

How to Use the Eclipse Debugger

Eclipse, like all IDEs, provides a built-in, interactive debugger.

With the interactive debugger you can perform tasks such as:









1. Set breakpoints in the code so you can “watch” it execute.
2. Examine and change variables.
3. Skip over code; Alter the logic flow of the code.
4. Set Watches to always display in-scope variables regardless of where in the code you stop.
5. Set conditional breakpoints.
6. Evaluate expressions coded in the program or those you’d like to test.

For details on all you can do with the Eclipse Debugger please visit the Eclipse web site:

https://www.eclipse.org/community/eclipse_newsletter/2017/june/article1.php#:~:text=The%20debug%20perspective%20offers%20additional,and%20provides%20operations%20on%20that

To start the Eclipse debugger on your code, simply click the  icon in the Eclipse toolbar.

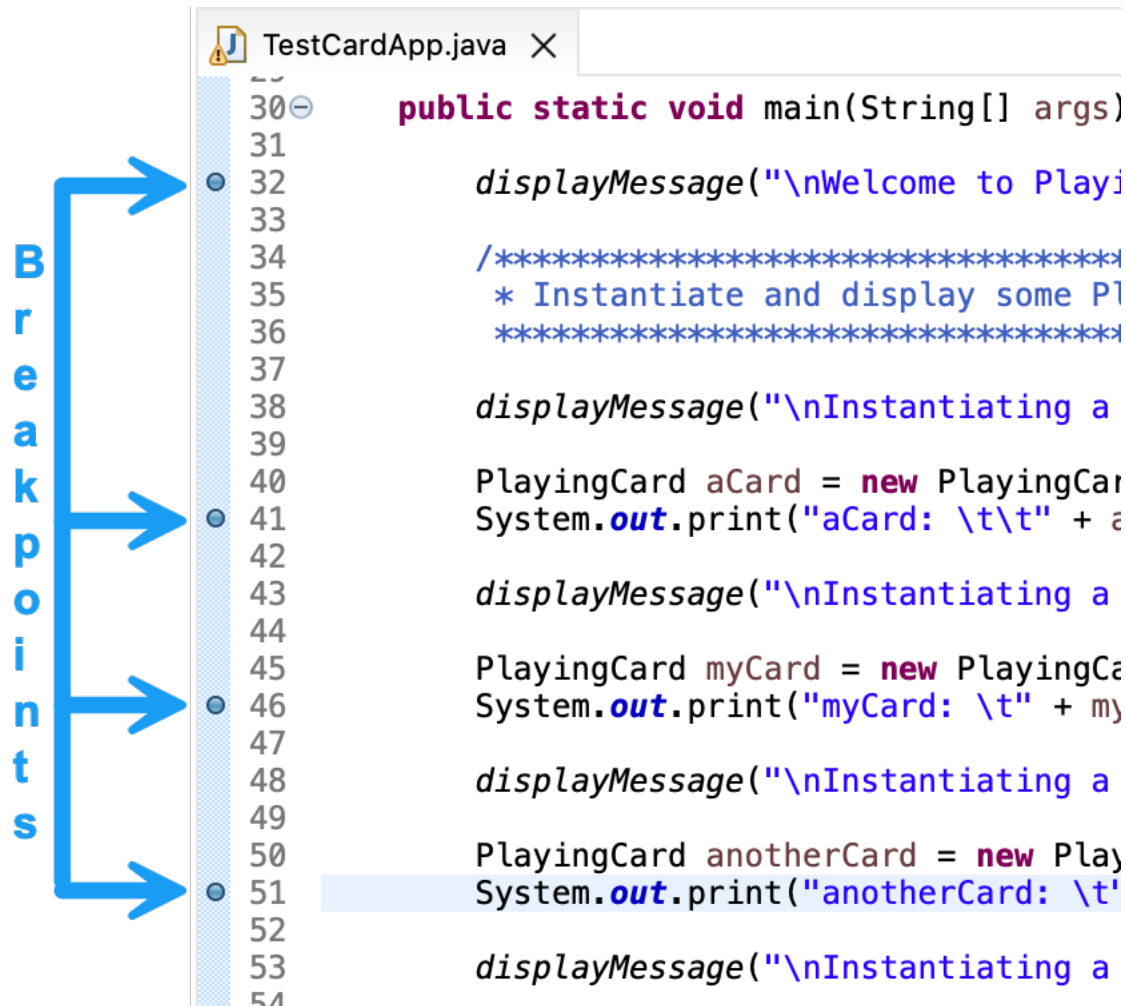
Once you are in debug, at a breakpoint, the following are available to resume execution:

		
F8 Stop F5 F6 F7		
Shortcut	Toolbar	Description
F5 (Step Into)		Steps into the call
F6 (Step Over)		Steps over the call
F7 (Step Return)		Steps out to the caller
F8 (Resume)		Resumes the execution
Ctrl + R (Run to Line)		Run to the line number of the current caret position
Drop to Frame		Rerun a part of your program
Shift + F5 (Use Step Filters)		Skipping the packages for Step into
Ctr + F5 / Ctrl + Alt + Click		Step Into Selection

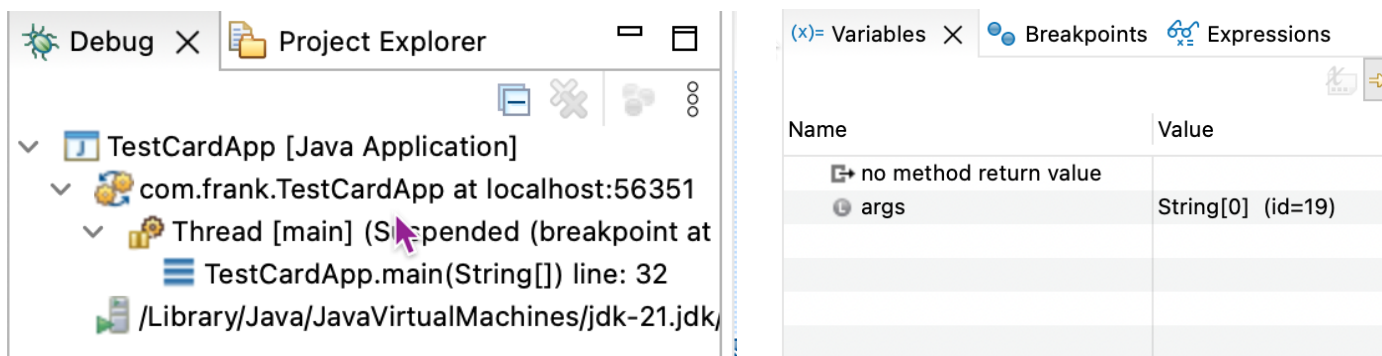
How to Use the Eclipse Debugger

Before you start debug for your code, you should set breakpoints at which you can suspend the execution of the code and interact with the debugger.

To set (or unset) a breakpoint, simply double click in the area to the left of line number at which you want the breakpoint set. A circle will appear to indicate a breakpoint has been set on the line.



Once execution encounters a breakpoint, program execution is suspended, and you are place in the Debug perspective of Eclipse:



How to Use the Eclipse Debugger

Once you are in the Debug Perspective with execution suspended you may perform any debugging you deem appropriate: Examine variables, change variables, evaluate expressions, add/remove watchpoints, add/remove breakpoints, et al.

When you are finished debugging, either terminate the execution by clicking the “little red box” in the tool bar or clear all breakpoints and let the program run to the end.

To return your workspace to the “Java Perspective” either click the “Java Perspective” icon in the upper right-hand corner of the screen or choose **Window → Open Perspective → Java Browsing** from the menu bar.