Beipackzettel

Eclipsekurs 2005

Zu riesigen Nebenwirkungen lesen Sie die Manuals
und fragen Sie in Newsgroups und Wikis!

28. April 2005

Essentielle Links

- `http://wiki.freitagsrunde.org/Eclipsekurs_2005` Die Webseite zum Kurs
- `http://wiki.freitagsrunde.org` Die Wiki-Startseite
- `http://www.freitagsrunde.org` Die Freitagsrunde
- `http://www.eclipse.org` Die Eclipse-Homepage
- `http://java.sun.com` Die Java Homepage
- `http://www.3plus4.de` Tutorials für Eclipse - leider derzeit offline
- `http://eclipsewiki.swiki.net` Verschiedene Eclipse-Tipps
- `http://eclipse-plugins.2y.net/eclipse/` Viele Plugins für Eclipse
- `http://www.eclipse.org/cdt` C/C++ Plugin
- `http://java.sun.com/j2se/1.4.2/docs/api/` Java API Dokumentation

Starten im Fakultätsnetz

Startet Eclipse im Fakultätsnetz durch den Aufruf von

```
/home/pub/lib/eclipse-latest/eclipse
```

in einer Shell, dort liegt immer die neueste Version, z.Zt. Version 3.1 Milestone 6.

Weg durch Eclipse

Neues Projekt

Wenn ihr Eclipse zum ersten Mal startet, werdet ihr nach einem Workspace-Verzeichnis gefragt. Unterhalb dieses Verzeichnisses liegen später eure Programme. Der vorgeschlagene Name ist idR auch erst mal okay. Zusätzlich solltet ihr diese Einstellung auch zum Standard machen, indem ihr den Haken vor der Option "Use this as the default and do not ask again" aktiviert. Dann auf "OK" klicken.

Nachdem Eclipse gestartet ist, klickt ihr auf das Icon ganz rechts ("Go to the workbench").

Anlegen des ersten Projekts:

- Menü File ▷ New ▷ Project ▷ Java-Project, klickt auf Next
- Gebt einen Projektnamen ein (z.B. Blatt 1)
- Aktiviert die Option "Create separate source and output folders", klickt auf Finish
- Die Frage, ob ihr zur Java-Perspektve wechseln wollt, könnt ihr ruhig mit Yes beantworten.

Anlegen einer Klasse:

- Menü File ▷ New ▷ Class
- Gebt einen Klassennamen an
- Außerdem solltest ihr für die erste Klasse auch gleich public static void main (Option ziemlich weit unten) erzeugen lassen.
- Im Quelltextfenster könnt ihr programmieren, bis ihr mit dem Resultat zufrieden seid.

Ausführen des Programms:

- Menü Run ▷ Run...
- Wenn der Projektname noch nicht automatisch eingetragen ist, könnt ihr mittels ihn "Browse" herausfinden.
- Die "Main class" sucht euch Eclipse automatisch, wenn ihr auf "Search" geht.
- Wenn ihr jetzt unten den Button "Run" anklickt, wird das Programm ausgeführt.
- Ausgaben über System.out.println könnt ihr unten rechts in der "Console" sehen.

Debugging

Um ein Programm zu debuggen, müsst ihr es erst einmal anlegen und ausführen können (siehe vorheriger Abschnitt).

- Auf die Java-Coding-Perspektive wechseln, wenn ihr da noch nicht seid (oben rechts)
- Im Programmtexteditor auf der ersten Zeile der Main-Methode auf dem linken Rand mit einem Doppelklick einen Breakpoint erzeugen (Eclipse rutscht den automatisch nach unten, wenn der etwas tiefer sitzen muss)
- Menü Run ▷ Debug History ▷ "Der Name des Projekts das ihr angelegt habt"
- Eventuell Dialoxbox bestätigen, dass ihr in die Debugging-Perspektive wechseln möchtet.

Los gehts! Man braucht:

- Menü Run ▷ Terminate : Beendet das Programm sofort
- Menü Run ▷ Step Over : Überspringt die Beobachtung einer Methode
- Menü Run ▷ Step Into : Überspringt die Methode nicht
- Menü Run ▷ Step Return : Führt die Methode bis zum Ende aus

Wichtige Kniffe

Navigieren im Projekt

Die Namen von Typen, Variablen und Methoden können als Hyperlink zu ihrer Definition dienen. Einfach mit Ctrl+linke Maustaste anklicken und schon springt man zur Definition des Elements.

Fehleranzeige

Alle Fehler die Eclipse im Code findet werden markiert indem sie rot unterstrichen werden, außerdem erscheint rechts neben dem Editor eine kleine rote Markierung, die man auch anklicken kann um zum Fehler zu springen.
Bei Warnungen (also nicht-kritischen Fehlern) passiert das gleiche, nur die Markierungen sind gelb.

Quick Fix

Ihr könnt euch bei Fehlern eine Liste mit Korrekturvorschlägen anzeigen lassen, indem ihr denn Cursor auf den Fehler setzt und dann:
Menü Edit ▷ Quick Fix (oder einfach Ctrl + 1).

Vorgegebene Dateien

Wenn ihr vorgegebene Class-Dateien ohne Source-Code bekommt, dann müsst ihr die in den Buildpath aufnehmen:

- Den Ordner mit den Class-Dateien importieren: Menü File ▷ Import ▷ File System
- Menü Project ▷ Properties ▷ Java Build Path ▷ Add Class Folder...

Quelltextvorgaben packt ihr einfach zu den anderen Sourcen dazu: Menü File ▷ Import...

Abgabe für Übungen erzeugen

Um eine ZIP-Datei mit eurem Projekt zu erzeugen:
Menü File ▷ Export ▷ Zip file
Jetzt könnt ihr eure Lösung einfach an eine Email anhängen.

Anzeige von Erinnerungen

Wenn ihr in einem Kommentar TODO oder FIXME schreibt, kann der dahinterstehende Text als Erinnerung unter dem Quelltext angezeigt werden. Geht dazu folgendermaßen vor:

- Zurücksetzen der Java-Perspektive (Menü Window ▷ Reset Perspective)
- Einschalten der Task-Anzeige (Menü Window ▷ Show View ▷ Other ▷ Basic ▷ Tasks)

Versionserklärung

Es gibt (derzeit) zwei benutzbare Versionen:
Version 3.0 (genauer 3.0.1): Das ist der letzte "stable build", dass heißt die letzte fertige Version.
Version 3.1 (genauer 3.1 Milestone 6): Das ist die Version, an der gerade entwickelt wird. Die ist nicht ganz so stabil, enthält auch ab und an mal unfertige Dinge, aber natürlich auch sehr interessante Features.

Download

Hier: `http://eclipse.org/downloads/index.php`
Da wählt ihr einfach die Version aus die zu eurem Betriebsystem passt und gut ist.
Vergesst nicht erst den Java SDK (Software Development Kit) zu installieren.
Und zwar hier: `http://java.sun.com/j2se/1.4.2/download.html`