

Useful Eclipse features? [closed]

Asked 16 years, 4 months ago Modified 10 years, 9 months ago

Viewed 24k times



68



Closed. This question is [opinion-based](#). It is not currently accepting answers.



Want to improve this question? Update the question so it can be answered with facts and citations by [editing this post](#).

Closed 11 years ago.

[Improve this question](#)

I have been using Eclipse as an IDE for a short amount of time (about 3 months of full use) and almost every day I learn about some shortcut or feature that I had absolutely no idea about. For instance, just today I learned that

`Ctrl` + `3` was the shortcut for a Quick Access window.

I was wondering what your most useful/favorite Eclipse features are. With the IDE being so big, it would be helpful to learn about the more commonly used parts of the program.

eclipse

Share

edited Nov 12, 2010 at 6:45

Improve this question

Follow

community wiki

2 revs, 2 users 80%

Adam Mika

-
- 1 Linked: stackoverflow.com/questions/54886/hidden-tricks-for-eclipse – Gnoui Jan 6, 2011 at 9:10
-

23 Answers

Sorted by:

Highest score (default)



My most commonly used features are

73

`ctrl` + `1` quick-fix / spell-checker



opening files

- `ctrl` + `shift` + `t` load class file by classname
- `ctrl` + `shift` + `r` load any file by filename









matches are made on the start of the class/filename. start your search pattern with a * to search anywhere within the filename/classname.

Formatting


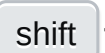


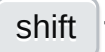


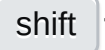

- `ctrl` + `shift` + `f` Format source file (set up your formatting style in Window | preferences | java | code style | formatter)

-  +  +  Organise imports


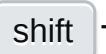

Generated code


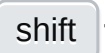



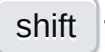



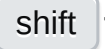


-  +  ,  to generate getters and setters
-  +  ,  to insert method signatures for overridden methods from superclass or interface


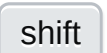

Refactorings







-  +  +  Extract text-selection as local variable (really handy in that it determines and inserts the type for you).
-  +  +  Extract text-selection as a method
-  +  +  inline selected method

Running and debugging.

 +  +  is a really handy prefix to run stuff in your current file.

-  +  +  ,  run unit tests in current file
-  +  +  ,  run main in current file
-  +  +  ,  run on server

There are more. The options are shown to you in the lower-right popup after hitting  +  + .






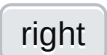
 +  +  can be switched for  +  +  in all the above examples to run in the debugger.

Validation

As of the recent Ganymede release, you can now switch of validation in specified files and folders. I've been waiting for this feature for ages.

- Go to Project | Properties | Validation
- click on the ... button in the settings column of the validator you want to shut up
- Add a rule to the *exclude group*

code navigation

- hold down  to make all variables, methods and classnames hyperlinks to their definitions.
-  +  to navigate back to where you clicked

-  +  to go "forwards" again

Share Improve this answer

edited Mar 22, 2013 at 10:23

Follow

answered Aug 21, 2008 at 17:39



[Cheekysoft](#)

35.6k ● 20 ● 74 ● 86



This is cool: If someone has emailed you a stack trace, you can copy and **paste the stack trace into Eclipse's**



Console window. You can then click on class names in the stack trace as if your own code had generated it.



Share Improve this answer

answered Aug 21, 2008 at 19:55

Follow



[David Koelle](#)

20.8k ● 23 ● 94 ● 130

5 Just to note, I had to enable this by clicking the "open console" icon on the far right of the console, then selecting the "Java Stack Trace Console". – [rescdsk](#) Sep 29, 2011 at 20:50



15



A shortcut that I use everyday is `Ctrl` + `K`. In your editor (not only Java file), simply select a text (like a variable, a function, etc.), and then use this shortcut to go to the next occurrence of this text in the current editor. It's faster than using the `Ctrl` + `F` shortcut...



Note also that you can use `Ctrl` + `Shift` + `K` to search backwards.

