<a href="#">PY-26749</a>	Can't create pyramid project if new venv is going to be created inside project
	Bug	<a href="#">PY-26657</a>	don't install Flask package in Flask project creation
	Bug	<a href="#">PY-26594</a>	Manually added paths to local interpreter get lost
<b>Skeletons</b>	Bug	<a href="#">PY-26751</a>	Pregenerated skeletons get deleted if SDK path does not exactly match