

Editing Java Easily

Roland Küstermann

Editing Java Easily

Roland Küstermann

Copyright © 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012 Roland Küstermann

Zusammenfassung

Diese Dokumentation bezieht sich auf die Anwendung Editing Java Easily, im weiteren Verlauf auch EJE genannt, in der Version 7.0.

Editing Java Easily ist eine IDE, die speziell auf die Bedürfnisse von Programmieranfängern ausgelegt ist. Daher wurde bewusst auf viele Merkmale einer professionellen IDE, wie Projektmanagement, verzichtet. EJE besticht durch seine Einfachheit: Beginnend mit einer kinderleichten Installation über Java Webstart oder Windows bis hin zum Compilieren und Interpretieren aus einer eigens angepassten Systemkonsole, so dass auch Kommandozeilenkonzepte mit dieser IDE geübt werden können. Die optische Aufbereitung der Fehlermeldungen und die Anbindung eines Learning Management Systems sowie der MindProd Fehlermeldungsbeschreibungen runden den Einsatz für die Programmierausbildung für Anfänger ab.

I. Arbeitshinweise	1
1. Die ersten Schritte	2
Der erste Start	2
Aktualisieren der Anwendung	3
Einrichten von Plugins	3
2. Erstellen, Öffnen, Ändern und Speichern von Dokumenten	5
Erstellen eines Dokuments	5
Öffnen und Ändern eines Dokuments	5
Speichern eines Dokuments	6
3. Compilieren und Fehlerbehandlung	8
Compilieren eines Dokuments	8
Arbeiten mit Fehlermeldungen	8
Fehlervermeidung	8
Fehlerbehandlung	8
Fehlerauflösung	9
4. Ausführen von Dokumenten	10
5. IOTools - Eine Bibliothek zur vereinfachten Eingabe von Werten elementarer Daten-	
typen	11
Installation	11
Verwendung	11
II. Menüs	13
1. Das Datei-Menü	14
2. Das Bearbeiten-Menü	15
3. Das Java-Menü	16
4. Das Tools-Menü	17
5. Das Plugin-Menü	18
6. Das Hilfe-Menü	19
III. Dialoge	20
1. Der Appletviewer Dialog	21
2. Der Suchen-Dialog	22
3. Der Ersetzen-Dialog	23
4. Der Werkzeug-Dialog	24
Eigenschaften des Werkzeug-Dialogs	24
Variablen	25
Beispiele	25
5. Der MindProd-Dialog	27
6. Die Systemkonsole	28
7. Der Einstellungen-Dialog	29
Allgemeine Einstellungen	29
Editor Einstellungen	30
Design Driven Development Einstellungen	31
Java Einstellungen für Fortgeschrittene	31
Quelltextvorlagen	32
IV. FAQs - Häufig gestellte Fragen und Antworten	34
1. Konfiguration und Anpassung von Editing Java Easily	35
Ort der Lizenzdatei	35
Ein/Ausschalten von Produktaktualisierungen	35
Zugriff auf Online-Lernmaterialien/Kurswebseiten	36
Ändern der Standardverzeichnisse	36
Verwendung von Proxyservern	37
V. Anhang	39
1. Impressum	40

Teil I. Arbeitshinweise

Kapitel 1. Die ersten Schritte

Zusammenfassung

Dieser Abschnitt beschreibt die ersten Schritte mit EJE.

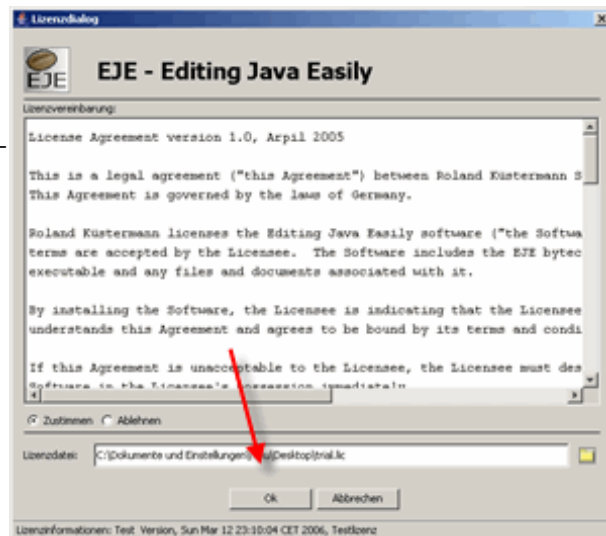
Der erste Start

Voraussetzung für diesen Schritt ist, dass Sie das Editing Java Easily sowie ein adäquates Java installiert ist. Auf dem Desktop liegt die bereits zuvor gespeicherte Lizenzdatei.

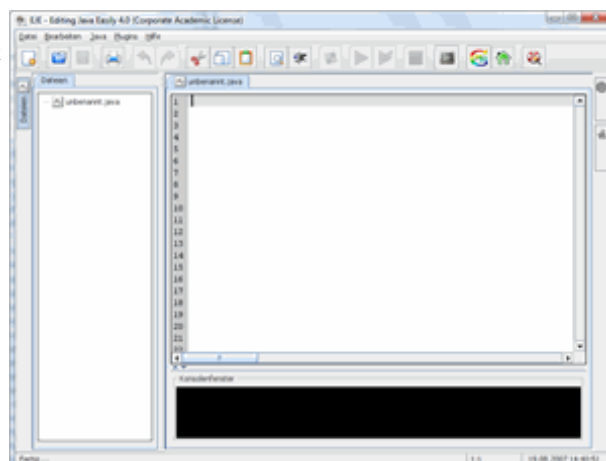
Starten Sie nun EJE per Doppelklick mit der linken Maustaste auf den Link auf Ihrem Desktop.



Wenn Sie EJE das erste Mal starten müssen Sie die Lizenzdatei angeben. Bestätigen Sie hier das Lizenzabkommen zu EJE und geben Sie dann den Ort der Lizenzdatei an. Wählen Sie OK, um EJE zu starten.



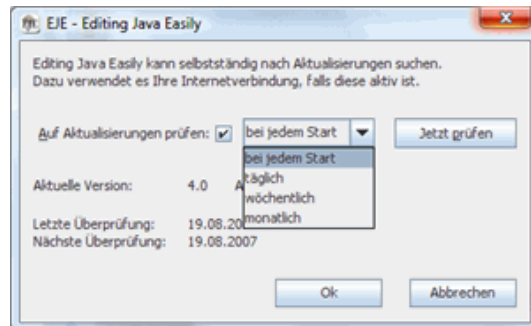
Herzlichen Glückwunsch! EJE ist nun installiert und Sie können jetzt loslegen.



Aktualisieren der Anwendung

Voraussetzung für diesen Schritt ist, dass EJE erfolgreich installiert wurde und Sie eine gültige Lizenzdatei beim ersten Start angegeben haben.

In dem geöffneten Fenster können Einstellungen der regelmäßigen Aktualisierungen vornehmen. Je nach dem welche Variante gewählt wird, verwendet EJE die momentan aktive Internetverbindung, um nach neuen Programmversionen zu suchen und diese entsprechend anzukündigen. Damit dieser Vorgang funktioniert muss zuvor eine Internetverbindung bestehen. EJE informiert darüber, wenn es keine Verbindung herstellen kann.

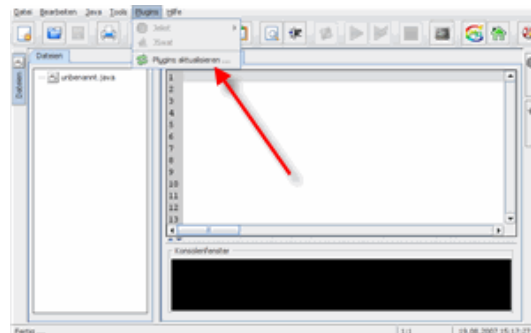


Einrichten von Plugins

Voraussetzung für diesen Schritt ist, dass EJE erfolgreich installiert wurde und Sie eine gültige Lizenzdatei beim ersten Start angegeben haben. Weiterhin muss eine Internet-Verbindung aktiv sein.

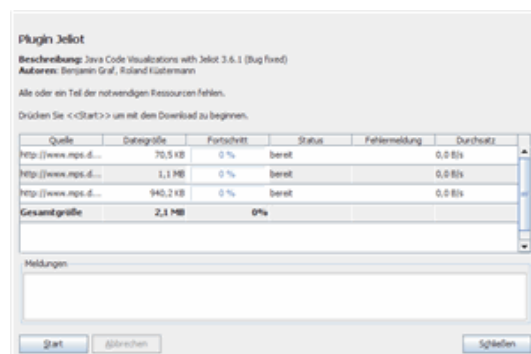
EJE verfügt über eine Plugin-Schnittstelle, die es erlaubt externe Anwendungen direkt in EJE zu integrieren. Eine Übersicht über die verfügbaren Plugins finden Sie hier [<http://www.eje-home.de/deutsch/plugins/index.html>] .

Über den Menüpunkt Plugins aktualisieren im Menü Plugins können die Plugins aktualisiert werden. Der Aktualisierungsprozess wird exemplarisch am Plugin Jeliot beschrieben.

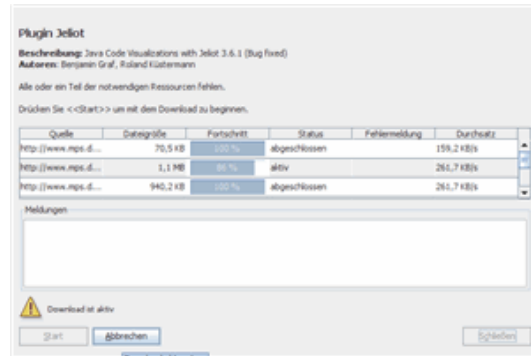


Wenn neue Plugins oder neue Versionen eines Plugins verfügbar sind (nicht automatisch!), so wird der Download Dialog angezeigt. Dieser stellt die heruntergeladenen Dateien in einer Liste dar. Mit einem Klick auf den Schalter Start wird der Download-Prozess gestartet.

