Composer Help

Run Menu

12/12/2025

Run Menu

- See the Debugging voice applications and Debugging routing applications topics for supported functionality.

The Run Menu contains all of the actions required to run, debug, step through code and work with breakpoints. Different parts of the menu are visible at different times, as each perspective can be customized to show only specific capabilities. The Run menu contains the following items:

Resume	Resumes execution of the currently selected Debug target.
Suspend	Halts the execution of the currently selected thread in a debug target. Once the selected thread is suspended, you can then examine it.
Terminate	Terminates the selected debug target.
Step Into	[Disabled for GVP Debugger]
Step Over	Steps over the highlighted statement. Execution will continue at the next line either in the same method or (if you are at the end of a method) it will continue in the method from which the current method was called. The cursor jumps to the declaration of the method and selects this line.
Step Return	[Disabled for GVP Debugger]
Run to Line	[Disabled for GVP Debugger]
Use Step Filters (Shift+F5)	Toggles step filters on and off. When on, all step functions apply step filters.
Run	Re-launches the most recently launched application, or launches the selected resource or active editor depending on the launch operation preference settings found on the Run/Debug > Launching preference page.
Debug	Re-launches the most recently launched application under debugger control, or launches the selected resource or active editor depending on the launch operation preference settings found on the Run/Debug > Launching preference page.
Run History	Displays a submenu of the recent history of launch configurations launched in run mode
Run As	When a callflow is selected, displays Run Callflow. In the Run mode, call traces are provided and the application continues without any breakpoints. Note: Run on Server is an Eclipse feature and is not used by Composer.
Run Configurations	Used for debugging callflow diagrams. Opens the

	Run Configurations dialog box that lets you create, manage, and run launch configurations of different types.
Debug History	Displays a submenu of the recent history of launch configurations launched in debug mode.
Debug As	Displays a sub menu of registered debug launch shortcuts. Launch shortcuts provide support for workbench or active editor selection sensitive launching. Note: Debug on Server is an Eclipse feature and is not used by Composer.
Debug Configurations	Used for debugging callflow diagrams. Opens the Debug Configurations dialog box that lets you create and modify launch configurations and debug applications.
External Tools	Displays external tools that allow you to configure and run programs, batch files, Ant buildfiles, and others using the Workbench. You can save these external tool configurations and run them at a later time. Output from external tools is displayed in the console view. Selecting External Tools presents the following sub-menus: Run As, External Tool Configurations, Organize Favorites.
Create URL Breakpoint	Creates a breakpoint, which suspends the execution of a workflow at the location where the breakpoint is set.
Toggle Breakpoint	Appears in Debugging perspective. Select to suspend the execution of a program at a particular location in a callflow. When a breakpoint is encountered during execution of a program, the program suspends and triggers a SUSPEND debug event with BREAKPOINT as the reason.
Toggle Line Breakpoint	Select to set a breakpoint on an executable line of a program.
Toggle Method Breakpoint	Use when working with types that have no source code (binary types). Open the class in the Outline View, and select the method where you want to add a method breakpoint. Select Toggle Method Breakpoint to have a breakpoint appear in the Breakpoints View. If source exists for the class, then a breakpoint also appears in the marker bar in the file's editor for the method that was selected. While the breakpoint is enabled, thread execution suspends when the method is entered, before any line in the method is executed.
Toggle Watchpoint	Appears in GVP Debugging perspective. You must select a Java field object to use this command. Use after you have created a watchpoint on the currently selected field. Whenever that field is accessed or modified, execution will be suspended. If the selected field already has a watchpoint, selecting this command will remove it.

Skip All Breakpoints	Select to mark all breakpoints in the current view as skipped. Breakpoints marked as skipped will not suspend execution.
Remove All Breakpoints	Select to remove all breakpoints from the Breakpoints View.