



Eclipse

Überblick und Grundlagen



Cegos Group

inspire
qualify
change



Inhalts- verzeichnis



Erstes arbeiten



Die Eclipse Workbench






Effizientes Arbeiten mit Eclipse



Erstes arbeiten



-  Entstehungsgeschichte
-  Download und Installation
-  Ein erstes Beispiel



Entstehungsgeschichte



Historie

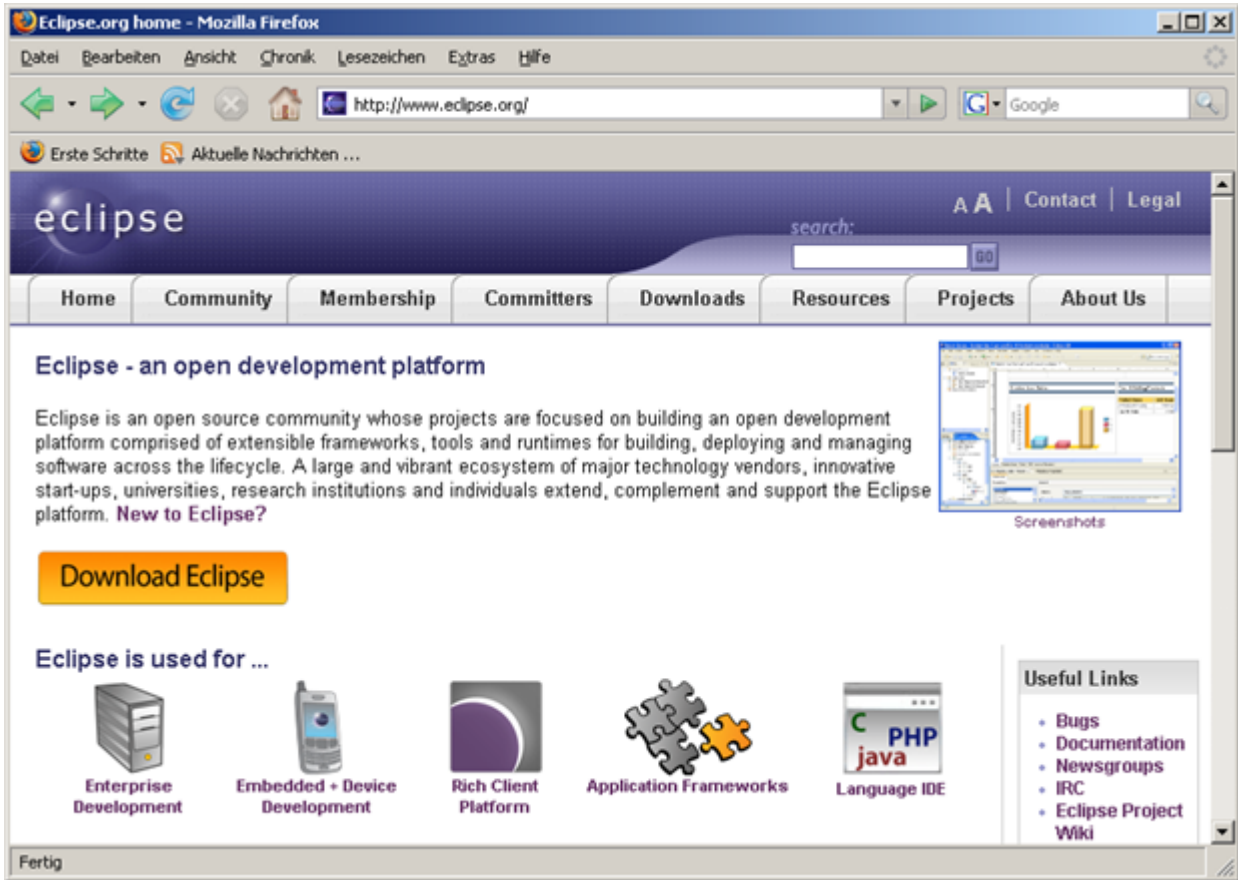
- Entwickelt von IBM
- Als Editor innerhalb des Websphere Studio Application Developer
- Nachfolger von VisualAge
- Jetzt in den Händen der Open Source Gemeinde