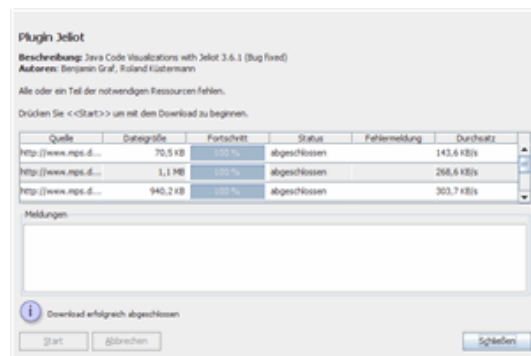
Sollten das Herunterladen wider Erwarten selbstständig abbrechen, so können Sie diesen einfach durch einen erneuten Klick auf den Start-Schalter wiederholen.



Das Achtung-Symbol zeigt, ob der Download-Prozess noch aktiv ist. Der Prozess kann allerdings auch jederzeit durch einen Klick auf den Schalter Abbrechen unterbrochen werden.

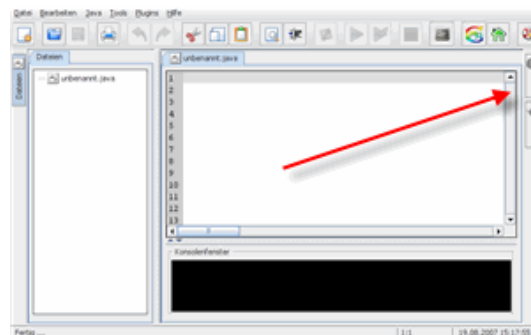


Dass der Download abgeschlossen ist, zeigt das Informations-Symbol. Ein Klick auf den Schalter Schließen schließt den Dialog.



Sie können auf die Plugins über die Schalterleiste am rechten Rand der Anwendung zugreifen.

Je nach Konfiguration des Plugins ist die Ausführung über diese Leiste an bestimmte Zustände des aktuell geöffneten Dokuments sein. So können bestimmte Plugins nur angewendet werden, wenn der Quelltext des aktuellen Dokuments zuvor kompiliert und nicht mehr modifiziert wurde. Weitere Möglichkeiten und Einstellungen finden Sie im Menü Plugins.



Kapitel 2. Erstellen, Öffnen, Ändern und Speichern von Dokumenten

Zusammenfassung

Dieser Abschnitt beschreibt typische Arbeitsabläufe mitEJE. Die Themen Erstellen, Öffnen, Ändern und Speichern werden diskutiert.



Erstellen eines Dokuments

In EJE kann man Dokumente auf drei Wege erstellen.

Bild	Beschreibung	Taste
	Unter Verwendung des Icons Neu in der Toolbar: Ein neuer Editor wird mit der Standardvorlage geöffnet. Je nach Vorlage bekommt die Datei den Namen unbenannt . java oder den vorgeschlagenen Klassennamen.	ctrl + N
	Unter Verwendung des Menüeintrags Neu im Datei Menu: Ein neuer Editor wird mit der Standardvorlage geöffnet. Je nach Vorlage bekommt die Datei den Namen unbenannt . java oder den vorgeschlagenen Klassennamen.	
	Unter Verwendung einer Vorlage aus dem Vorlagen Untermenü im Datei Menü: Ein neuer Editor wird mit der entsprechend ausgewählten Vorlage geöffnet. Je nach Vorlage bekommt die Datei den Namen unbenannt . java oder den vorgeschlagenen Klassennamen.	

Öffnen und Ändern eines Dokuments

Es gibt zwei Möglichkeiten Dokumente zu öffnen. In allen Fällen wird ein neuer Editor geöffnet und die Datei, sofern vorhanden, darin angezeigt.

Bild	Beschreibung	Taste
	Unter Verwendung des Icons Öffnen aus der Toolbar: Ein Öffnen-Dialog wird angezeigt, in dem man, ausgehend vom zuletzt verwendeten Verzeichnis, einen Dateinamen auswählen kann.	ctrl + O
	Unter Verwendung des Menüeintrags Öffnen im Datei Menü: Ein Öffnen-Dialog wird angezeigt, in dem man, ausgehend vom zuletzt ver-	

Bild

Beschreibung

Taste

wendeten Verzeichnis, einen Dateinamen auswählen kann.

Wichtig

Man kann geänderte Dokumente daran erkennen, dass sich der Dateiname in der Registerseite blau färbt. Dokumente können nur gespeichert werden, wenn sie geändert wurden.

Speichern eines Dokuments

Es gibt verschiedene Wege das aktuelle Dokument zu speichern. Das fokussierte Fenster entscheidet, welches das aktuelle Dokument ist.

Bild

Beschreibung

Taste



Unter Verwendung des icon Speichern in der Toolbar: Speichert das aktuelle Dokument. Beim ersten Speichern versucht EJE einen Dateinamen vorzuschlagen und es öffnet sich ein Speichern-Unter Dialog.

ctrl + s



Unter Verwendung des Menüeintrags Speichern im Menü Datei : Speichert das aktuelle Dokument. Beim ersten Speichern versucht EJE einen Dateinamen vorzuschlagen und es öffnet sich ein Speichern-Unter Dialog..



Unter Verwendung des Menüeintrags Speichern unter im Menü Datei : Speichert das aktuelle Dokument unter einem neuen Namen. Dazu wird der Speichern-Unter Dialog geöffnet.



Unter Verwendung des Menüeintrags Alles speichern im Menü Datei : Speichert alle geöffneten und modifizierten Dokumente. Dazu wird die speichern-Methode eines jeden Editors separat aufgerufen.



Unter der Verwendung des Compiler: Vor dem Compilieren wird die Datei automatisch gespeichert, allerdings nur, wenn Sie schon einmal gespeichert wurde.

Wenn ein Dokument erstellt wird lautet der Dateiname unbenannt . java. Beim ersten Speichern versucht EJE einen passenden Dateinamen für das Dokument anhand des Inhalts zu bestimmen. Dazu wird der Quelltext nach einer öffentlichen Klasse durchsucht, die eine main-Methode enthält. Findet EJE diesen Klassennamen nicht, so wird eine entsprechende Meldung ausgegeben. Abspeichern können Sie aber in jedem Fall.

Wichtig

Sollte EJE den Dateinamen tatsächlich nicht bestimmen können, werfen Sie einen Blick auf Ihre Klasse. In der Regel haben Sie dann die main-Methode falsch geschrieben bzw. die

Signatur passt nicht. Denken Sie auch daran, dass Java zwischen Groß- und Kleinschreibung unterscheidet.



Kapitel 3. Compilieren und Fehlerbehandlung

Zusammenfassung

Dieser Teil beschreibt, wie man compiliert, Fehler vermeidet und mit Fehlermeldungen umgeht.

Compilieren eines Dokuments

Wenn das Programm geschrieben und das Dokument *mindestens einmal gespeichert* wurde, ist der Compiler verwendbar und die zugehörige Funktion aktiviert.

Bild	Beschreibung	Taste
	Unter Verwendung des Icons Compile in der Toolbar. Das aktuelle Dokument wird gespeichert und anschließend compiliert.	F5
	Unter Verwendung des Menüeintrag Compilieren im Menü Java : Das aktuelle Dokument wird gespeichert und anschließend compiliert.	

Arbeiten mit Fehlermeldungen

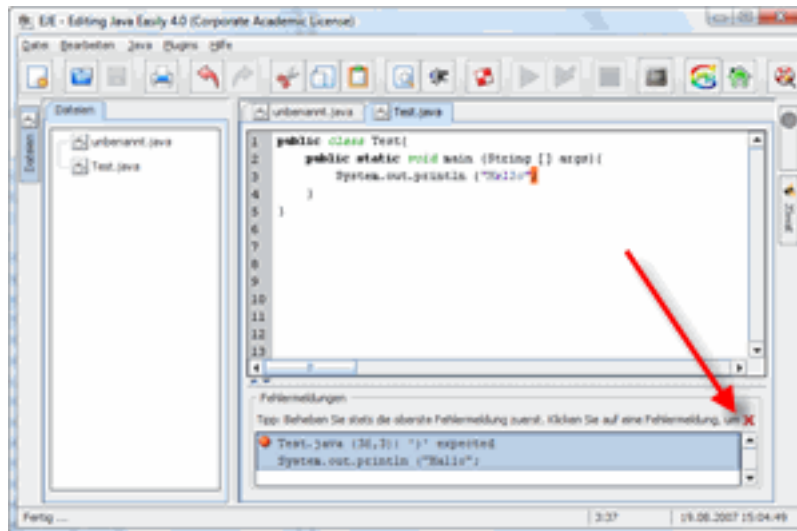
In der Regel baut man beim Compilieren doch den ein oder anderen Fehler ein. EJE unterstützt auch in dieser Phase des Programmierens.

Fehlervermeidung

- *Syntax Highlighting* : Verwende die Syntax-Hervorhebung um Literale verschiedener Datentypen, Schlüsselwörter und vorgebene Datentypen voneinander zu unterscheiden.
- *Auto Bracing* : Durch das automatische Einfügen von Klammern kann man viele Fehler vermeiden.
- *Automatisches Einrücken* : Im Zusammenspiel mit dem *Auto Bracing* kann man feststellen, ob sich die Variablen und Kontrollstrukturen im richtigen Gültigkeitsbereich befinden.

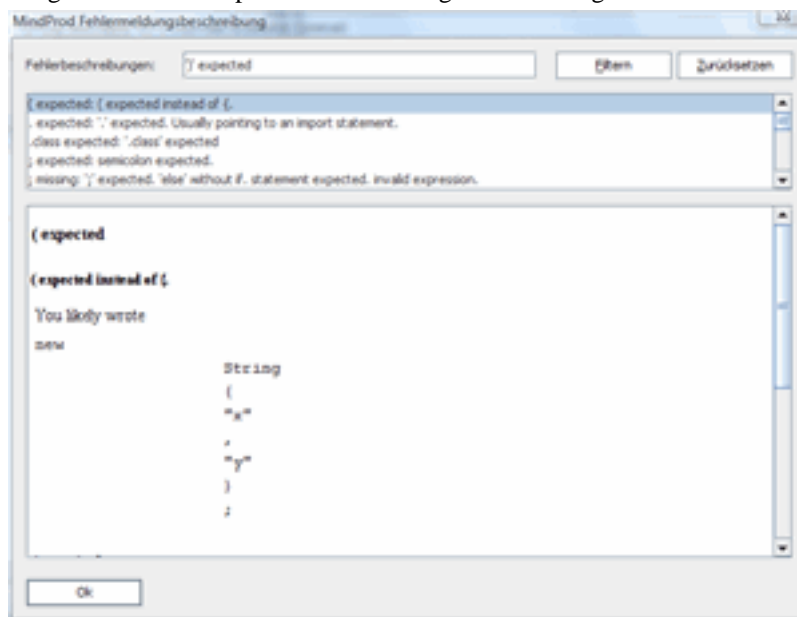
Fehlerbehandlung

Syntaxfehler-Aufbereitung : Syntaxfehler werden optisch und inhaltlich aufbereitet. Durch einen Klick auf den Fehler wird die vom Compiler als verursachende Zeile hervorgehoben. Durch Klicken auf das rote Kreuz kann man auch zur Originalfehlermeldung wechseln.