Share Improve this answer

answered [Oct 12, 2010 at 21:02](#)

Follow

community wiki

[Romain Linsolas](#)

Nice one, only Ctrl and K are too far away for left hand (while right selecting with mouse) – [serg](#) Oct 12, 2010 at 21:11

Yes, indeed. But this shortcut is really useful also when you want to manually replace a short code snippet.

– [Romain Linsolas](#) Oct 12, 2010 at 21:18



Ctrl - **Alt** (**up** / **down**) Copy selected line(s) above or below current line.

15

Alt (**up** / **down**) Move current (or multiple selected) lines up or down



Ctrl - **Shift** - **R** Bring up the resource window, start typing to find class/resource



Ctrl - **O** Bring up all methods and fields for the current class. Hitting it again will bring up all methods and fields for current class and super classes.



Ctrl - **/** or **Ctrl** - **Alt** - **C** Comment single or multiple lines with //

Ctrl - **Shift** - **/** Comment selected lines with /* */

Ctrl - **.** Take you to the next error or warning line

Share Improve this answer

edited Feb 6, 2013 at 6:15

Follow



Naveed S

5,226 ● 4 ● 36 ● 52

answered Aug 21, 2008 at 17:12



N8g

636 ● 2 ● 8 ● 19



In terms of actual features, rather than shortcuts, I strongly recommend taking a look at [Mylyn](#).

14



It essentially skins Eclipse with a task focussed view. It tracks the files you touch when working on a task, and focusses many aspects of the UI onto the resources that it decides are relevant to the job in hand. Eclipse can be somewhat busy to look at, especially with a big multi module project, and Mylyn helps cut through the cruft. The connectivity to issue tracking software and source control repositories is also excellent.

In my experience, it polarises opinion amongst those who try working with it, which is probably a sign that it is offering something interesting... Don't mean to sound like a fanboy - it is definitely worth a look though.

[Share](#) [Improve this answer](#)

[edited Aug 21, 2008 at 21:13](#)

[Follow](#)

answered Aug 21, 2008 at 20:04



[serg10](#)

32.6k ● 16 ● 75 ● 94

10 minutes is all it takes to become a total convert to MyLyn.
If task Notes editor is rich (i want bullet items) it'll be great
– user414441 Mar 22, 2009 at 10:16



CTRL + **PAGE DOWN** / **CTRL** + **PAGE UP** to switch between opened editors

11



CTRL + **E** to also switch between opened editors (allows to type the name)



CTRL + **O** is extremely important for me. You don't longer need the Outline View then (you can close it which will give you more space). Then, you can type a method name or just the beginning of it and you quickly can get to it. I also use it to inspect what stuff is available. For example: **CTRL** + **O** and then type get ... now I see all getters.

F3 while an element is selected in the code: brings you to its definition or it's source. e.g. used on a method call it brings you into the source code of that method.





CTRL + **M** to maximize the current window

As already said, **CTRL** + **3** is extremely good. It basically allows you to use Eclipse completely without a mouse. Just type **CTRL** + **3** and then package explorer for example.

CTRL + **F8** cycle trough perspectives

CTRL + **L** allows to type a line number and brings you directly to that line.

CTRL + **SHIFT** + **G** searches for all references to the selected element in the workspace.

And not a shortcut: In the project settings under Java Editor you can find Save Actions. This allows you to set up the project so that the code is automatically cleaned up and formatted when you save a file. That's very good it safes you from constantly pressing  +  and  + .

Share Improve this answer

edited Nov 12, 2010 at 7:16

Follow

community wiki

2 revs, 2 users 76%

Alex

Ctrl+3 is a gold nugget. – [demonkoryu](#) Sep 21, 2010 at 5:16



10



Eclipse auto refresh isn't on by default so if you make changes to a file outside of eclipse, the change won't be reflected in your build. this is very annoying if you just did an svn/git update/rebase and things aren't working the way they're supposed to.



Turn it on in windows->preferences->workspace and tick Refresh Automatically.



