

# Highlights of NetBeans IDE 8.0 Keyboard Shortcuts & Code Templates

## Finding, Searching, and Replacing

|                |                                   |
|----------------|-----------------------------------|
| Ctrl-F3        | Search word at insert point       |
| F3/Shift-F3    | Find next/previous in file        |
| Ctrl-F/H       | Find/Replace in file              |
| Alt-F7         | Find usages                       |
| Ctrl-Shift-F/H | Find/replace in projects          |
| Alt-Shift-U    | Find usages results               |
| Alt-Shift-H    | Turn off search result highlights |
| Ctrl-R         | Rename                            |
| Ctrl-U, then U | Convert selection to uppercase    |
| Ctrl-U, then L | Convert selection to lowercase    |
| Ctrl-U, then S | Toggle case of selection          |
| Ctrl-Shift-V   | Paste formatted                   |
| Ctrl-Shift-D   | Show Clipboard History            |
| Ctrl-I         | Jump to quick search field        |
| Alt-Shift-L    | Copy file path                    |

## Navigating through Source Code

|                           |                                    |
|---------------------------|------------------------------------|
| Ctrl-O/Alt-Shift-O        | Go to type/file                    |
| Ctrl-Shift-T              | Go to JUnit test                   |
| Ctrl-Shift-B              | Go to source                       |
| Ctrl-B                    | Go to declaration                  |
| Ctrl-G                    | Go to line                         |
| Ctrl-Shift-M              | Toggle add/remove bookmark         |
| Ctrl-Shift-Period / Comma | Next/previous bookmark             |
| Ctrl-Period / Comma       | Next/previous usage/compile error  |
| Alt-Shift-Period / Comma  | Select next/previous element       |
| Ctrl-Shift-1/2/3          | Select in Projects/Files/Favorites |
| Ctrl-[                    | Move caret to matching bracket     |
| Ctrl-K/Ctrl-Shift K       | Next/previous word match           |
| Alt-Left/Alt-Right/Ctrl-Q | Go backward/forward/to last edit   |
| Alt Up / Down             | Next/previous marked occurrence    |

## Coding in C/C++

|             |                     |
|-------------|---------------------|
| Alt-Shift-C | Go to declaration   |
| Ctrl-F9     | Evaluate expression |

## Coding in Java

|                              |                                |
|------------------------------|--------------------------------|
| Alt-Insert                   | Generate code                  |
| Ctrl-Shift-I                 | Fix all class imports          |
| Alt-Shift-I                  | Fix selected class's import    |
| Alt-Shift-F                  | Format selection               |
| Alt-Shift Left/Right/Up/Down | Shift lines left/right/up/down |
| Ctrl-Shift-R                 | Rectangular Selection (Toggle) |
| Ctrl-Shift-Up/D              | Copy lines up/down             |
| Ctrl/Alt-F12                 | Inspect members/hierarchy      |
| Ctrl-Shift-C/ Ctrl-/         | Add/remove comment lines       |
| Ctrl-E                       | Delete current line            |

## Compiling, Testing, and Running

|                |                               |
|----------------|-------------------------------|
| F9             | Compile package/ file         |
| F11            | Build main project            |
| Shift-F11      | Clean & build main project    |
| Ctrl-Q         | Set request parameters        |
| Ctrl-Shift-U   | Create Unit test              |
| Ctrl-F6/Alt-F6 | Run Unit test on file/project |
| F6/Shift-F6    | Run main project/file         |

## Opening and Toggling between Views

|                    |  |
|--------------------|--|
| Ctrl-Tab (Ctrl-`)  | Switch between open documents by order used    |
| Shift-Escape       | Maximize window (toggle)                       |
| Ctrl-F4/Ctrl-W     | Close selected window                          |
| Ctrl-Shift-F4      | Close all windows                              |
| Shift-F10          | Open contextual menu                           |
| Ctrl-PgUp / PgDown | Switch between open documents by order of tabs |
| Ctrl-Alt-T         | Reopen recently closed file                    |

Ctrl-Alt-PgUp / PgDown      Toggle between editor types

Alt-Mouse Wheel Up / Down      Zoom text in / out

Ctrl-Shift-S      Toggle Inspect Mode

## Debugging

|               |                                 |
|---------------|---------------------------------|
| Ctrl-F5       | Start debugging main project    |
| Ctrl-Shift-F5 | Start debugging current file    |
| Ctrl-Shift-F6 | Start debugging test for file   |
| Shift-F5/F5   | Stop/Continue debugging session |
| F4            | Run to cursor location in file  |
| F7/F8         | Step into/over                  |
| Ctrl-F7       | Step out                        |
| Ctrl-Alt-Up   | Go to called method             |
| Ctrl-Alt-Down | Go to calling method            |
| Ctrl-F9       | Evaluate expression             |
| Ctrl-F8       | Toggle breakpoint               |
| Ctrl-Shift-F8 | New breakpoint                  |
| Ctrl-Shift-F7 | New watch                       |

When typing in the Source Editor, generate the text in the right-column below by typing the abbreviation that is listed in the left-column and then pressing Tab.