Fehlerauflösung

Wir empfehlen stets mit der ersten Fehlermeldung zu beginnen. Ein Doppelklick auf die Fehlermeldung öffnet die Mindprod Fehlermeldungsbeschreibungen.



Dieses Feature kann man in den Einstellungen deaktivieren.





Kapitel 4. Ausführen von Dokumenten

Zusammenfassung

Dieser Abschnitt beschreibt, wie man Anwendungen und Applets ausführt.

Wenn alle Fehlermeldungen beseitigt wurden und das Programm erfolgreich kompiliert wurde, kann man den Interpreter nutzen, um das Programm auszuführen. Wir unterscheiden hier zwischen Client-Anwendungen (Applications) und Web-Anwendungen (Applets). Client-Anwendungen können mit Argumenten gestartet werden. Applets werden in der Regel in eine HTML-Datei eingebettet und über den Appletviewer oder den Internet Browser ausgeführt.

Ein Dokument kann nur dann ausgeführt werden, wenn Sie zuvor kompiliert und damit seit der letzten Compilierung nicht mehr verändert wurde.

Bild	Beschreibung	Taste
	Unter Verwendung des Icons Ausführen in der Toolbar: Das aktuelle Programm wird ohne Argumente ausgeführt.	F7
	Unter Verwendung des Menüeintrags Ausführen im Menü Java : Das aktuelle Programm wird ohne Argumente ausgeführt.	
	Unter Verwendung des Menüeintrags Ausführen mit Argumenten in der Toolbar: Vor der Ausführung öffnet sich ein Dialog-Fenster, in welchem man die Argumente mit Leerzeichen getrennt eingeben kann.	F9
	Unter Verwendung des Menüeintrags Ausführen mit Argumenten im Menü Java : Vor der Ausführung öffnet sich ein Dialog-Fenster, in welchem man die Argumente mit Leerzeichen getrennt eingeben kann.	
	Unter Verwendung des Menüeintrags Appletviewer in der Toolbar. Es öffnet sich der Applet-Dialog, welcher die zugehörige HTML-Datei automatisch erzeugt und den Appletviewer aufruft.	
	Unter Verwendung des Menüeintrags Appletviewer ... im MenüJava: Es öffnet sich der Applet-Dialog, welcher die zugehörige HTML-Datei automatisch erzeugt und den Appletviewer aufruft.	

Kapitel 5. IOTools - Eine Bibliothek zur vereinfachten Eingabe von Werten elementarer Datentypen

Da in Java der Zugriff auf die Eingaben per Tastatur relativ kompliziert ist, wurde am Institut für Angewandte Informatik und Formale Beschreibungsverfahren eine Klasse entworfen, die dieses ermöglicht. Durch die Bündelung von Klassenmethoden kann man nun Werte einfach von der Tastatur einlesen. Für jeden der in Java vertretenen einfachen Datentypen wurde eine Klassenmethode, wahlweise mit oder ohne Prompt verwendbar, implementiert. Für die genaue Anwendung siehe Abschnitt Verwendung.

Installation

EJE bündelt die IOTools automatisch. Damit ist eine Installation nicht nötig. Nichtsdestotrotz gibt es Möglichkeit externe Bibliotheken in EJE einzubinden. In den Experten-Einstellungen gibt es die Möglichkeit, eine Komma-separierte Liste von Bibliotheken und Pfaden anzugeben.

Verwendung

Wollen Sie die Klasse `IOTools` in Ihrem Programm verwenden, so müssen Sie in die erste Zeile Ihres Programms nur noch

```
import Prog1Tools.IOTools;
```

einfügen.

Folgende Methoden sind dadurch unter anderem bereitgestellt:

- Die Methode `readInteger` oder `readInt` liest eine Zahl vom Typ `int` von der Tastatur ein und gibt diese als Ergebnis zurück. Um beispielsweise zwei ganze Zahlen von der Tastatur einzulesen und in den Variablen `a` und `b` zu sichern, genügt folgendes Programmstück

```
int a = IOTools.readInteger();
int b = IOTools.readInteger();
```

- Die Methode `readDouble` liest eine Zahl vom Typ `double` ein. Obiges Beispiel würde also für `double` -Zahlen wie folgt aussehen:

```
double a = IOTools.readDouble();
double b = IOTools.readDouble();
```

- Die Methode `readLong` liest eine Zahl vom Typ `long` ein. Die Methoden `readShort` und `readFloat` tun dies für die Datentypen `short` und `float`.
- Die Methode `readLine` liest eine ganze Textzeile (abgeschlossen durch den Druck auf die Eingabetaste).
- Die Methode `readString` liest ein einzelnes "Textwort" von der Tastatur. Ein solches Textwort besteht aus einem `String`, der weder durch Leer- noch Tabulator- bzw. Zeilenende-Zeichen auseinandergerissen ist. Geben wir beispielsweise die Zeile

ein und rufen den Befehl `readString` auf, so liefert er lediglich `Dies` als Ergebnis. Um an das nächste Wort zu gelangen, muss die Methode erneut aufgerufen werden.

- Die Methode `readChar` liest ein einzelnes Zeichen, welches nicht gleich dem Leerzeichen, Zeilenendezeichen oder dem Tabulatorzeichen ist. Die Methode basiert hierbei auf der `readString`-Methode, das heißt, es werden Textworte eingelesen und in ihre einzelnen Komponenten aufgespalten. Das Programmstück

```
IOTools.readChar();  
char a=IOTools.readChar();  
int b=IOTools.readInteger();
```

bei der Eingabe `abc123 456` also `a='b'` und `b=456`, da die Ziffern 123 noch zum ersten Textwort gehören.

- Die Methode `readBoolean` liest einen boolschen Wert ein. Hierbei ist auf Groß- und Kleinschreibung zu achten; die Eingabe `True` kodiert beispielsweise keinen Wert vom Typ `boolean`. Es muss vielmehr `true` heißen. Wie wir in obigen Beispielen gesehen haben, können auch mehr als eine einzulesene Information pro Zeile eingegeben werden (man muss sie lediglich durch Leerzeilen trennen). Hierbei muss man natürlich auf die Reihenfolge der Eingaben achten. Der Befehl `readInteger` wird bei der Eingabe

Ich gebe jetzt einmal 13 ein.









als Ergebnis den Wert 13 zurückgeben, da dies die erste gültige Ganzzahl ist. Die zuvor stehenden Textworte werden verworfen.

Teil II. Menüs

Kapitel 1. Das Datei-Menü

Zusammenfassung

Dieser Teil beschreibt das Datei-Menü. Das Datei-Menü beinhaltet alle Aktionen, die das Dokumentenmanagement betreffen.

Bild	Aktion	Beschreibung	Taste
	Neu	Öffnet einen neuen Editor unter Verwendung der gewünschten Standardvorlage.	ctrl + N
	Vorlagen	Auswahl einer Vorlage zum Öffnen eines neuen Editors. Vorlagen können in den Einstellungen geändert werden.	
	Öffnen	Öffnet eine Datei aus dem System.	ctrl + O
	Schließen	Schließt den aktuellen Editor und speichert die Änderungen, falls nötig.	ctrl + W
	Alle Schließen (außer aktives)	Schließt alle geöffneten Fenster, außer das aktive.	
	Alle Schließen	Schließt alle Fenster.	
	Speichern	Speichert den Inhalt des aktuellen Editors und öffnet den Speichern-unter-Dialog, falls das Dokument das erste Mal gespeichert wird.	ctrl + S
	Speichern unter ...	Zeigt den Speichern-unter-Dialog für den aktuellen Editor an.	
	Alles speichern	Speichert alle Inhalte aller geöffneten Editoren, falls notwendig.	
	Einstellungen	Zeigt den Einstellungen -Dialog	
	Drucken	Druckt den Inhalt des aktuellen Editors.	ctrl + P
	Letzte Dateien	Führt die vier zuletzt benutzten Dateien auf.	
	Beenden	Beendet die Anwendung.	alt + F4

Kapitel 2. Das Bearbeiten-Menü

Zusammenfassung

Dieser Teil beschreibt das Bearbeiten -Menü. Alle Aktionen dieses Menüs beziehen sich auf den selektierten Editor.

Bild	Aktion	Beschreibung	Taste
	Rückgängig	Macht die letzten Änderungen rückgängig.	alt + Backspace
	Wiederherstellen	Stellt die zuletzt rückgängig gemachten Änderungen wieder her.	ctrl + Z
	Ausschneiden	Schneidet den selektierten Text aus und kopiert ihn in die Zwischenablage.	ctrl + X
	Kopieren	Kopiert den selektierten Text in die Zwischenablage.	ctrl + C
	Kopieren (RTF)	Kopiert den selektierten Text als RTF in die Zwischenablage. Die Formatierungen bleiben erhalten.	ctrl + C
	Einfügen	Fügt den Inhalt der Zwischenablage an die aktuelle Cursorposition ein.	ctrl + V
	Blockkommentar	Fügt Blockkommentar zur selektierten Zeile hinzu.	ctrl + B
	Zeilenkommentar	Fügt Zeilenkommentar zur aktuellen Zeile hinzu.	ctrl + L
	Zeile duplizieren	Die aktuelle Zeile wird dupliziert.	shift + ctrl + D
	Zeile löschen	Die aktuelle Zeile wird gelöscht.	ctrl + D
	Suchen	Öffnet den Suchen -Dialog.	ctrl + F
	Ersetzen	Öffnet den Ersetzen -Dialog.	ctrl + R
	Einrücken	Wendet die Quelltext-Formatierungen auf den aktuellen Inhalt an. <i>Bestehende Formatierungen werden verworfen.</i>	ctrl + alt + I
	Gehe zu	Öffnet ein Eingabefenster, in dem man die Zeile angeben kann, zu der man springen möchte.	ctrl + G

Kapitel 3. Das Java-Menü

Zusammenfassung

Dieser Teil beschreibt das Java -Menü. Dieses Menü beinhaltet alle Aktionen, die Java betreffen, und bezieht sich auf den aktuellen Editor.