Share Improve this answer

answered Dec 24, 2009 at 3:30

Follow



[Charles Ma](#)

49k ● 22 ● 91 ● 101



Local History

9



It's a great feature. Eclipse has its own mini-CVS for all files in a project. If you want to revert some change you made, or even restore deleted file - you can right click on the project and select "Restore from Local History".



Just saved my ass *tears of joy*



Share Improve this answer

answered [Sep 24, 2010 at 2:37](#)

Follow

community wiki
[serg](#)

1 Encouraging a dev not to use real source control is flawed
IMO – [cmcginty](#) Mar 18, 2011 at 3:37



CTRL + Shift + P to **jump to the matching bracket/parenthesis**.

7



Share Improve this answer

answered [Jan 19, 2011 at 20:35](#)

Follow



community wiki
[karlphillip](#)



One key feature: **Shift + Alt + T** for the refactoring menu.

4

Share Improve this answer

edited Sep 21, 2010 at 6:15

Follow



answered Aug 21, 2008 at 19:03






abyx






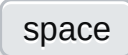
72.6k ● 19 ● 97 ● 121



4



-  -  - , but only type the initial characters (and even a few more) of the class you're looking for. For example, you can type "NetLi" to find NetworkListener

- In the Search window,  -  takes you to the first leaf of a tree branch
-  -  is Word Completion. Slightly different from  - 

Share Improve this answer

edited Feb 6, 2013 at 6:13

Follow



Naveed S

5,226 ● 4 ● 36 ● 52

answered Aug 21, 2008 at 19:52









David Koelle

20.8k ● 23 ● 94 ● 130



3

-  +  +  selected text becomes **UPPERCASE**
-  +  +  selected text becomes **lowercase**



Share Improve this answer

answered Jun 10, 2011 at 12:08

Follow



community wiki
[lostiniceland](#)



3



I'd like to add two additional shortcuts:

- `CTRL` + `F6` Switch between open editors (`CTRL` + `SHIFT` + `F6` to scroll through the list in the opposite direction)
- `CTRL` + `F11` start program in debug mode
- `F11` start program in normal mode

Share Improve this answer

edited Sep 10, 2012 at 13:13

Follow



[Jainendra](#)

25.1k ● 30 ● 128 ● 171

answered Aug 21, 2008 at 21:19



[Johannes K. Lehnert](#)

949 ● 1 ● 10 ● 9

- 1 @Jaguar - please don't edit four-year-old answers by adding minor formatting. See: meta.stackexchange.com/a/116660
– [LittleBobbyTables - Au Revoir](#) Sep 10, 2012 at 13:16



A lot of the key bindings depend on the perspective and view currently active.

3

My most used ones for the Java perspective:



- `ctrl` - `shift` - `r` open resource
- `ctrl` - `shift` - `t` open type
- `ctrl` - `1` quick fix/refactor
- `ctrl` - `j` incremental search
- `ctrl` - `h` search in files (select a base directory and set scope to selected resources)
- `ctrl` - `o` list methods
- `ctrl` - `alt` - `h` open call hierarchy
- `ctrl` - `shift` - `I` list shortcut keys
- hit `ctrl` - `shift` - `I` again to go directly to preferences to change key mappings

Share Improve this answer

edited Feb 6, 2013 at 6:34

Follow



Naveed S

5,226 ● 4 ● 36 ● 52

answered Aug 21, 2008 at 19:16



David Schlosnagle

4,323 ● 2 ● 24 ● 16



2



I am also a fan of Eclipse, however since I spend a majority of my time in Visual Studio, I will suggest that you read Eric Sink's series of articles "C# to Java" (parts 1-4). Not only is Eric always an entertaining read, but this brief series covers some awesome Eclipse insight for those who have not been into Eclipse or have been away



from it for a while:



[From C# to Java: Part 1](#)

[From C# to Java: Part 2](#)

[From C# to Java: Part 3](#)

[From C# to Java: Part 4](#)

Share Improve this answer

Follow

answered Aug 21, 2008 at 17:17



[Ian Patrick Hughes](#)

10.4k ● 3 ● 32 ● 37



2



`Ctrl` - `Shift` - `M` while the cursor is on a class name in your java file, will specifically add that and only that class to your imports. This is different from `Ctrl` - `Shift` - `O` which will not only add those imports not already defined, but will also remove imports not currently needed, something you might not necessarily want to do.