Download und Installation

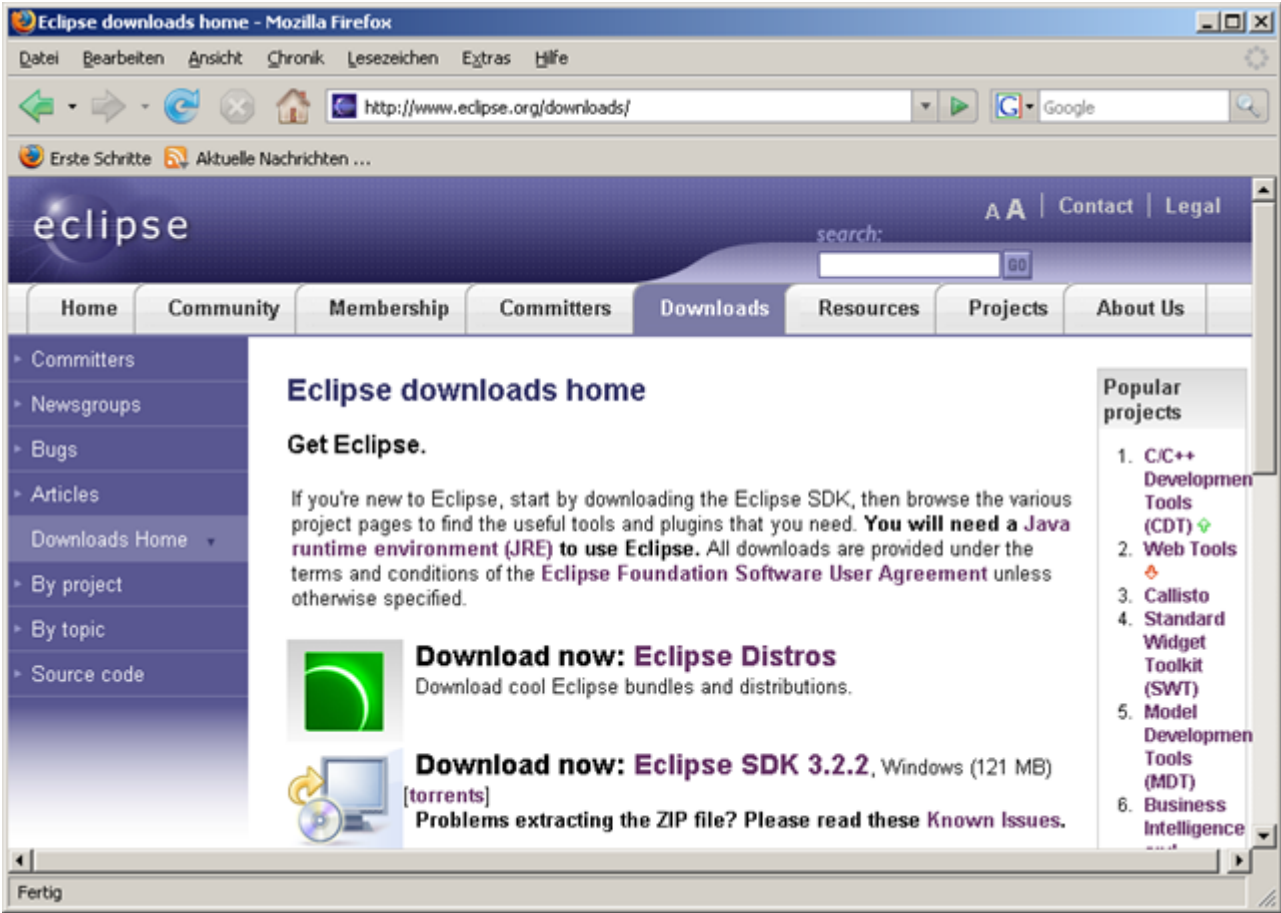
Eclipse Site

(www.eclipse.org)





Eclipse Project Downloads





Eclipse Project Downloads

- Download Sites
 - Haupt-Download Seite in Nordamerika
 - Mirror Sites in der restlichen Welt
- Downloads
 - Releases freigegebene Entwicklungsstände
 - Stable Builds stabiler aktueller Entwicklungsstand
 - Integration Builds Testversion für einen neuen Stable Build
 - Nightly Build aktueller Entwicklungsstand
 - Maintenance Build Wartungsstand der aktuellen Release

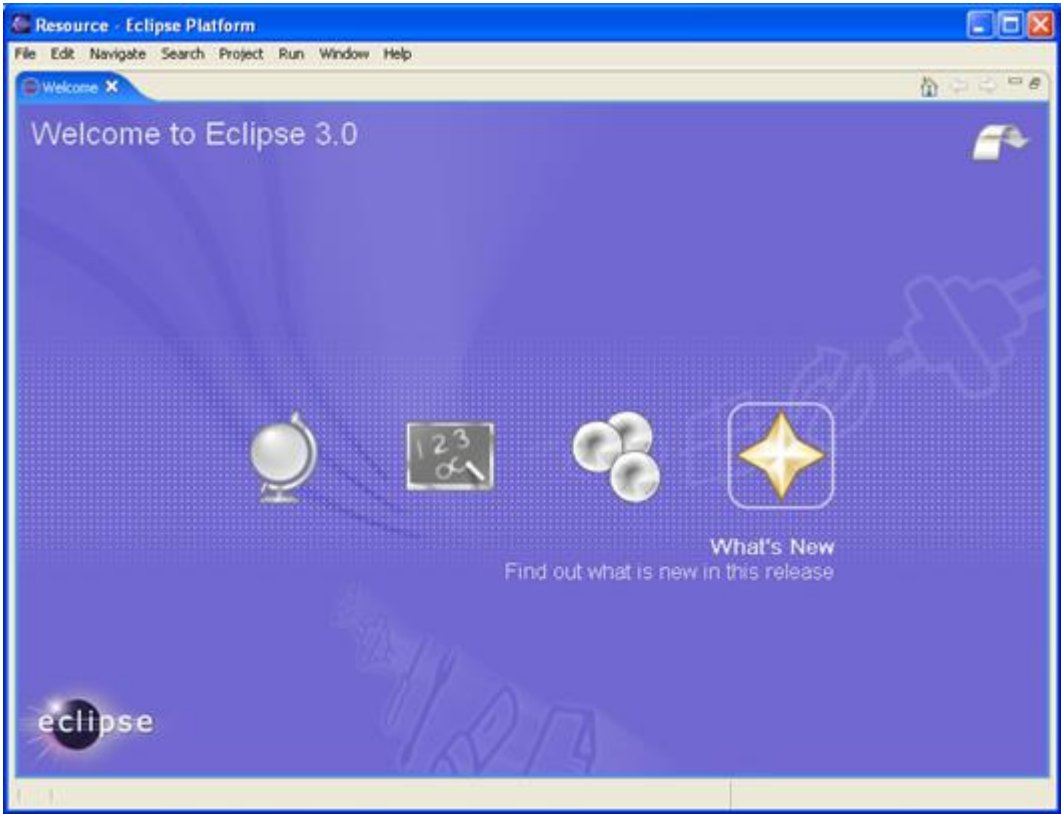


Installation

- Download
 - Herunterladen eines Releases oder anderen Entwicklungsstandes
- Installation
 - Entzippen der Dateien auf ein beliebiges Verzeichnis der Festplatte, z. B. c:
 - Eclipse befindet sich dann im Unterverzeichnis `c:/eclipse`
 - Ausführen des Programms `c:/eclipse/eclipse.exe`



Installation (Demo)





Startparameter

- Parameter
 - -configuration Konfigurationsdatei für den Programmstart
 - -data Workspace-Verzeichnis
 - -refresh Workspace mit dem Dateisystem beim Programmstart synchronisieren
 - -showlocation Installationsverzeichnis im Workspace anzeigen
 - -vm Angabe der zu verwendenden VM
 - -vmargs zusätzliche VM Parameter, z. B. -Xmx256M etc.



Ein erstes Beispiel



Ein erstes Programm: Hello World

- Zum Testen und zum Kennenlernen von Eclipse:
 - ein neues Projekt anlegen (z.B. ein Java-Projekt)
 - innerhalb des Projektes ein Paket anlegen
 - innerhalb des Paktes eine Klasse anlegen
 - Name HelloWorld
 - mit main-Methode
 - in der main-Methode eine Anweisung einfügen:
 - `System.out.println("Hello World");`
 - speichern und automatisch compilieren
 - ausführen



