

Configure Keymap

The license could not be verified: License Certificate has expired!

[Previous](#)

[Top](#)

[Next](#)

When you switch to a different IDE one of the things you miss most is your old favourite keyboard shortcuts.

IntelliJ IDEA has its own keyboard shortcuts but also allows you to choose from a variety of alternate keymaps from a few popular IDE's including Eclipse, Visual Studio, and EMACS. A keymap is simply a list of bindings established between a keyboard shortcut and a command available from the IDE. For this reason, two keymaps may be fully comparable only if both IDEs expose the same set of commands.

Eclipse Keymap vs. IntelliJ IDEA Keymap

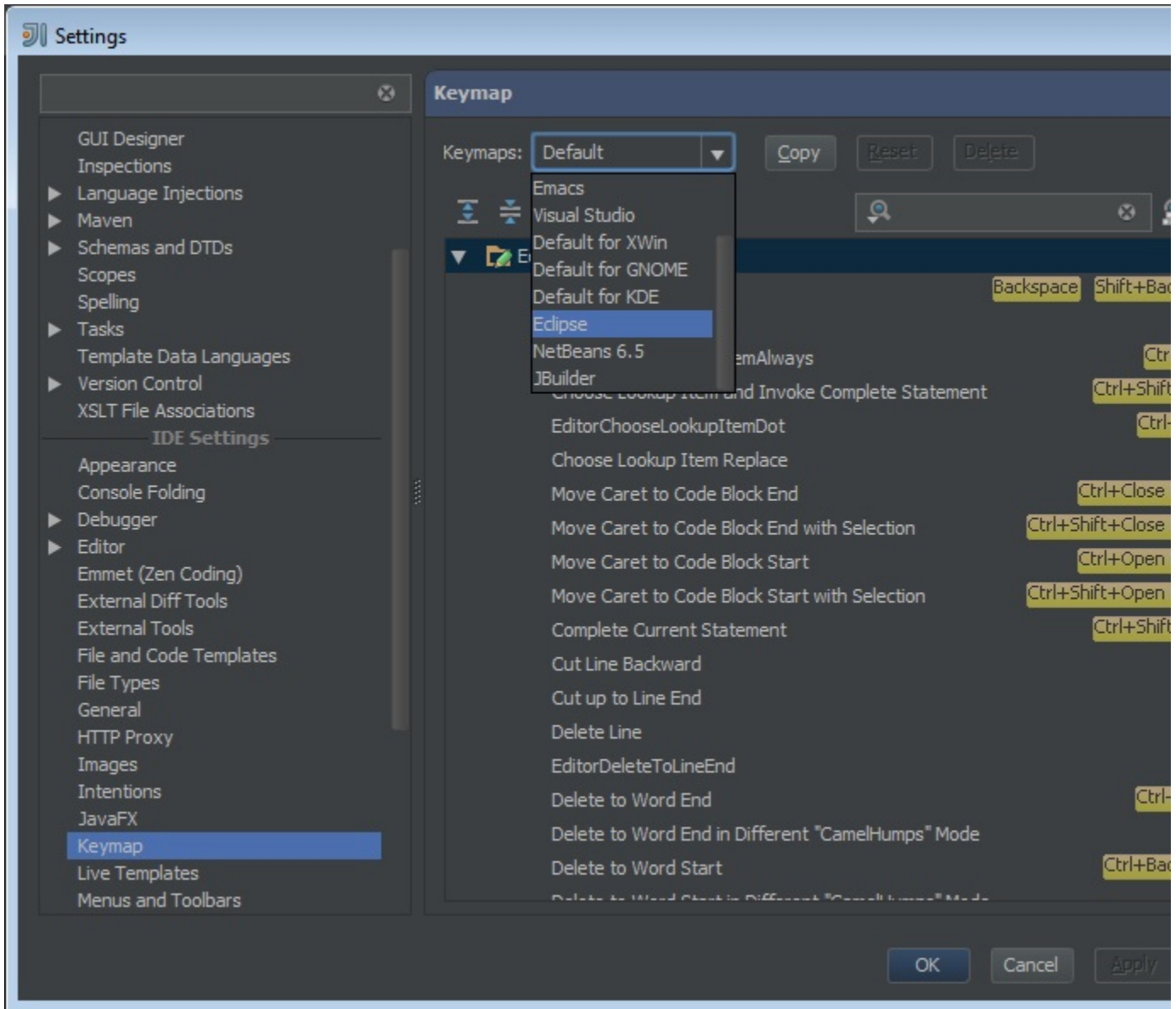
Let's review the major differences existing between the Eclipse and IntelliJ IDEA default keymaps. The table below lists a few of the differences but it should not be considered an exhaustive list.