## Java Editor Code Templates

|      |                             |
|------|-----------------------------|
| En   | Enumeration                 |
| Ex   | Exception                   |
| Ob   | Object                      |
| Psf  | public static final         |
| Psfb | public static final boolean |
| Psfi | public static final int     |
| Psfs | public static final String  |
| St   | String                      |
| ab   | abstract                    |
| as   | assert true;                |
| bcom | /**/                        |
| bo   | boolean                     |
| br   | break;                      |

# Highlights of NetBeans IDE 8.0 Keyboard Shortcuts & Code Templates

|         |  |
|---------|--|
| ca      | catch (  |
| cl      | class  |
| cn      | continue   |
| db      | double   |
| df      | default:   |
| dowhile | do {<br>} while ( <i>condition</i> );  |
| eq      | equals   |
| ex      | extends  |
| fa      | false  |
| fcom    | // <editor-fold></editor-fold>   |
| fi      | final  |
| fl      | float  |
| forc    | for (Iterator <i>it</i> = <i>collection</i> .iterator();<br><i>it</i> .hasNext();) {<br><i>Object elem</i> = ( <i>Object</i> ) <i>it</i> .next();<br>} |
| fore    | for ( <i>Object elem</i> : iterable) {<br>}  |
| fori    | for (int <i>i</i> = 0; <i>i</i> < <i>arr</i> .length; <i>i</i> ++) {}  |
| forl    | for (int <i>i</i> = 0; <i>i</i> < <i>lst</i> .size(); <i>i</i> ++) {<br><i>Object object</i> = <i>lst</i> .get( <i>i</i> ); }                          |
| forst   | for (StringTokenizer <i>st</i> = new<br>StringTokenizer(""); <i>st</i> .hasMoreTokens();) }  |
| forv    | for (int <i>i</i> = 0; <i>i</i> < <i>vct</i> .size(); <i>i</i> ++) {<br><i>Object object</i> = <i>vct</i> .elementAt( <i>i</i> );}                     |
| fy      | finally { }  |
| ie      | interface  |
| ifelse  | if ( <i>condition</i> ) {} else {<br>}   |
| iff     | if ( <i>exp</i> ) {}   |
| im      | implements   |
| inst    | if ( <i>exp instanceof Object</i> ) {<br><i>Object obj</i> = ( <i>Object</i> ) <i>exp</i> ;  |
| iof     | instanceof   |
| ir      | import   |
| le      | length   |
| na      | native   |
| newo    | <i>Object name</i> = new <i>Object</i> ( <i>args</i> );  |
| pe      | protected  |
| pr      | private  |

|          |  |
|----------|--|
| psf      | private static final   |
| psfb     | private static final boolean   |
| psfi     | private static final int   |
| psfs     | private static final String  |
| pst      | printStackTrace();   |
| psvm     | public static void main(String[] args){<br>}   |
| pu       | public   |
| re       | return   |
| runn     | Runnable runnable = new Runnable() {<br>public void run() {} };  |
| serr     | System.err.println ("");   |
| sh       | short  |
| sout     | System.out.println ("");   |
| soutv    | System.out.println(" <i>Object</i> = " + <i>Object</i> );  |
| st       | static   |
| su       | super  |
| sw       | switch (var) { case val: break;<br>default: throw new AssertionError();}                                     |
| sy       | synchronized   |
| tds      | Thread.dumpStack();  |
| th       | throws   |
| tr       | transient  |
| trycatch | try {<br>catch (Exception <i>e</i> ) {}  |
| tw       | throw  |
| vo       | volatile   |
| wh       | while (  |
| whileit  | while ( <i>it</i> .hasNext()) {<br><i>Object elem</i> = ( <i>Object</i> ) <i>it</i> .next();}                |
| whilen   | while ( <i>en</i> .hasMoreElements()) {<br><i>Object elem</i> = ( <i>Object</i> ) <i>en</i> .nextElement();} |
| whilexp  | while ( <i>exp</i> ) {}  |

## JSP Editor Code Templates

|    |                             |
|----|-----------------------------|
| al | window.alert(\$ {message}); |
| br | \$ {no-indent} break;       |

|       |   |
|-------|---|
| ca    | \$ {no-indent} case<br>catch (\$ {exception}) {<br>\$ {cursor}<br>}       |
| cond  | window.console.debug(\$ {debug});   |
| coni  | window.console.info(\$ {info});   |
| conl  | window.console.log(\$ {log});   |
| conw  | window.console.warn(\$ {warn});   |
| ct    | \$ {no-indent} catch  |
| do    | \$ {no-indent} document   |
| dw    | document.write(\$ {message});<br>Else {<br>\$ {cursor}<br>}               |
| else  | \$ {cursor}   |
| eq    | \$ {no-indent} equals   |
| fa    | \$ {no-indent} false  |
| forin | for (var \$ {item} in \$ {object}) {<br>\$ {selection} \$ {cursor}<br>}   |
| fun   | function \$ {name} (\$ {parameters}) {<br>\$ {selection} \$ {cursor}<br>} |
| hi    | \$ {no-indent} window.history   |
| if    | if (\$ {expr}) {<br>\$ {selection} \$ {cursor}<br>}                       |
| iof   | \$ {no-indent} instanceof   |
| json  | "\$ {field}": "\$ {value}"  |
| lo    | \$ {no-indent} window.location  |
| new   | \$ {no-indent} var \$ {name} = new \$ {type} (\$<br>arguments);           |
| prf   | \$ {propertyName}: function (\$ {parameters}) {<br>\$ {cursor}<br>},      |
| re    | \$ {no-indent} return   |
| var   | \$ {no-indent} var \$ {name} = \$ {value};                                |
| win   | \$ {no-indent} window   |