Bild	Aktion	Beschreibung	Taste
	Compilieren	Compiliert das aktuelle Programm. Falls vorher Änderungen zu speichern waren, wurde dies automatisch gemacht.	F5
	Ausführen	Führt das Programm aus. <i>Diese Aktion ist nur aktiviert, wenn das Programm vorher compiliert wurde.</i>	F7
	Ausführen mit Argumenten	Führt das Programm mit Argumenten an, welche in einem Eingabefenster angegeben werden. <i>Diese Aktion ist nur aktiviert, wenn das Programm vorher compiliert wurde.</i>	F9
	Ausführung unterbrechen	Unterbricht die Ausführung des aktuell laufenden Programms. <i>Diese Aktion ist nur aktiviert, wenn das Programm vorher compiliert wurde und gerade ausgeführt wird.</i>	
	Checkstyle ausführen	Prüft das aktuelle Programm auf die Standard-Code-Conventionen.	F12
	Javadoc (Datei)	Führt Javadoc für die aktuelle Datei im Unterverzeichnis "docs".	F12
	Javadoc (Verzeichnis)	Führt Javadoc für das aktuelle Verzeichnis aus.	F12
	System-Konsole	Öffnet die System-Konsole .	ctrl + K
	Aufräumen	Löscht alle class-Dateien im aktuellen Verzeichnis.	ctrl + K
	Mindprod	Öffnet die Mindprod Fehlerbeschreibungsdatenbank.	ctrl + M

Kapitel 4. Das Tools-Menü

Zusammenfassung

Dieser Teil beschreibt das Tools-Menü. Dieses Menü ist nur aktiv, wenn der Anwendungsmodus auf "Full Feature" gestellt ist (siehe Einstellungen-Dialog).


Dieses Menü beinhaltet externe Werkzeuge auf, die konfiguriert wurden sowie den Konfigurationsdialog.

Bild	Aktion	Beschreibung	Taste
	Werkzeug x	Beispiel-Werkzeug	ctrl + 1
	Konfiguration	Öffnet den Konfigurations -Dialog, um externe Werkzeuge zu definieren.	

Kapitel 5. Das Plugin-Menü

Zusammenfassung





Dieser Teil beschreibt das Plugin -Menü. Das Plugin-Menü beinhaltet haltet alle Aktionen, externe Plugins betreffend. Auf der Produkt- Homepage [<http://www.eje-home.de>] werden die verfügbare Plugins aufgelistet.

Bild	Aktion	Beschreibung	Taste
	Plugin	Beispiel-Plugin	
	Plugins aktualisieren	Aktualisiert die Plugins aus dem Internet. Eine Internetverbindung ist dazu notwendig.	

Kapitel 6. Das Hilfe-Menü

Zusammenfassung

Dieser Teil beschreibt das Hilfe-Menü. Es beinhaltet Aktionen zur Hilfe, Updates und weiteren Informationen.

Bild	Aktion	Beschreibung	Taste
	Hilfe	Hilfe anzeigen	F1
	Kursseite anzeigen	Öffnet die in den Einstellungen festgelegte Kursseite.	ctrl + I
	EJE Webseite	Öffnet den Internet Browser mit EJE Homepage.	
	Lizenzdatei aktualisieren	Löscht die aktuelle Lizenzdatei. <i>Führen Sie dies nur durch, wenn Sie eine neue Lizenzdatei erhalten haben.</i>	
	Produkt aktualisieren ...	Prüft, ob neue Aktualisierungen vorliegen.	
	Über	Zeigt Versionsinformationen an.	

Teil III. Dialoge

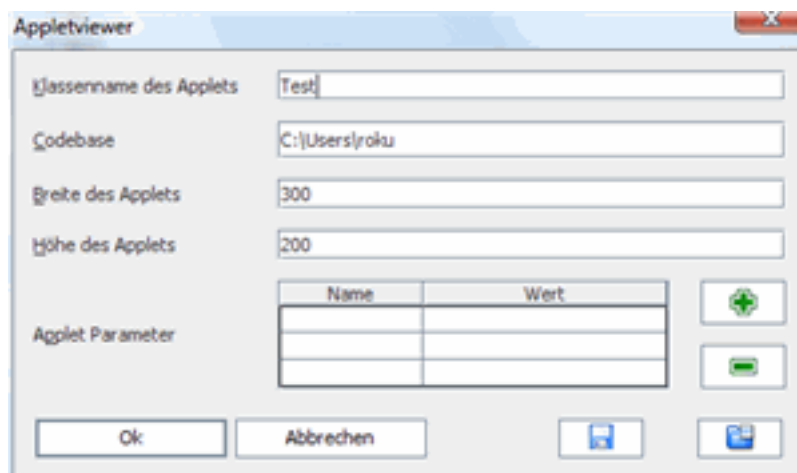
Kapitel 1. Der Appletviewer Dialog

Zusammenfassung

Dieser Teil beschreibt den Appletviewer Dialog.

Java Applets werden in der Regel im Browser angezeigt und erfordern daher eine HTML-Datei mit einem Applet-Tag, der die Informationen über das zu startende Applet enthält.

Der Appletviewer-Dialog vereinfacht das Starten von Java-Applets. Er extrahiert die Informationen aus dem aktuellen Editor und stellt diese im Dialog zur Verfügung. Diese Einstellungen werden in eine temporäre HTML-Datei gesteckt, welche dann vom Appletviewer zum Start verwendet werden kann.



Eigenschaft	Beschreibung	Applet-Tag
Klassenname	Startklasse, welche von <code>java.applet.Applet</code> erben muss.	<code>code</code> attribute
Codebase	Verzeichnis, von dem die Applet Ressourcen geladen werden	<code>codebase</code> attribute
Breite	Höhe des Applets	<code>width</code> attribute
Höhe	Breite des Applets	<code>height</code> attribute
Applet Parameter	Entspricht den Werten im <code>param</code> -Tag des Applet-Tags. Über Hinzufügen und Löschen können Sie Schlüssel/Wert Paar als Parameter zum Applet hinzufügen.	<code>param</code> tag (name, value) combination

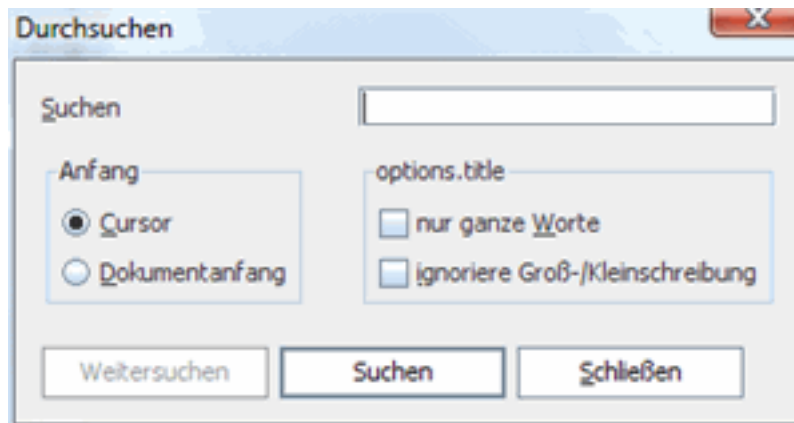
Über Laden und Speichern können Sie die aktuellen Einstellungen abspeichern, so dass Sie diese beim nächsten Mal wieder verwenden können.

Kapitel 2. Der Suchen-Dialog

Zusammenfassung

Dieser Teil beschreibt den Suchen-Dialog.

Der Suchen-Dialog ermöglicht das Suchen im aktuellen Editor.



Eigenschaft	Beschreibung
Nur ganze Worte	Die Suche bezieht sich nur auf Wortgrenzen; d.h. die Suche nach "name" in "dies ist ein namenname" stimmt nicht überein.
Ignore Groß-/Kleinschreibung	Beim Suchen wird die Groß- und Kleinschreibung ignoriert; d.h. die Suche nach der Zeichenkette "Name" stimmt mit "name" und "NAME" im Text überein.
Cursor	Die Suche beginnt an der aktuellen Cursorposition und endet vor dieser.
Dokumentanfang	Die Suche beginnt am Anfang und endet am Ende des Dokuments.

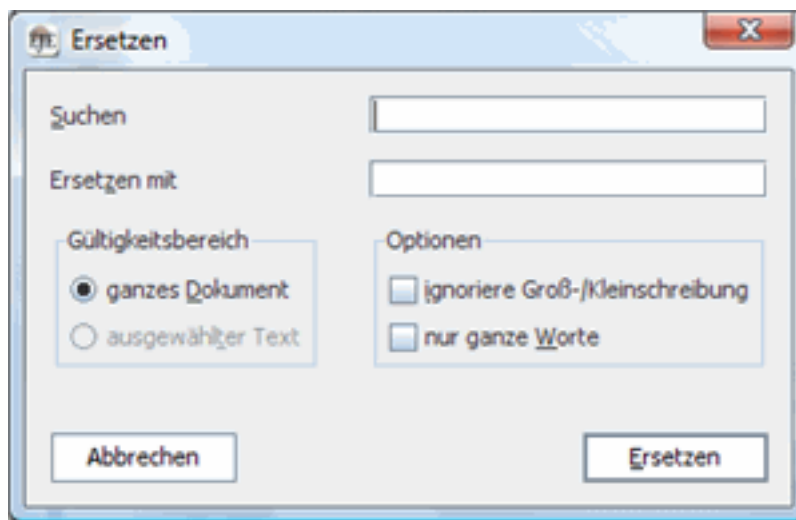
Kapitel 3. Der Ersetzen-Dialog

Zusammenfassung

Dieser Teil beschreibt den Ersetzen-Dialog

Der Suchen/Ersetzen Dialog ermöglicht das Suchen und Ersetzen in der Datei im aktuellen Editor. Die Suche startet am Anfang des Dokuments bzw. am Anfang der zu durchsuchenden Textauswahl.