Action	IntelliJ IDEA	Eclipse
Open any file quickly	Ctrl + Shift + N	Ctrl + Shift + R
Find a type or class	Ctrl + N	Ctrl + Shift + T
Show all the members of current type	Ctrl + F12	Ctrl + O
Jump to source	F4	F3
To jump from one member to the next (or previous)	Alt + Alt +	Ctrl + Shift + Ctrl + Shift +
Go to parent class	Ctrl + B	Ctrl + T
Back and forth through the files you have visited	Ctrl + Alt + Ctrl + Alt +	Alt + Alt +
Last edit location	Ctrl + Shift + Backspace	Ctrl + Q
Delete line	Ctrl + Y	Ctrl + D
Duplicate line	Ctrl + D	Ctrl + Alt +
Open hierarchy	Ctrl + H	F4
Organize imports	Ctrl + Alt + O	Ctrl + Shift + O
Find next / previous	F3 / Shift + F3	Ctrl + K / Shift + Ctrl + K
Debug	Shift + F9	F11
Run	Shift + F10	Ctrl + F11
Step Into	F7	F5
Step Over	F8	F6
Step Out	Shift + F8	F7
Resume	F9	F8
Rename in place	Shift + F6	Alt + Shift + R
Format code	Ctrl + Alt + L	Ctrl + Shift + F
Change signature	Ctrl + F6	Alt + Shift + C
Quick fix	Alt + Enter	Ctrl + 1
View Javadoc for member	Ctrl + Q	Mouse over

Note that in Eclipse you can also get the list of shortcuts by hitting Ctrl + Shift + L If you want to look (or even print) at the entire set of IntelliJ IDEA keyboard shortcuts, instead, you click the Default Keymap Reference element on the Help menu.

Importing the Eclipse Keymap in IntelliJ IDEA

From the Settings dialog box, you select Keymap from the IDE settings block. That's enough to map common actions to the same typical shortcuts in use within Eclipse.

