

(Java) Programmieren mit vim

Kathrin Hanauer

21. Juni 2010

- ▶ eingebauter Befehl: :mak oder :make
- ▶ führt standardmäßig make aus
- ▶ Fehler anzeigen: :cl, :cn, :cp ...
- ▶ auszuführendes Programm: makeprg
- ▶ Fehlerformat: errorformat

Beispiel:

```
" compile settings
makeprg=javac\ %\ 2>&1\ \\\|\ sed\ -e\ \"s/\\t/ /g\"
errorformat=%A%f:%l:\ %m,%-Z%p^,%Csymbol\ \ : \ %m,%-C%.%#

" shortcuts
map <F5> <ESC>:wa<CR>:make<CR>
map <F6> <ESC>:!java %<<CR>
```

Abkürzungen allgemein:

- ▶ Definition mit iab <short> <long>
- ▶ Ersetzung nach Abschlusszeichens, z.B. Leerzeichen, Tabulator, <Enter> ..., oder Strg-]
- ▶ spart Tipparbeit

Beispiel:

```
iab sysout System.out.println();<LEFT><LEFT>  
iab syserr System.err.println();<LEFT><LEFT>  
iab psvm public static void main(String[] args)  
\ {<CR><CR>}<ESC><UP>i<TAB><TAB>
```

Sehr einfache Template-Funktion:

```
map <F8> <ESC>:set paste<CR>ggi/**<CR>
\ * Class <C-R>=expand("%:t:r")<CR><CR><CR>
\ * @author <C-R>=expand("$USER")<CR><CR>
\ */<CR>public class <C-R>=expand("%:t:r")<CR> {
\<CR><CR><CR>}<UP><ESC>:set nopaste<CR>i<TAB>
```

- ▶ entspricht Simulation einer manuellen Eingabe

Template mit Skeleton-Datei (z.B. ~/.vim/skel.java):

```
/**                                     /**
 * Class <CLASSNAME>                   * Class TestClass
 * Created on <DATE>                     * Created on Jun 20 2010
 *                                       *
 * @author <USERNAME>                   * @author kaljinka
 */                                       */
public class <CLASSNAME> {               public class TestClass {
}                                         }
```

Laden und Platzhalter ersetzen in vim:

```
function JavaSkel()
Or ~/.vim/skel.java
execute "%s/<CLASSNAME>/" . expand("%:t:r") . "/"g"
execute "%s/<USERNAME>/" . expand("$USER") . "/"g"
execute "%s/<DATE>/" . strftime("%b %d %Y") . "/"g"
endfunction
```

```
autocmd BufNewFile *.java call JavaSkel()
```

- ▶ Supertab
 - ▶ erlaubt Vervollständigung mit <TAB>
 - ▶ wählt kontextabhängig eines der vim ins-completions
file, keyword, omni, user
 - ▶ für Java gibt es standardmäßig kein omnicomplete

- ▶ Supertab
 - ▶ erlaubt Vervollständigung mit <TAB>
 - ▶ wählt kontextabhängig eines der vim ins-completions file, keyword, omni, user
 - ▶ für Java gibt es standardmäßig kein omnicomplete
- ▶ Javacomplete
 - ▶ omnicomplete für Java
 - ▶ ansprechbar mit Strg-X Strg-O
 - ▶ funktioniert auch zusammen mit Supertab

Beispieleinstellungen:

```
set omnifunc=javacomplete#Complete
set completfunc=javacomplete#CompleteParamsInfo

" necessary for javacomplete with <TAB>
let g:SuperTabDefaultCompletionType="<C-X><C-O>"
```

- ▶ JCommenter
 - ▶ generiert Klassen-, Feld- und Methodenkommentare
 - ▶ aktualisiert Javadoc z.B. nach Änderung der Signatur

Einbindung:

```
source ~/.vim/macros/jcommenter.vim  
map <F2> :call JCommentWriter()<CR>
```


- ▶ Taglist
 - ▶ zeigt Fenster mit Klassen und Methoden an
 - ▶ erleichtert Navigieren durch den Quelltext
 - ▶ Ansicht ein-/ausblenden: `:TlistToggle`
 - ▶ Dateien hinzufügen: `:TlistAddFiles`
 - ▶ Ansicht aktualisieren: `:TlistUpdate`
 - ▶ ...

- ▶ Taglist
 - ▶ zeigt Fenster mit Klassen und Methoden an
 - ▶ erleichtert Navigieren durch den Quelltext
 - ▶ Ansicht ein-/ausblenden: `:TlistToggle`
 - ▶ Dateien hinzufügen: `:TlistAddFiles`
 - ▶ Ansicht aktualisieren: `:TlistUpdate`
 - ▶ ...
- ▶ VCScommand
 - ▶ CVS und SVN Unterstützung
 - ▶ Befehle: `:VCSAdd`, `:VCSUpdate`, `:VCSDelete`,
`:VCSvimDiff`, `:VCSCommit`

- ▶ Versuch, die Vorteile von eclipse und vim zu vereinigen
- ▶ besonders praktisch bei geringer Auflösung
- ▶ zwei Modi
 - ▶ vim als Editor in eclipse (hier nicht weiter vorgestellt)
 - ▶ vim als alleiniges Frontend, eclipse läuft als Server im Hintergrund
- ▶ Projektverwaltung mittels vim-Befehlen: `:ProjectCreate`, `:ProjectList`, `:ProjectTree` ...
- ▶ Code-Vervollständigung (`Strg-X Strg-U`)
- ▶ Import-Organisation (`:JavaImportMissing`, `:JavaImportClean`)
- ▶ Code-Korrektur (`:JavaCorrect`), Formatierung (`:JavaFormat`)
- ▶ Nachteil: funktioniert nur bedingt mit anderen Plugins

- ▶ **Supertab:**
http://www.vim.org/scripts/script.php?script_id=1643
- ▶ **Javacomplete:**
http://www.vim.org/scripts/script.php?script_id=1785
- ▶ **JCommenter:**
http://www.vim.org/scripts/script.php?script_id=20
- ▶ **Taglist:**
<http://vim-taglist.sourceforge.net/>
http://www.vim.org/scripts/script.php?script_id=273
- ▶ **VCScommand:**
http://www.vim.org/scripts/script.php?script_id=90
- ▶ **Eclim:**
<http://eclim.org/>