Die Eclipse Workbench



Aufbau



Perspektiven



Views



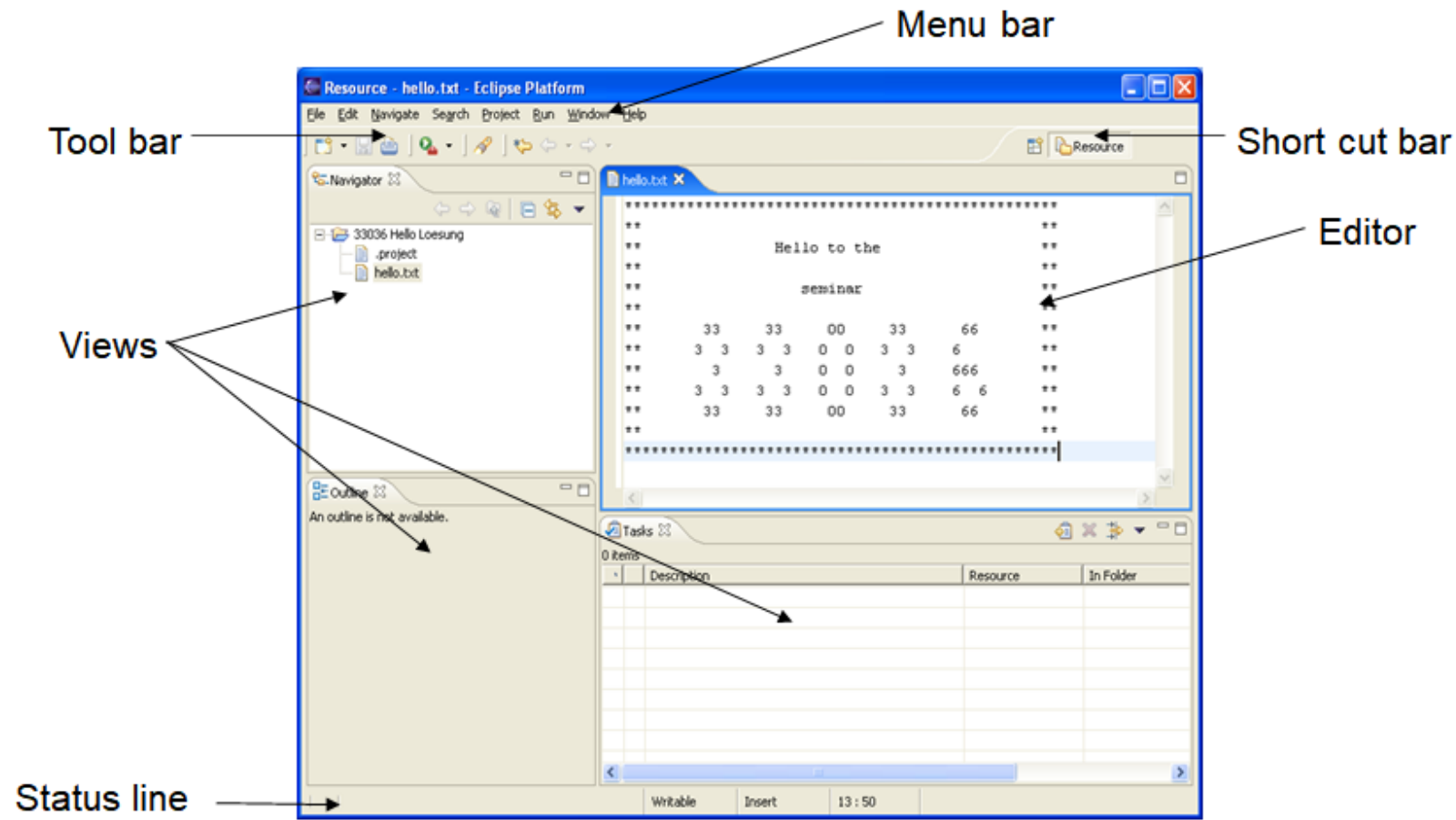
Editoren



Aufbau



Aufbau der Workbench





Perspektiven

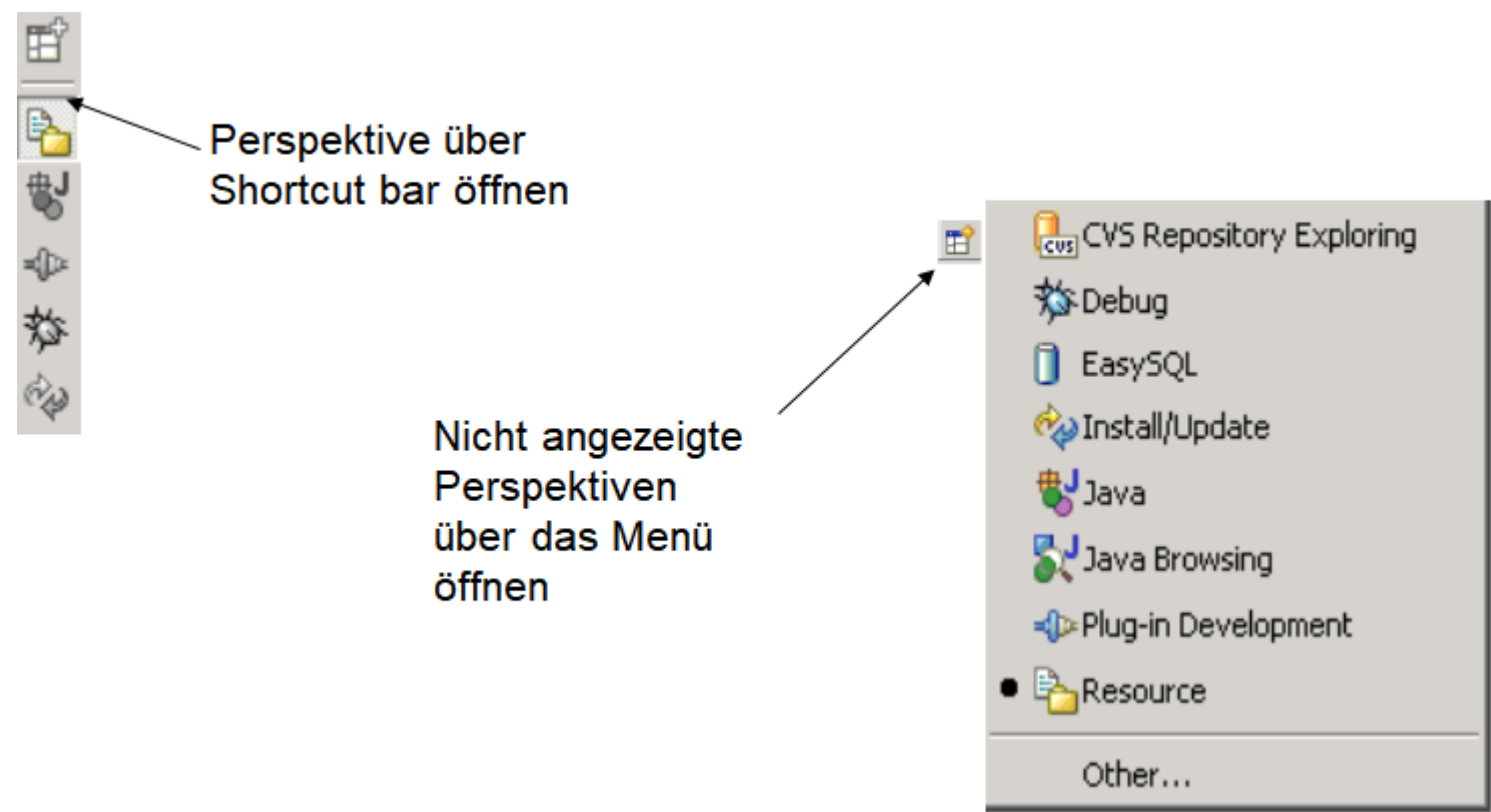


Perspektiven

- Was ist eine Perspektive?
 - Views
 - Editoren
 - Menüs
 - Toolbars
 - etc.