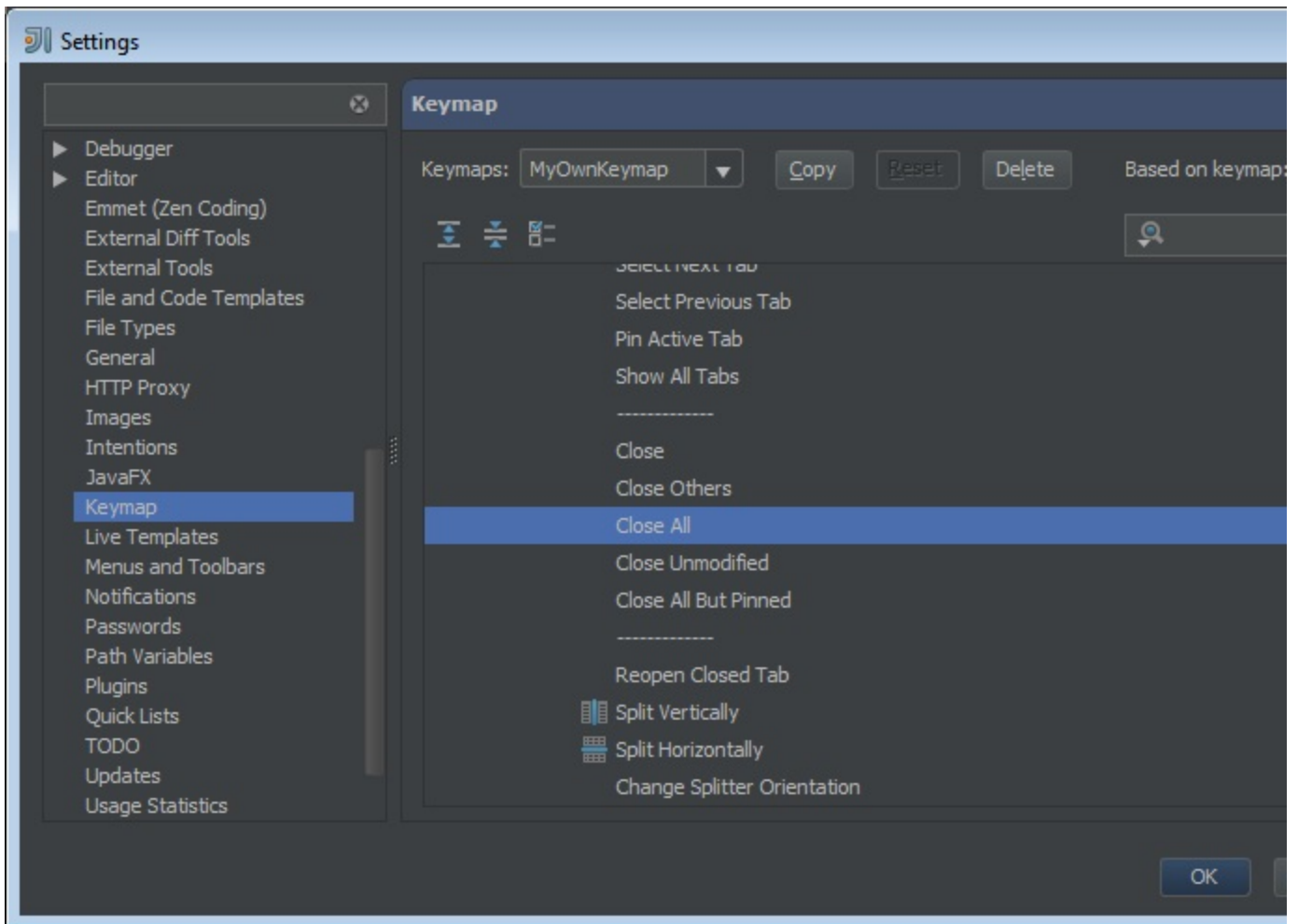


The imported keymap instructs IntelliJ IDEA to recognize typical Eclipse shortcuts for all editor actions that are common to both IDEs.

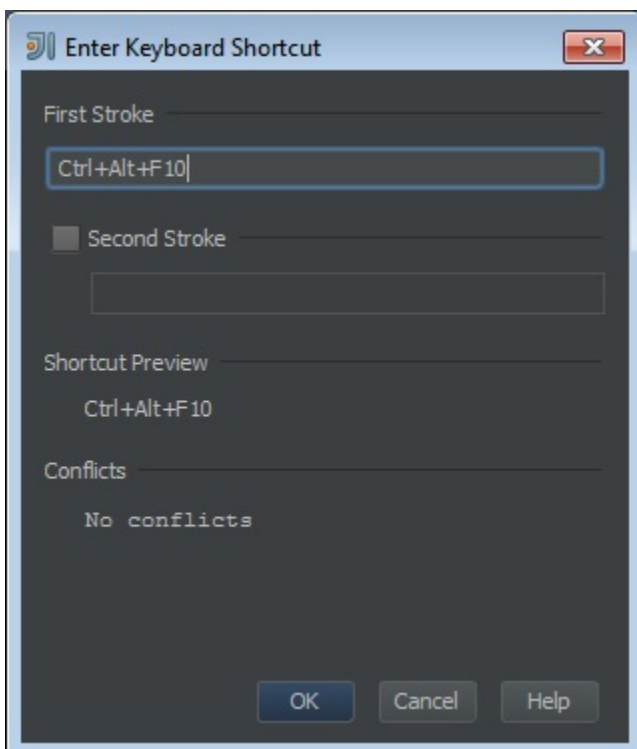
Creating Your Own Key Mappings

In IntelliJ IDEA you can create your own custom keyboard mappings in much the same way you do in Eclipse.

You should first create a copy of an existing keymap. This ensures that you can extend a base keymap without taking the risk of losing some of the predefined settings. To do so, you derive a copy of the selected keymap by clicking the Copy button. Next, you select the IDE command for which you want to add a shortcut. For example, suppose you want to extend the default key map with a shortcut to close all editor tabs instantaneously. The Close All command is not bound to any shortcut by default.



You right-click on the command to bring up the window through which IntelliJ IDEA allows you to add up to two keyboard combinations. If the selected command is already bound to the keyboard, you can remove one or both of the existing shortcuts.



IntelliJ IDEA accepts your shortcut only if it hasn't been assigned to other commands. By associating Close All with `Ctrl + Alt + F10` you now obtain the same effect as `Ctrl + Shift + F4` in Eclipse.

[Previous](#)

[Top](#)

[Next](#)