Wird die Zeichenkette gefunden, wird diese im Text markiert und es öffnet sich ein Dialog, der um Bestätigung der Ersetzung bittet. Der Prozess kann durch Abbrechen beendet werden.



Eigenschaft	Beschreibung
Ganzes Document	Die Suche beginnt am Anfang und endet am Ende des Dokuments.
Ausgewählter Text	Suche beginnt am Beginn der Selektion und endet am Ende der Selektion.
Ignoriere Groß-/Kleinschreibung	Beim Suchen wird nicht auf die Groß- und Kleinschreibung geachtet; d.h. die Suche nach der Zeichenkette "Name" stimmt sowohl mit "name" als auch mit "NAME" im Text überein.
Ganze Worte	Die Suche bezieht sich nur auf Wortgrenzen; d.h. die Suche nach "name" in "dies ist ein namenname" stimmt nicht überein.

Kapitel 4. Der Werkzeug-Dialog

Zusammenfassung

Dieser Teil beschreibt den Werkzeug-Dialog. Darüber hinaus werden Beispielkonfigurationen beschrieben.

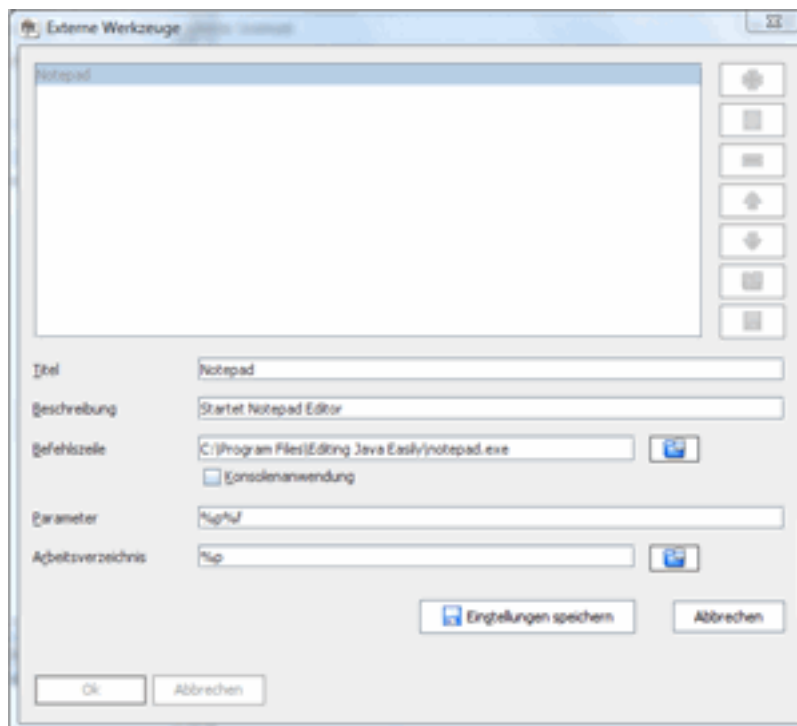
Werkzeuge sind externe Anwendungen, die in diese Anwendung integriert werden sollen. Es wird zwischen Konsolenanwendungen und Fenster-Anwendungen unterschieden. Konsolenanwendungen zeichnen sich dadurch aus, dass Ihre Ein- und Ausgabe über die Konsole des aktuellen Editors läuft. Typische Konsolenanwendungen sind javadoc, zur Erstellung von Java Dokumentationen, oder jdb, dem Konsolen Debugger für Java Programme.

Diese Anwendung unterstützt den kontextsensitiven Zugriff auf den aktuellen Editor.

Beispiel: Externe Werkzeuge, JavaDoc oder der JDB (Debugger), können darüber definiert werden. (siehe Beispiele)

Eigenschaften des Werkzeug-Dialogs

Um ein Werkzeug hinzuzufügen, drücken sie Hinzufügen . Zum Ändern eines bestehenden Werkzeugs, selektieren Sie dieses in der Werkzeugliste und drücken Sie auf Editieren . Zum Übernehmen der Änderungen verwenden Sie den Knopf Einstellungen speichern , zum Verwerfen Abbrechen . Durch Drücken auf Löschen entfernen Sie den selektierten Eintrag.



Eigenschaft

Beschreibung

Titel

Titel der Anwendung, erscheint im Menu

Beschreibung

Beschreibung der Anwendung, erscheint als Tooltip

Eigenschaft	Beschreibung
Befehlszeile	Geben Sie hier die Befehlszeile der externen Anwendung an. Nutzen Sie dazu die Durchsuchen-Funktion., z.B.: c:\windows\notepad.exe
Konsolenanwendung	Wenn die Anwendung die Ein- und Ausgabe über die Konsole erwartet, dann markieren Sie bitte diese Einstellung.
Parameter	Geben Sie hier die anwendungsspezifischen Argumente an (Standard: aktuelle Datei). Sie können direkt auf die im aktuellen Fenster geöffnete Datei zugreifen. Verwenden Sie dazu die hier definierten Variablen .
Arbeitsverzeichnis	Geben Sie hier das Arbeitsverzeichnis an (Standard: Pfad der aktuellen Datei). Auch hier können Sie Variablen verwenden. (siehe Variablen)

Variablen

*Alle Variablen beziehen sich auf die aktuelle geöffnete Datei, welche unter **c:\java\beginners\HelloWorld.java** abgelegt ist.*

Variable	Beschreibung	Beispiel
%p	Pfad der aktuellen Datei	c:\java\beginners
%f	Dateiname der aktuellen Datei	HelloWorld.java
%n	Dateiname der aktuellen Datei ohne Suffix	HelloWorld
%e	Suffix der aktuellen Datei	.java
%P	Verzeichnisname der aktuellen Datei	beginners
%o	Pfad zum Verzeichnis der aktuellen Datei	%o = %p - %P = c:\java.
%cp	Klassenpfad der aktuellen Datei	hängt von der aktuellen Konfiguration des Systems ab, beinhaltet aber immer %p .

Beispiele

Das folgende Beispiel definiert die Anwendung javadoc , die die aktuelle Datei nach Java Kommentaren durchsucht und die javadoc Dokumentation im Unterverzeichnis **docs** generiert . Wir nehmen an, dass das JDK im Verzeichnis **c:\jdk** installiert ist und die Anwendung im Unterverzeichnis **bin** liegt.

Eigenschaft	Wert
Befehlszeile	c:\jdk\bin\javadoc.exe
Parameter	-d docs -classpath %cp %f

Eigenschaft	Wert
--------------------	-------------

Arbeitsver- zeichnis	%p
-------------------------	----

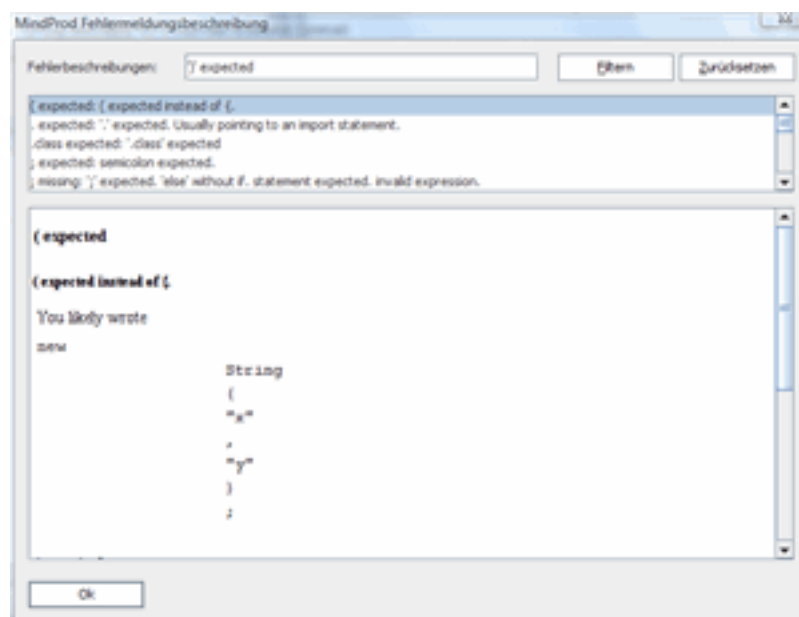
Kapitel 5. Der MindProd-Dialog

Zusammenfassung

Dieser Teil beschreibt den MindProd Dialog.

Dieser Dialog beinhaltet eine Suchmaschine für Syntaxfehlermeldungen.

Die Fehlermeldungen sind nicht 100% genau, da sie sich mit jeder Compiler-Version ändern können. Nichts desto trotz kann man durch die Eingabe von Schlüsselwörtern den Bereich eingrenzen. Es werden die Fehlermeldung und der Schlüssel durchsucht, nicht aber die Erklärung an sich. Die Suchbegriffe sind mit "oder" verknüpft, d.h. es werden alle Ergebnisse zurückgeliefert, in denen mindestens ein Suchbegriff vorkommt. Die Fehlermeldungsbeschreibungen enthalten Hinweise und Anregungen, wodurch ein Syntaxfehler entstanden sein könnte.



Eigenschaft

Beschreibung

Filtern	Suchbegriffe, nach denen gesucht werden soll
Zurücksetzen	Suchergebnis zurücksetzen

Die Fehlermeldungsbeschreibungen wurden freundlicherweise von Canadian MindProd [<http://www.mindprod.com>] zur Verfügung gestellt.