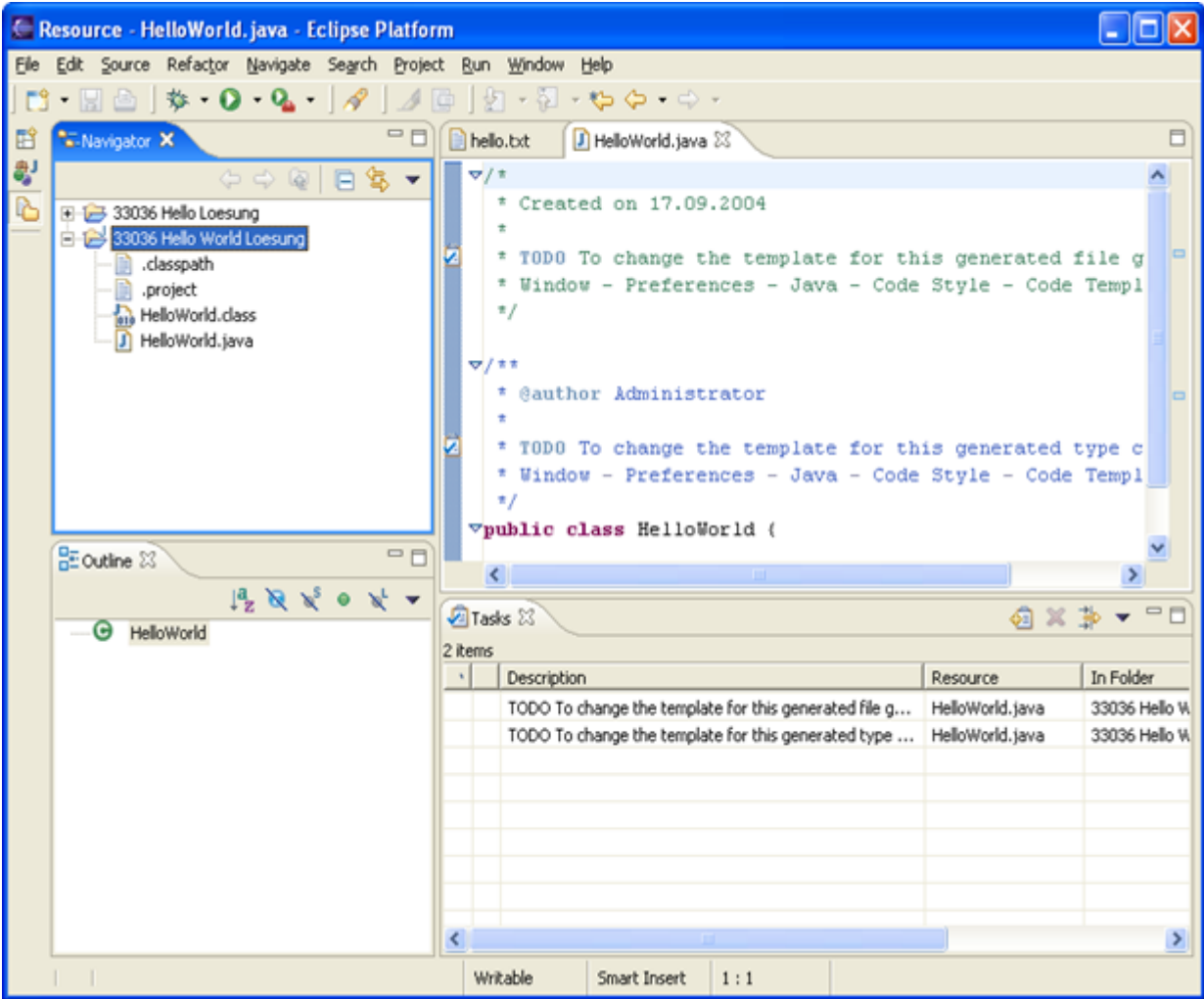
die zu einem Anwendungsgebiet gehören, z.B. Java, CVS o.ä.