I forgot about [`Ctrl` + `2` -> `r`] scope variable rename. Place the cursor in the variable you wish to rename, press `Ctrl` + `2`, then `r`, then type the new name watching all instances of that variable get renamed at the same time.

It's *awesome* at refactoring Hungarian Notation.

Share Improve this answer

Follow

edited Sep 10, 2012 at 13:21



[Jainendra](#)

25.1k ● 30 ● 128 ● 171

answered Sep 30, 2008 at 19:51



Spencer Kormos

8,431 ● 3 ● 30 ● 47

-
- 2 @Jaguar - please don't edit four-year-old answers by adding minor formatting; these are **minor edits** and should not be approved. See: meta.stackoverflow.com/a/116660
– [LittleBobbyTables - Au Revoir](#) Sep 10, 2012 at 13:24
-



2

`alt` - `shift` - `a` is extremely useful in a few situations.

Share Improve this answer

edited Feb 6, 2013 at 6:32

Follow



community wiki

2 revs, 2 users 67%



[CromTheDestroyer](#)



1

I use a lot of the above and also like for quick search:

`CTRL` + `J` then type what I am looking for, then `CTRL` + `K` for next occurrence.

Share Improve this answer

edited Nov 12, 2010 at 7:17

Follow



community wiki

2 revs, 2 users 67%



[oshai](#)



1

Lately I've been using the [MouseFeeds](#) plugin to automatically tell me what the key stroke combinations are. That way by repetition I remember them better.



This [link](#) has a better picture and description of what it looks like and does.



Share Improve this answer

answered [Nov 16, 2011 at 20:12](#)

Follow

community wiki

[James-Jesse Drinkard](#)

For those who think this program will be blessing - it doesn't work with Eclipse Indigo or Juno. – [tomrozb](#) May 9, 2013 at 12:39



Shift + **Alt** + **b** for the simple navigation row over the editor.

1



Share Improve this answer

edited Feb 6, 2013 at 6:33

Follow



community wiki

3 revs, 3 users 50%

Jakub

To disable it you have to click "Toggle breadcrumb" – [tomrozb](#)
May 9, 2013 at 11:48



I've just released this blog post about [Top 5 Useful Hidden Eclipse Features](#). It contains:

1



1. Favorites: Types and members that will always show up in auto-completion
2. [The awesome block selection mode](#): For multi-line editing
3. The EGit staging view: Much better than git itself
4. Type filters: To remove awt and `java.lang.Object` stuff from auto-completion
5. [Formatter tags](#): To delimit code sections that shouldn't be auto-formatted



Share Improve this answer

edited May 23, 2017 at 12:26

Follow

community wiki

[2 revs](#)

[Lukas Eder](#)



Alt + **left** and **Alt** + **right** will navigate you forward and back.

1

[Share](#) [Improve this answer](#)

[edited Feb 27, 2014 at 20:38](#)



[Follow](#)



community wiki

[2 revs, 2 users 67%](#)

[Jeevs](#)



I find the project-specific settings useful in Eclipse 3.3.

0

For example, if you have several developers working on a project who have different code styles for curly braces, line spacing, number of imports etc. then you can specify the style settings for the project. Then you can configure the save actions so that the code is automatically formatted when a file is saved. The result is everyone's code is formatted the same before it's checked in.



[Share](#) [Improve this answer](#)

[answered Aug 21, 2008 at 17:17](#)

[Follow](#)



[Michael Sharek](#)

5,069 ● 2 ● 31 ● 33