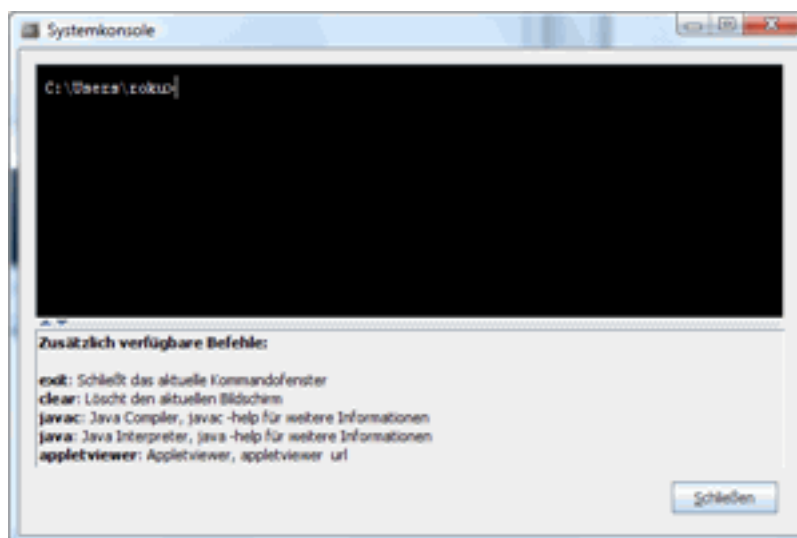
Kapitel 6. Die Systemkonsole

Zusammenfassung

Dieser Teil beschreibt die Systemkonsole.

Die Systemkonsole ist eine Simulation der Konsole des Betriebssystems. Sie wurde entwickelt, da es hin und wieder notwendig ist direkt von der Konsole Programme zu compilieren oder zu starten. Sie ermöglicht es dem Entwickler auf einfache Art und Weise die Kommandos Compilieren, Interpretieren und Ausführen direkt von der Konsole aus durchzuführen. Aufwendige Konfigurationen der Umgebungsvariablen entfallen.

Neben den erweiterten Funktionen unterstützt die Systemkonsole alle Kommandos, die auch auf Ihrem System installiert sind. Die Systemkonsole kann man als nicht-modalen Dialog (Standard) oder als eigenen Prozess in einer separaten JVM starten. Dies hat den Vorteil, dass man den Editor schließen, die Konsole aber weiterhin offen halten kann. Weitere Informationen zu den Einstellungen finden Sie [hier](#).

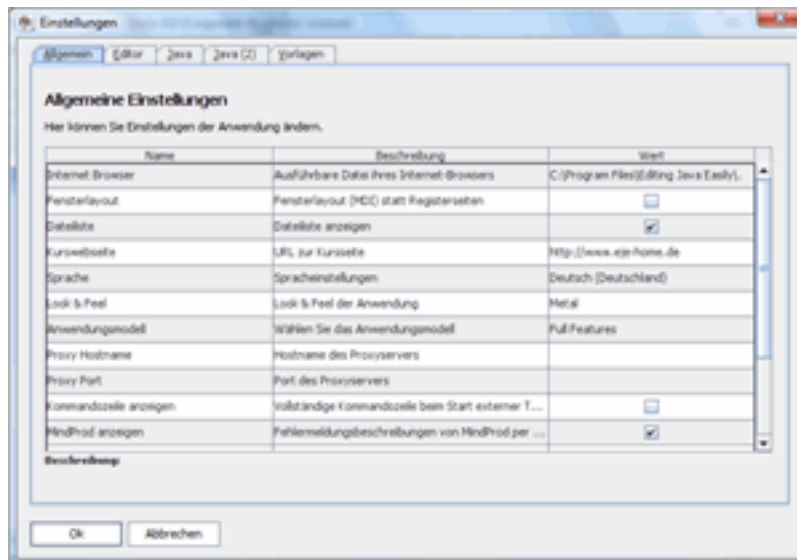


Kapitel 7. Der Einstellungen-Dialog

Zusammenfassung

Dieser Teil beschreibt den Einstellungen-Dialog.

EJE bietet eine Reihe von Möglichkeiten, um an die persönlichen Anforderungen angepasst zu werden. In den Allgemeinen Einstellungen kann das generelle Verhalten und Aussehen von EJE beeinflusst werden. Die Einstellung im Editor erlauben die Anpassung des Editors an verschiedene Nutzungsszenarien. Sollten Sie nicht den standardmäßig installierten Compiler nutzen wollen, so lässt Sie EJE diesen selbst in den Java Einstellungen festlegen. Über das Vorlagenmanagement kann man eigenen Quelltextvorlagen definieren.

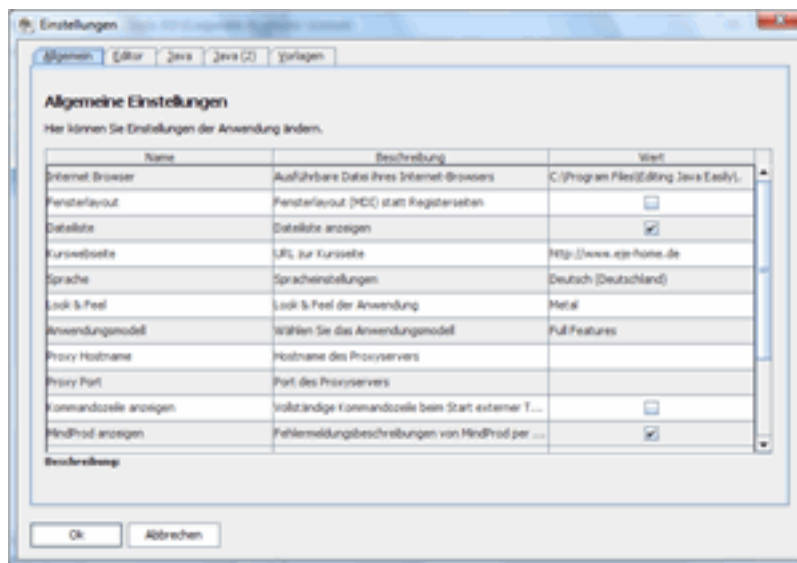


Allgemeine Einstellungen

Dieser Teil beschreibt die allgemeinen Einstellungen zu EJE.

- Internet Browser: EJE versucht den eingestellten Internet Browser automatisch zu bestimmen. Sollte dies nicht funktionieren, kann man hier die Datei des bevorzugten Internet Browsers. z.B. **c:\programme\internet explorer\iexplore.exe** definieren.
- Fenster-Layout: Tab-basiertes Layout (Standard) oder MDI-basiertes (Multiple Document Interface) Layout verwenden. Ein Wechsel benötigt Anwendungsneustart.
- Dateiliste: Hier kann man einstellen, ob man den Datei-Explorer nutzen möchte, oder nicht.
- Kursseite: Sie können hier ein Bookmark hinterlegen, zum Beispiel die Homepage des Kurses.
- Sprache: Sie können zwischen Deutsch und Englisch wählen. Übersetzer anfragen sind jederzeit willkommen.
- Look & Feel: Auswahl zwischen Cross-Plattform Look&Feel (Metal), System- oder Kunststoff Look&Feel (Standard). Ein Wechsel benötigt Anwendungsneustart
- Proxy Hostname: Wenn man einen Proxy-Server für den Internetzugriff benötigt, kann man diesen hier definieren. Den Hostnamen des Proxy-Servers erfahren Sie von Ihrem Administrator. Unter Umständen muss der Proxy Port noch definiert werden.
- Proxy Port: Wenn man einen Proxy-Server für den Internetzugriff benötigt, und dieser einen Port benötigt, kann man diesen hier definieren. Den Port des Proxy-Servers erfahren Sie von Ihrem Administrator.

- Befehlszeile anzeigen: Zeigt beim Ausführen externer Programme (Compiler, Interpreter, Tools, etc.) die Kommandozeile im Ausgabebereich des aktuellen Editors an. Dies ist nützlich, wenn Sie prüfen wollen, warum extern eingebunden Werkzeuge nicht funktionieren.
- Mindprod verwenden: Aktiviert beim Doppelklick auf eine Syntaxfehlermeldung das Anzeigen der Mindprod Fehlermeldungsbeschreibung
- Java Einstellungen prüfen: Prüft beim Start, ob die aktuellen Java Einstellungen korrekt sind.
- Standardtemplate: Wenn beim Start eine neue Datei erzeugt wird, wird die Standardvorlage verwendet.
- Letzte Dateien öffnen: Öffnet alle beim letzten Schließen geöffneten Dateien beim Start.
- Startbild anzeigen: Zeigt das Startbild beim Starten der Anwendung.
- Auf neue Version prüfen: Prüft beim Start, ob eine neue Version der Anwendung verfügbar ist.

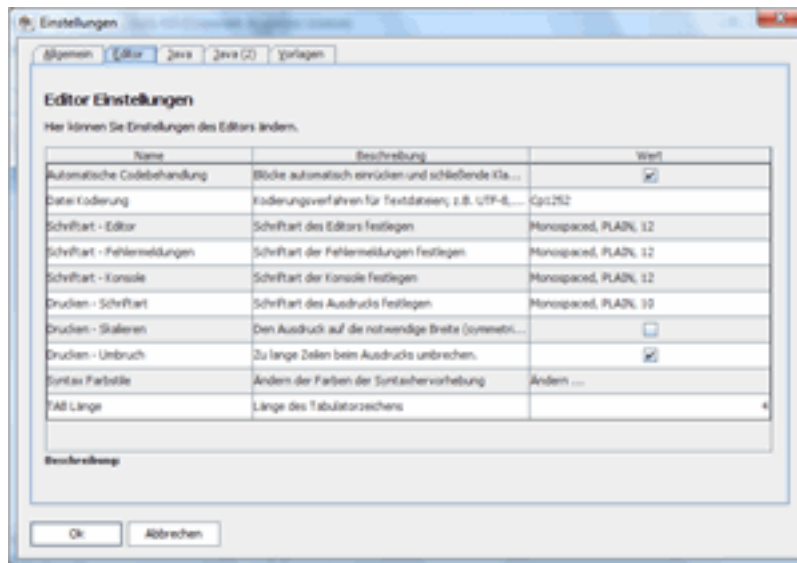


Editor Einstellungen

Dieser Teil beschreibt die Einstellungsmöglichkeiten des Editor.

- Automatische Codebehandlung: Fügt automatisch geschweifte Klammern ein und rückt entsprechend der Formatierungsregeln den Code beim Öffnen eines neuen Blocks ein. Vervollständigt runde und eckige Klammern.
- Automatische Codevervollständigung: Ermöglicht per ctrl+SPACE das Aufrufen der APIs. (Experimentell)
- Top-Level-Pakete: Schränkt die Code-Vervollständigung auf die genannten Top-Level-Pakete ein (Performance-Optimierung).
- Datei Kodierung: Legt die Kodierung fest, in der eine Datei gespeichert und geladen werden soll. Standardmäßig wird die Standardkodierung des verwendeten Betriebssystems verwendet (Windows: windows-1252, Linux: UTF-8). Wenn allerdings Dateien zwischen verschiedenen Betriebssystemen ausgetauscht werden sollen, dann empfiehlt sich die Einstellung UTF-8.
- Schriftart - Editor: Legt die Schriftart und -größe des Editor fest.
- Schriftart - Fehlermeldungen: Legt die Schriftart und -größe des Editor fest.
- Schriftart - Konsole: Legt die Schriftart und -größe der Konsole fest.