Perspektive auswählen/öffnen

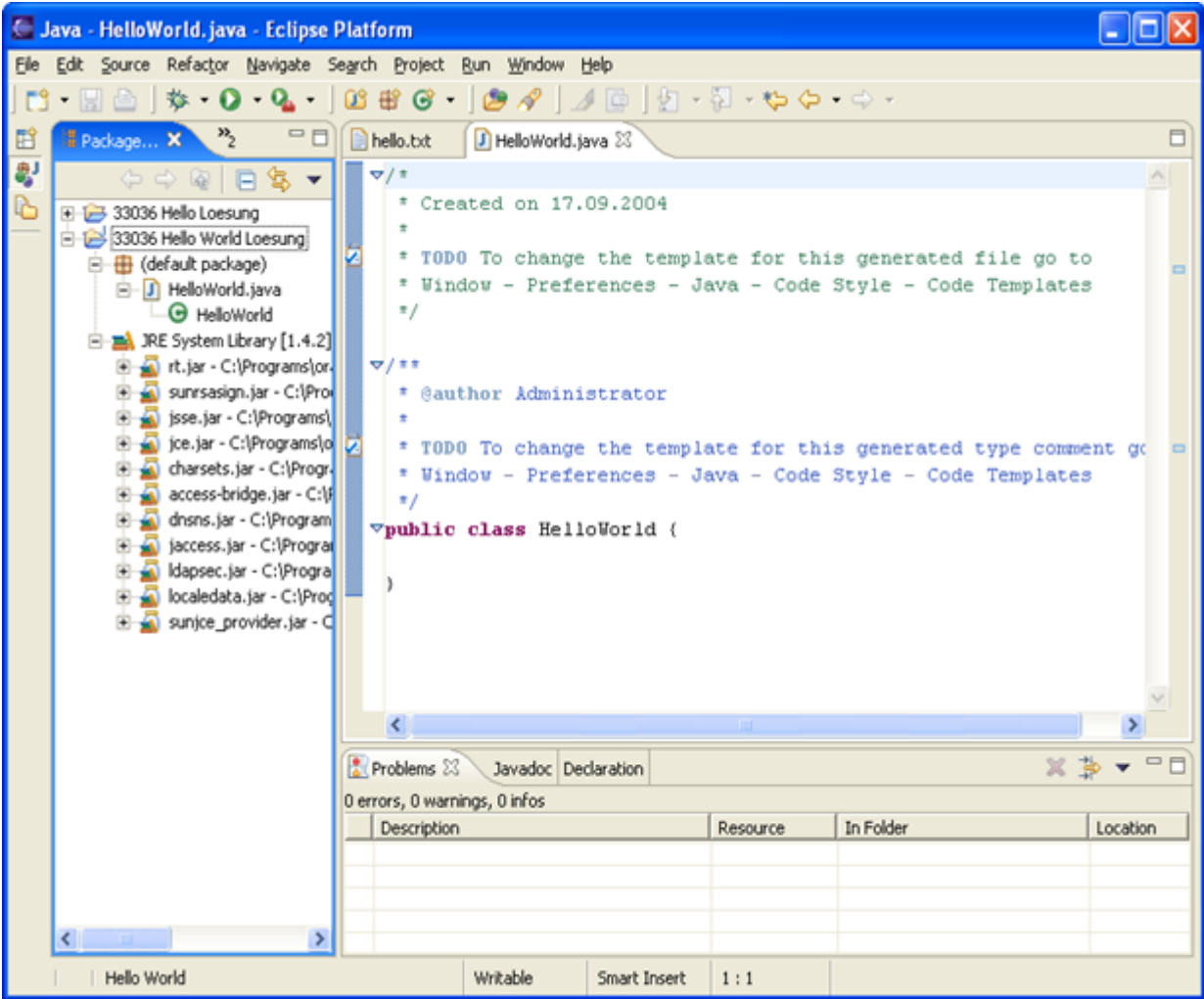




Resource Perspektive

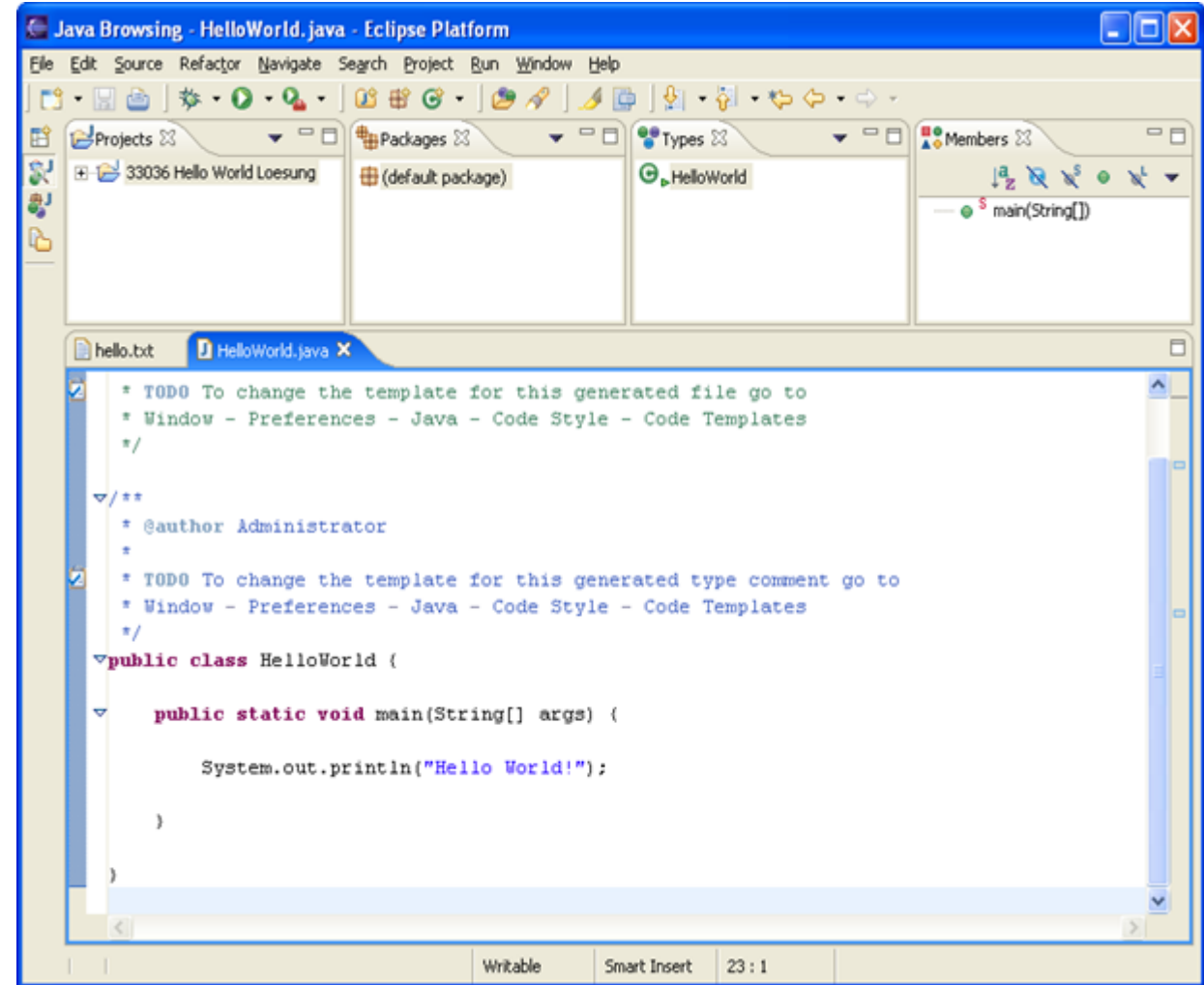


Java Perspektive



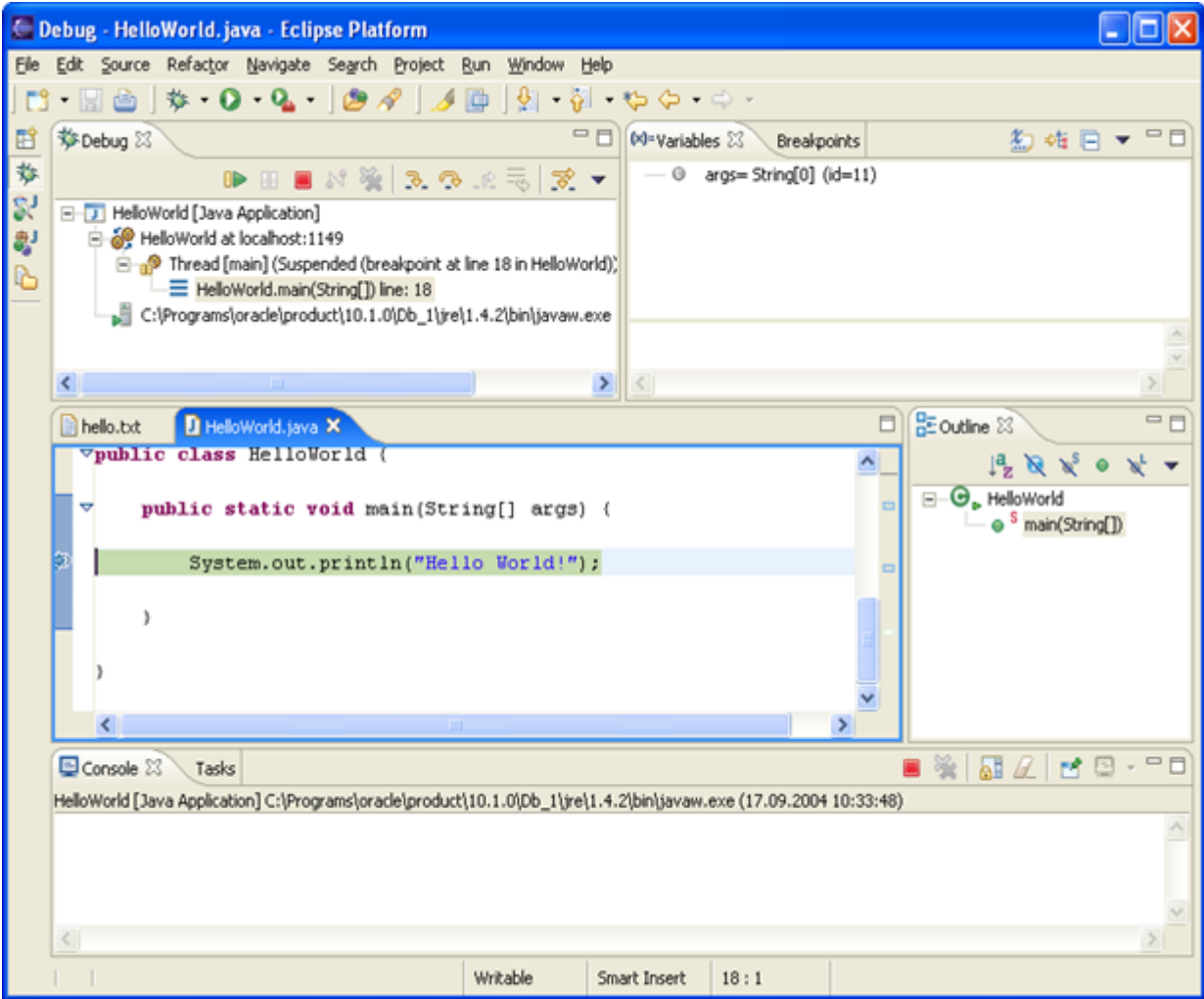


Java Browsing Perspektive





Debug Perspektive





Views

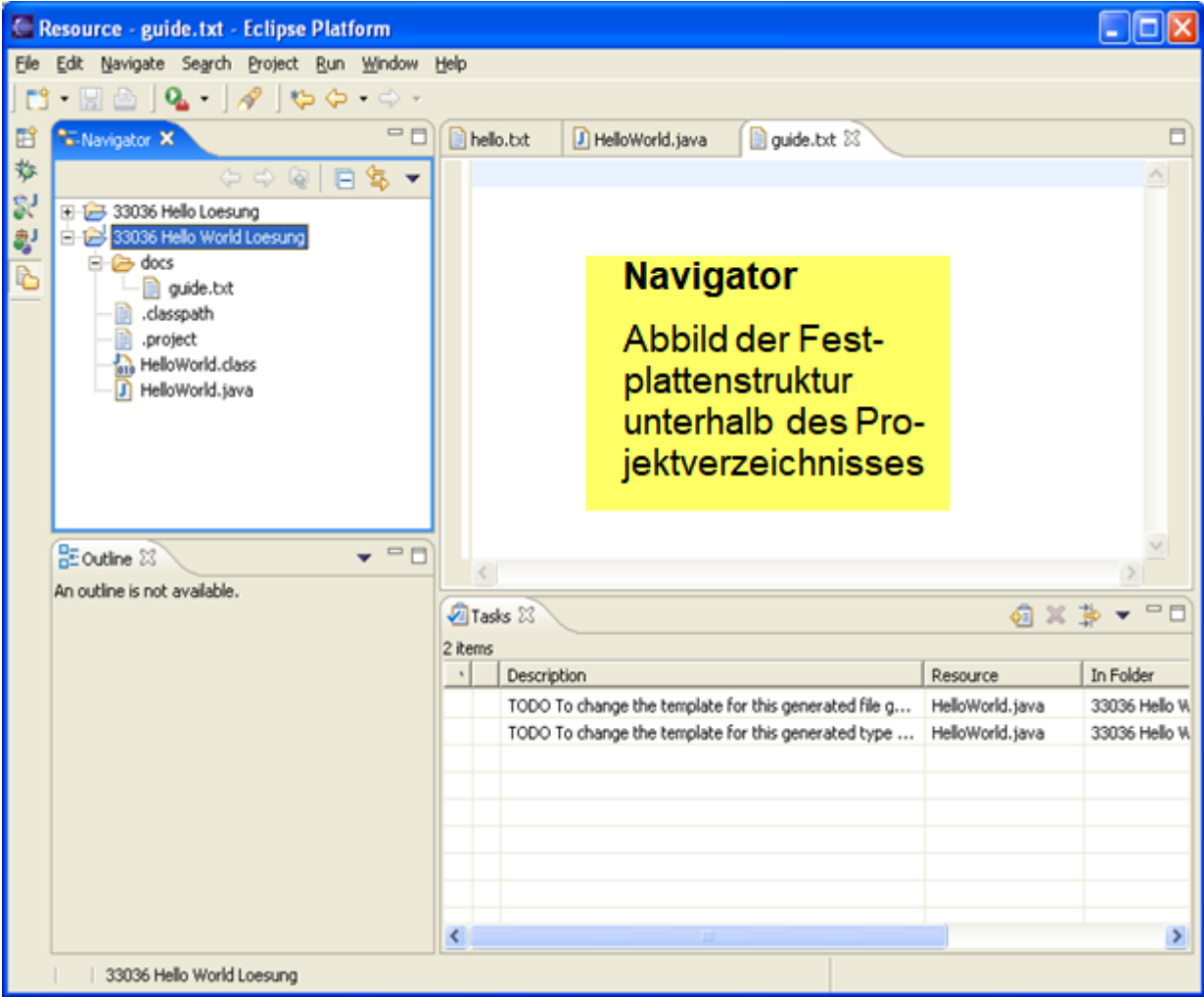


Views

- Was ist eine View?
 - Anzeige einer Resource, z.B. Java Klasse, XML Schema, etc.
 - Wird oft als Strukturanzeige verwendet, z. B. Dateibaum, Klassenaufbau, Klassenhierarchie etc.
 - Meistens Baum- oder Tabellenansicht.
- Beispiele
 - Outline View zeigt z. B. den Strukturbaum einer Java-Klasse
 - Tasks View zeigt Fehler und noch zu erledigende Aufgaben
 - Navigator zeigt den Verzeichnisbaum der Festplatte

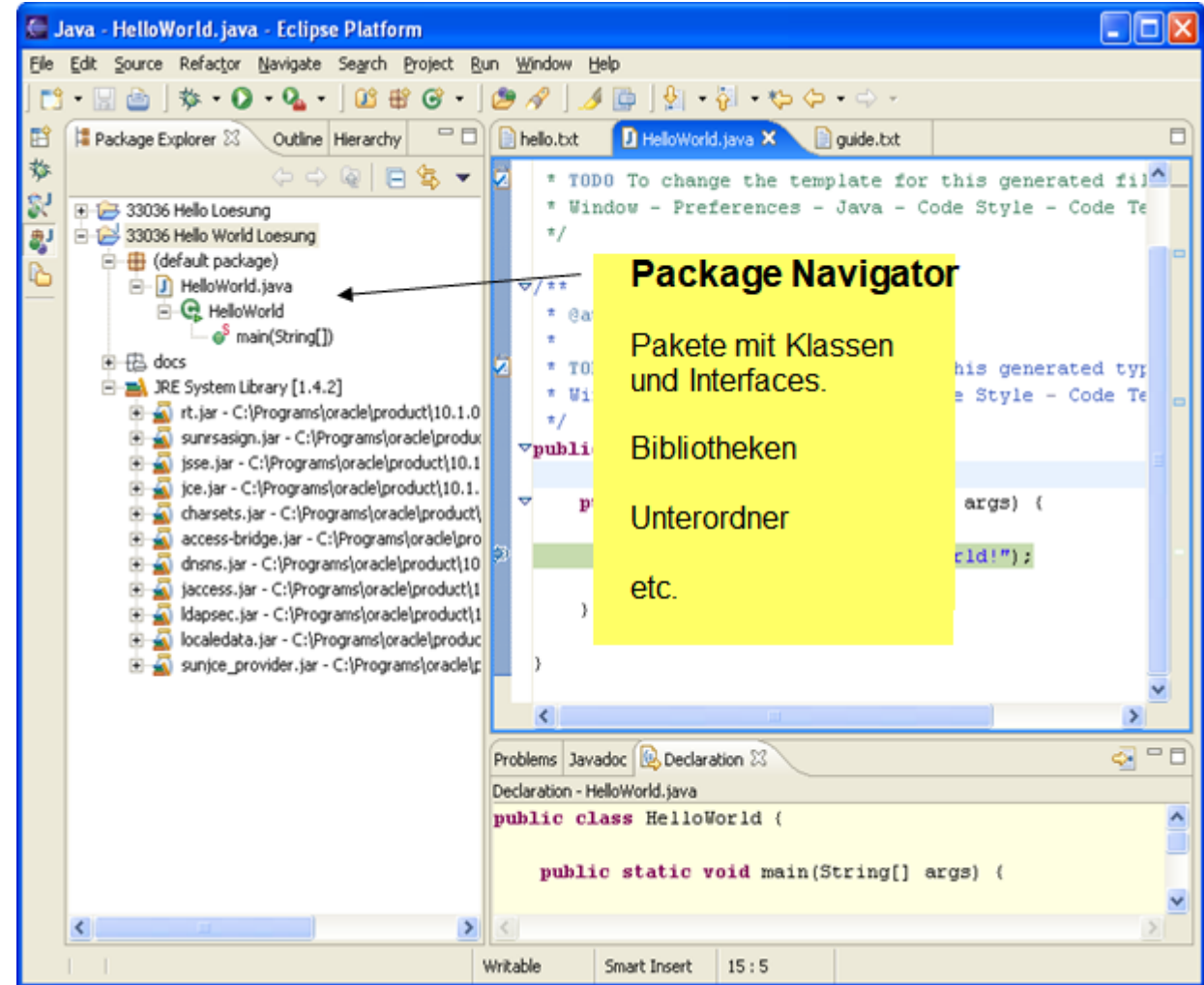


Navigator View





Package Navigator View





Editoren



Editoren

- Was ist ein Editor?
 - Auf eine Resource, z.B. Java-Datei, spezialisiertes Eingabeelement
 - Bietet spezielle, angepasste Editierfunktionalität, z.B. Syntax-Highlighting, Autoergänzung, Validierung etc.
- Beispiele
 - Java-Editor Editieren von Java Source-Dateien
 - plugin.xml Editieren der *plugin.xml* Datei
 - JSP-Editor Editieren von JSP-Dateien (extra Plugin!)
 - ...



Effizientes Arbeiten mit Eclipse



Die Standard-Perspektiven



Beispiele für Views und Editoren



Suchen mit Eclipse



Working Sets



Die Standard- Perspektiven



Standard Perspektiven

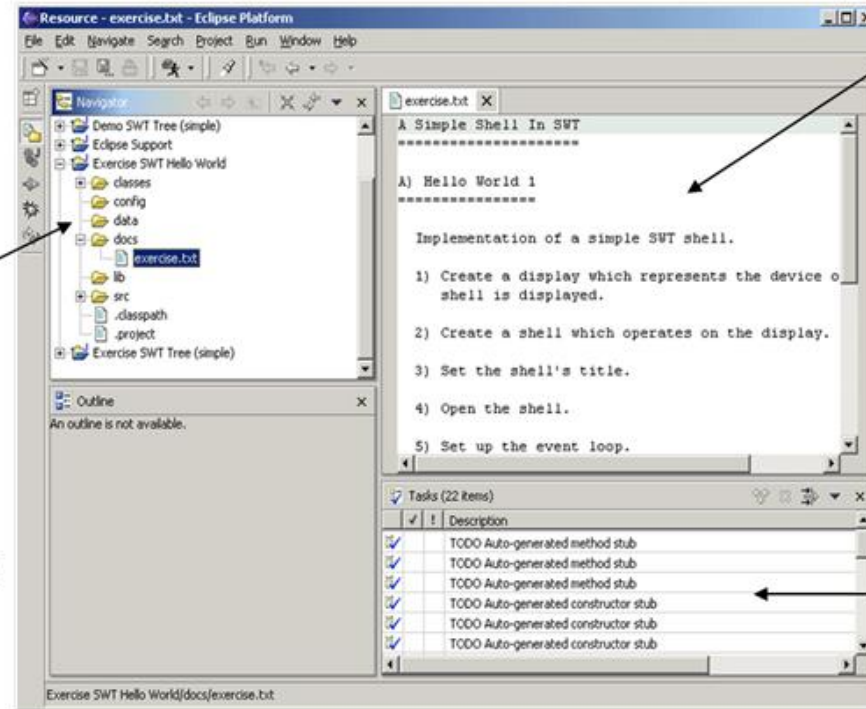
- Standard Perspektiven der Eclipse Umgebung
 - Resource allgemeine Projekte
 - Java Java Projekte
 - Java Browsing Browsen von Packages, Klassen und Interfaces
 - Java Type Hierarchy Anzeige der Java Typen Hierarchie
 - Debug Ausführung des Java Debuggers
 - CVS Repository Anbindung an CVS Versionsverwaltungen Exploring
 - Plugin Dev... Plugin Entwicklung
 - Team Synch... Abgleich mit Repository bei Teamarbeit