- Syntax Farbstile: Hier können Farben und Schriftartstile für die verschiedenen Tokenarten definiert werden.



Design Driven Development Einstellungen

Dieser Teil beschreibt die Einstellungen zu Design Driven Development mit C4J (<http://c4j.sourceforge.net>).

- C4J-Engine aktivieren: Mit diesem Schalter wird die C4J-Engine aktiviert.
- Protokollierung: Legt das Niveau der Protokollierung fest.
- Trace: Stellt ausführliche Informationen bei der Ausführung zur Verfügung.
- Verträge ausschließen: Regulärer Ausdruck, der passende Verträge ausschließt.
- Verträge einschließen: Regulärer Ausdruck, der passende Verträge einschließt, hat Vorrang vor Ausschluss.
- Ausführung unterbrechen: Unterbricht die Ausführung des Programms, falls Verträge einer abgesicherten Klasse nicht gefunden werden konnten.
- Ausgabeverzeichnis: Speichert die instrumentierten Class-Dateien im angegebenen Ausgabeverzeichnis.

Java Einstellungen für Fortgeschrittene

Wichtig

Dieser Teil beschreibt die Einstellungen zu Java. Ändern Sie diese Einstellungen nur, wenn Sie sich über der Auswirkung im Klaren sind. Diese Einstellungen sind nur verfügbar, wenn Sie die Anwendung im "Full Feature" Modus gestartet wird (siehe Einstellungen-Dialog).

- Checkstyle Konfigurationsdatei: Hier können Sie eine neue Checkstyle-Datei hinterlegen. Klicken Sie auf den Wert und laden Sie eine neue Datei hoch.
- Classpath : Hier können Sie zusätzliche Pfade und Archive eintragen, um auf externe Klassen zugreifen zu können. Trennen Sie die Einträge mit dem systemabhängigen Pfadtrennzeichen (Windows: Semikolon, Linux: Doppelpunkt)
- Compile -d: Compilieren Sie Ihre Programme mit Debug-Informationen. Diese Einstellung ist standardmäßig aktiviert.

-
- The screenshot shows the 'Einstellungen' (Settings) dialog box in the Eclipse IDE. The 'Java' tab is selected, and the 'Java Feineinstellungen' (Java Fine-tuning) section is expanded. Below the section title, there is a text box stating: 'Hier können Sie Feineinstellungen des Compilers und Interpreters vornehmen.' (Here you can make fine-tuning settings for the compiler and interpreter.)
- Below the text box is a table with three columns: 'Name', 'Beschreibung', and 'Wert'.
- | Name | Beschreibung | Wert |
|------------------------------------|--|-------------------------------------|
| Classpath | Classpath zum Einbinden zusätzlicher Klassen und Archive | |
| Zusätzliche Compiler-Einstellungen | Mit zusätzlichen Compiler-Einstellungen compilieren | |
| compile -g | Mit Debugging Informationen compilieren | <input checked="" type="checkbox"/> |
| compile -d | Verzeichnis, in dem die compilierten Programme abgelegt werden sollen | |
| compile -verbose | Mit mehr Informationen über die compilierten Programme compilieren | <input type="checkbox"/> |
| run -ms | Zusicherungen (Assertions) bei der Ausführung aktivieren, erforderlich | <input checked="" type="checkbox"/> |
| run -verbose | Mit mehr Informationen über das ausgeführte Programm interpretieren | <input type="checkbox"/> |
- Below the table, there is a section titled 'Beobachtung' (Observation) with a text box containing the following text: 'Die Einstellungen sind in der Regel so konfiguriert, wie sie in der Eclipse IDE sind. Wenn Sie die Einstellungen ändern, können Sie die Einstellungen in der Eclipse IDE ändern.' (The settings are usually configured as they are in the Eclipse IDE. If you change the settings, you can change the settings in the Eclipse IDE.)
- At the bottom of the dialog box, there are two buttons: 'Ok' and 'Abbrechen' (Cancel). The 'Ok' button is highlighted.

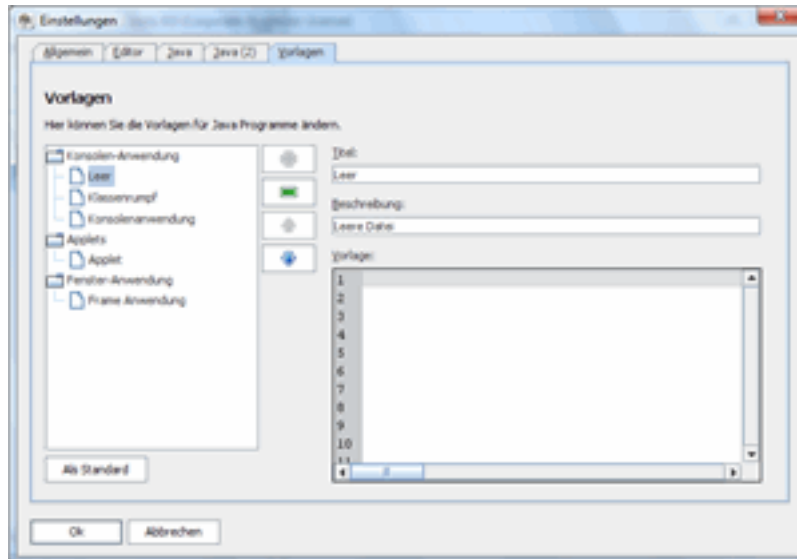
Dieser Teil beschreibt das Vorlagen-System.

Derzeit unterstützt die Anwendung folgende Variablen:

-
- 32

- Die Variable `${GENERIC}` enthält den Bezeichner des Typparameters inklusive der `<-` und `>-` Zeichen, falls der Klassennamen generisch ist.

Die Standardvorlage ist die Vorlage, die bei der Erstellung eines neuen Editors verwendet wird. Über den Button Als Standard kann eine Vorlage als Standard festgelegt werden. Ob dieses Template verwendet werden soll, wird bei den allgemeinen Einstellungen festgelegt.



Teil IV. FAQs - Häufig gestellte Fragen und Antworten

Kapitel 1. Konfiguration und Anpassung von Editing Java Easily

Neben den persönlichen Einstellungen, die jeder Anwender vornehmen kann, ist es möglich, bestimmte Einstellungen beim Start als Vorbelegung mitzugeben. Dies ist insbesondere dort notwendig, wo Software zentral verwaltet wird und bestimmte Rahmenbedingungen erfüllt werden sollen.

Die meisten Konfiguration werden vorgenommen, indem beim Start der Anwendung Konfigurationsparameter (Konfigurationsvariable mit zugehöriger Wertzuweisung) an die Java Virtual Machine (JVM) weitergegeben werden. Dieser wird dann von EJE ausgewertet. Die folgenden Konfigurationsvariablen werden stets mit Beschreibung und einem Beispiel angegeben. *Hinweis:* der Konfigurationsvariablen muss stets das Präfix -D vorangestellt werden.

Hinweis: Alle Einstellungen, welche mit -D an das Programm weitergegeben werden, werden in der Registry hinterlegt, d.h. wenn das Programm ohne diese Parameter das nächste Mal aufgerufen wird, dann werden die Einstellungen aus der Registry gelesen. Dies ist insbesondere dann notwendig, wenn die Anwendung direkt per Doppelklick auf die ausführbare Datei oder über eine Dateieindungsverknüpfung gestartet wird.

Beispiel : Wenn man den Windows-Installer verwendet, kann man die Verknüpfung manipulieren. Alternativ liegt der Installation auch ein Shellscript bei, welches angepasst werden kann. `"C:\Program Files\Editing Java Easily\javastart.exe" --jvmmmin=1.5.0 -jar bin/eje.jar -Dlms.url=http://moodle.mywebsite.de -Dplugins.home=\\server\eje -Deje.home=\\user .`

Ort der Lizenzdatei

Standardmäßig wird die Lizenzdatei im Anwendungsverzeichnis von EJE gesucht. Das Anwendungsverzeichnis setzt sich aus einem ".", dem Anwendungskürzel und der aktuellen Haupt-Versionsnummer zusammen; beispielsweise `.eje-4.0`. Dieses Verzeichnis wird in der Regel im Profilverzeichnis oder Homeverzeichnis angelegt. Da unter Umständen eine Sammellizenz erworben wurde oder das Profilverzeichnis regelmäßig neu initialisiert wird, würde dies bedeuten, dass jeder Anwender die Anwendung freischalten muss.

Lösung: bei der Softwareverteilung wird die Lizenzdatei (umbenannt in `license.txt`) in das selbe Verzeichnis kopiert, in dem die Datei `eje.jar` zu finden ist. Dies ist in der Regel das `bin`-Verzeichnis).

Nebeneffekt : Der Anwender kann die Lizenzdatei nun nicht mehr selbstständig über den Menüpunkt Hilfe-Lizenzdatei aktualisieren aktualisieren.

Ein/Ausschalten von Produktaktualisierungen

Standardmäßig kann der Nutzer entscheiden, wann und wie häufig die Anwendung nach Produktaktualisierungen sucht. Dies ist nun über einen Schalter ein-/ausschaltbar.

Tabelle 1.1. Konfigurationsvariable: Aktualisierungen

Schalter	<code>eje.update</code>
Beschreibung	Zeichenkette <code>true</code> oder <code>false</code>
Beispiel	<code>-Deje.update=false</code> Verhindert, dass das Programm nach Aktualisierungen sucht und überschreibt die persönlichen Einstellungen. <i>Hinweis:</i> Wenn der Schalter auf <code>false</code> steht, ist der Menüpunkt Produkt aktualisieren ... nicht auswählbar.

Zugriff auf Online-Lernmaterialien/Kurswebseiten

EJE bietet die Möglichkeit eine URL auf eine bestimmte Kursseite einzustellen. Standardmäßig weist diese URL auf die Webseite der Anwendung (<http://www.eje-home.de>). Diese URL kann aber an die lokalen Bedingungen angepasst werden, damit der Anwender beispielsweise standardmäßig auf die Informationsseite des Instituts geleitet wird. Der Anwender selbst kann diese dann immer noch über die persönlichen Einstellungen anpassen.

Tabelle 1.2. Konfigurationsvariable: Kurswebseite

Schalter	<code>lms.url</code>
Beschreibung	Zeichenkette, welche die URL enthält
Beispiel	<code>-Dlms.url=http://ilias.learning.de</code> Weist dem zugehörigen Button den angegeben Link zu.

Ändern der Standardverzeichnisse

Es gibt eine Reihe von Verzeichnissen, welche bei EJE angepasst werden können. Dazu gehören das Anwendungsverzeichnis, das Tools-Verzeichnis und das Plugin-Verzeichnis. Das Anwendungsverzeichnis setzt sich aus einem ".", dem Anwendungskürzel und der aktuellen Haupt-Versionsnummer zusammen - beispielsweise `.enje-3.0` - und wird in der Regel im Profilverzeichnis oder Homeverzeichnis angelegt. Das Plugin-Verzeichnis ist standardmäßig ein Unterverzeichnis des Anwendungsverzeichnis mit dem Namen `plugins`. Da es jedoch Konstellationen gibt, bei denen das Profilverzeichnis in regelmäßigen Abständen gelöscht wird, gibt es die Möglichkeit diese Verzeichnisse anzupassen.

Tabelle 1.3. Konfigurationsvariable: Anwendungsverzeichnis

Schalter	<code>eje.home</code>
Beschreibung	Zeichenkette, welche eine Verzeichnis oder ein gültiges Laufwerk enthält
Beispiel	<code>-Deje.home="U:\program settings"</code> Legt das Anwendungsverzeichnis im gegebenen Verzeichnis an. <i>Hinweis:</i> Im Registrierschlüssel <code>JavaSoft\Prefs\de\kuestermann\eje\4\0</code> wird der Wert <code>eje.home</code> bei jeder Änderung abgespeichert. Dies ist notwendig, damit die Anwendung, wenn Sie beispielsweise über eine Dateiendungsverknüpfung gestartet wird, weiß, an welcher Stelle die Einstellungen zu finden sind.

Tabelle 1.4. Konfigurationsvariable: Plugins-Verzeichnis

Schalter	<code>plugins.home</code>
Beschreibung	Zeichenkette, welche eine Verzeichnis oder ein gültiges Laufwerk enthält
Beispiel	<code>-Dplugins.home="c:\programme\editing java easily\plugins"</code> Speichert die Plugins im angegebenen Verzeichnis. Diese Variante bietet sich an, um die Plugins zentral zu verwalten.

Nebeneffekte: Plugins können nur aktualisiert werden, wenn Schreibrechte auf das angegebene Pluginverzeichnis gegeben werden.

Tabelle 1.5. Konfigurationsvariable: Tools-Verzeichnis

Schalter	<code>tools.home</code>
Beschreibung	Zeichenkette, welche eine Verzeichnis oder ein gültiges Laufwerk enthält
Beispiel	<code>-Dtools.home="c:\programme\editing java easily\tools"</code> Speichert die Tool-Konfigurationen im angegebenen Verzeichnis. Diese Variante bietet sich an, um externe Werkzeugkonfigurationen zentral zu verwalten.

Nebeneffekte: Tools können nur aktualisiert werden, wenn Schreibrechte auf das angegebene Pluginverzeichnis gegeben werden.

Verwendung von Proxyservern

Einige Internet-Service-Provider verwenden Proxy-Server, um den Zugang zum Internet zu gewähren. Dies geschieht entweder aus Performanzgründen oder um eine zusätzliche Authentifizierungsschicht

zu verwenden. Da EJE das http-Protokoll verwendet, um auf neue Version zu prüfen und Plugins zu aktualisieren, muss er entsprechend angepasst werden.

Damit man in Java auf einen Proxy-Server zugreifen kann, muss man den Zugriff darauf konfigurieren. Bitte kontaktieren Sie Ihren Administrator, um die richtigen Werte für die folgenden Einstellungen zu ermitteln.

```
http.proxyHost (default: <none>)  
http.proxyPort (default: 80)
```

Falls der Proxy-Server verschlüsselt arbeitet, können Sie diese Eigenschaften verwenden

```
https.proxyHost (default: <none>)  
https.proxyPort (default: 80 if http.proxyHost specified)
```

Im Falle eines authentifizierten Zugriffs sind zusätzlich noch die Eigenschaften

```
http.proxyUserName (Username to use for authentication)  
http.proxyPassword (Password to use for authentication)
```

notwendig.

Damit die Eigenschaften Wirkung zeigen, müssen diese beim Starten des Java Programms an die virtuelle Maschine weitergegeben werden. Dazu öffnen Sie den Link, welcher EJE startet und tragen die oben definierten Eigenschaften mit einem vorgestellten "-D" ein.

Beispiel: Sie möchten eine Verbindung zum Internet über den Proxy-Server proxy.mycompany.com an Port 3128 herstellen. Öffnen Sie die Eigenschaft der Verknüpfung. Unter "Ziel" sollte dort

```
"C:\Programme\Editing Java Easily\javastart.exe"  
--jvmmmin=1.5.0 -jar bin/eje.jar
```

oder ähnliches zu finden sein. Fügen Sie nun die beiden Eigenschaften

```
-Dhttp.proxyHost=proxy.mycompany.com
```

und

```
-Dhttp.proxyPort=3128
```

direkt hinter das Kommando hinzu. Abschließend sollte der Aufruf also wie folgt aussehen:

```
"C:\Programme\Editing Java Easily\javastart.exe"  
-Dhttp.proxyHost=proxy.mycompany.com  
-Dhttp.proxyPort=3128  
--jvmmmin=1.5.0 -jar bin/eje.jar
```

Achtung: Achten Sie darauf, dass zwischen den einzelnen Argumenten stets mindestens ein Leerzeichen steht. Den hinzugefügten Eigenschaften geht nur ein (!) Bindestrich voraus.

Teil V. Anhang

Kapitel 1. Impressum

© 2004 - 2011 Roland
Küstermann
<info@eje-home.de>
Hedwig-Kettler-Str. 5
76137
Karlsruhe
Germany

Der Inhalt dieses Angebots ist urheberrechtlich geschützt. Die Vervielfältigung von Informationen oder Daten, insbesondere die Verwendung von Texten, Textteilen oder Bildmaterial bedarf der vorherigen schriftlichen Zustimmung von Roland Küstermann, im Folgenden Copyright-Inhaber genannt.

Haftungsausschluss

1. Inhalt der Dokumentation:

Der Copyright-Inhaber übernimmt keinerlei Gewähr für die Aktualität, Korrektheit, Vollständigkeit oder Qualität der bereitgestellten Informationen. Haftungsansprüche gegen den Copyright-Inhaber, welche sich auf Schäden materieller oder ideeller Art beziehen, die durch die Nutzung oder Nichtnutzung der dargebotenen Informationen bzw. durch die Nutzung fehlerhafter und unvollständiger Informationen verursacht wurden, sind grundsätzlich ausgeschlossen, sofern seitens des Copyright-Inhabers kein nachweislich vorsätzliches oder grob fahrlässiges Verschulden vorliegt. Alle Angebote sind freibleibend und unverbindlich. Der Copyright-Inhaber behält es sich ausdrücklich vor, Teile der Seiten oder das gesamte Angebot ohne gesonderte Ankündigung zu verändern, zu ergänzen, zu löschen oder die Veröffentlichung zeitweise oder endgültig einzustellen.

2. Verweise und Links:

Der Copyright-Inhaber erklärt hiermit ausdrücklich, dass zum Zeitpunkt der Linksetzung auf fremde Internetseiten keine illegalen Inhalte auf den zu verlinkenden Seiten erkennbar waren. Auf die aktuelle und zukünftige Gestaltung, die Inhalte oder die Urheberschaft der gelinkten/verknüpften Seiten hat das Unternehmen keinerlei Einfluss. Deshalb distanziert er sich hiermit ausdrücklich von allen Inhalten aller gelinkten /verknüpften Seiten, die nach der Linksetzung verändert wurden. Für illegale, fehlerhafte oder unvollständige Inhalte und insbesondere für Schäden, die aus der Nutzung oder Nichtnutzung solcherart dargebotener Informationen entstehen, haftet allein der Anbieter der Seite, auf welche verwiesen wurde, nicht derjenige, der über Links auf die jeweilige Veröffentlichung lediglich verweist.

3. Urheberrecht:

Alle innerhalb des Angebotes genannten und ggf. durch Dritte geschützten Marken- und Warenzeichen unterliegen uneingeschränkt den Bestimmungen des jeweils gültigen Kennzeichenrechts und den Besitzrechten der jeweiligen eingetragenen Eigentümer. Allein aufgrund der bloßen Nennung ist nicht der Schluß zu ziehen, dass Markenzeichen nicht durch Rechte Dritter geschützt sind! Das Copyright für veröffentlichte, vom Autor selbst erstellte Objekte bleibt allein beim Autor der Seiten. Eine Vervielfältigung oder Verwendung solcher Grafiken, Tondokumente, Videosequenzen und Texte in anderen elektronischen oder gedruckten Publikationen ist ohne ausdrückliche Zustimmung des Copyright-Inhabers nicht gestattet.

4. Datenschutz:

Sofern innerhalb des Angebotes die Möglichkeit zur Eingabe persönlicher oder geschäftlicher Daten (E-Mail-Adressen, Namen, Anschriften) besteht, so erfolgt die Preisgabe dieser Daten seitens des Nutzers auf ausdrücklich freiwilliger Basis.

5. Rechtswirksamkeit dieses Haftungsausschlusses:

Dieser Haftungsausschluss ist als Teil des Angebotes zu betrachten, von dem aus auf diese Seite verwiesen wurde. Sofern Teile oder einzelne Formulierungen dieses Textes der geltenden Rechtslage nicht, nicht mehr oder nicht vollständig entsprechen sollten, bleiben die übrigen Teile des Dokumentes in ihrem Inhalt und ihrer Gültigkeit davon unberührt.