Resource Perspektive

Navigator

Anzeige der Projekte und der darunterliegenden Verzeichnisstruktur, wie sie sich auf der Festplatte befindet.



Editor

Anzeige der selektierten Resource. Es wird soweit vorhanden ein spezialisierter Editor, z.B. Java Editor, verwendet, ansonsten der Text Editor.

Tasks View

Anzeige der noch offenen Aufgaben



Java Perspektive

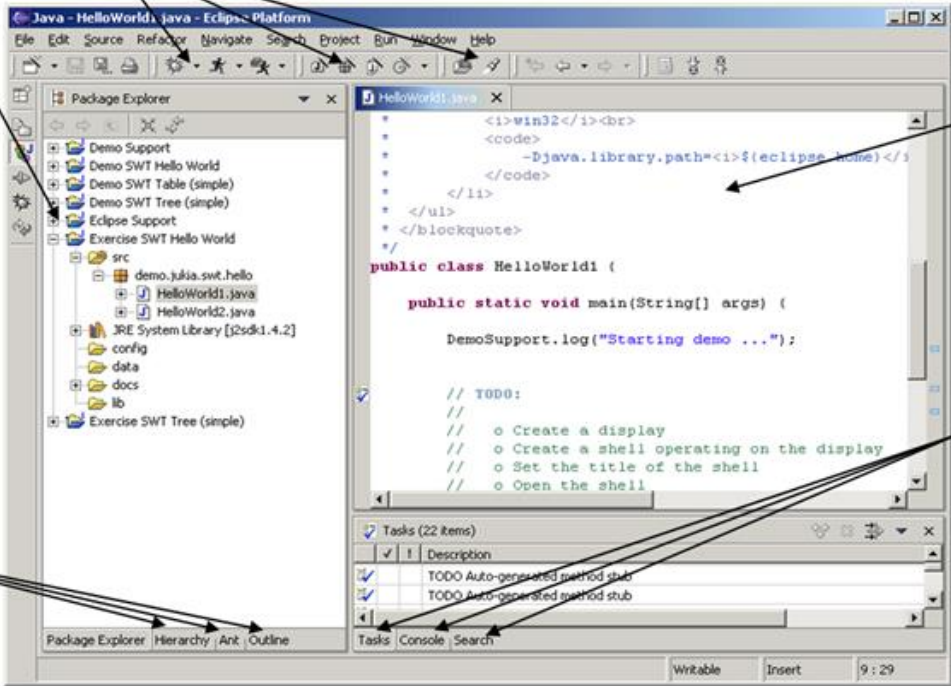
Package Explorer

Anzeige der Java Projekte. Spezielle Darstellung der Packages im Source Ordner und Elemente, die sich im Klassenpfad befinden.

Views

- Klassenhierarchie
- Ant Skripte
- Klassenstruktur

Toolbars speziell für Programmausführung, Debugging, Suche etc.



Java Editor

Anzeige der selektierten Java Datei mit Syntax Highlighting.

Views

- Tasks
- Konsole
- Suchergebnisse



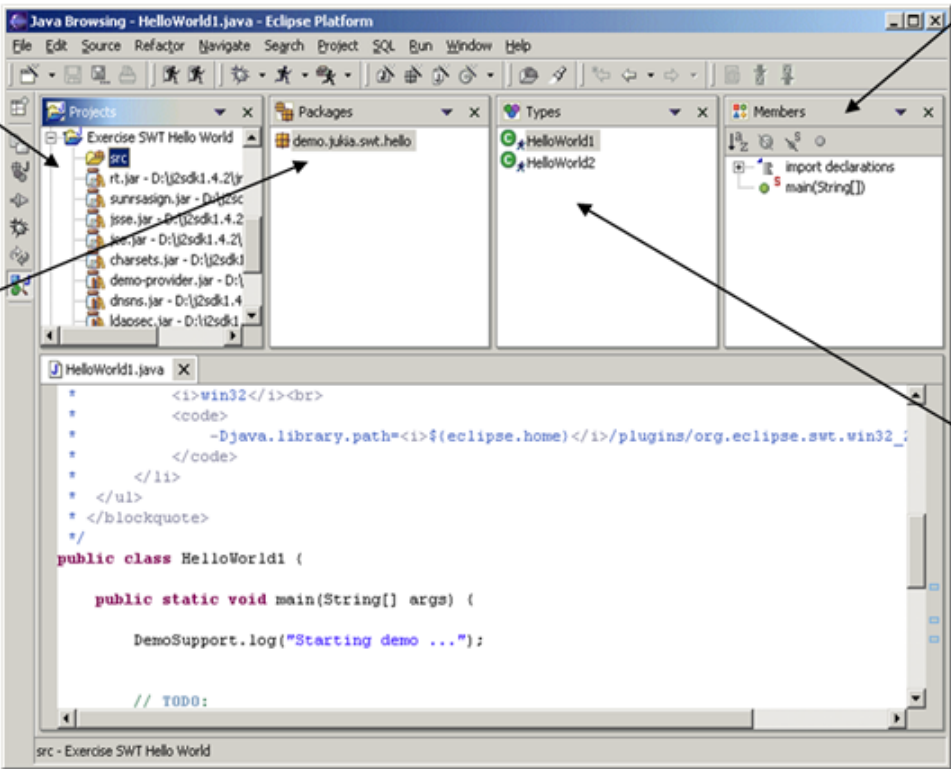
Java Browsing Perspektive

Java Projekte

Anzeige der vorhandenen Java Projekte.

Packages

Anzeige der in dem Projekt vorhandenen Packages.



Members

- Attribute
 - Methoden
 - etc.
- der selektierten Klassen oder Interfaces

Types

Anzeige der in dem selektierten Package vorhandenen Klassen und Interfaces.



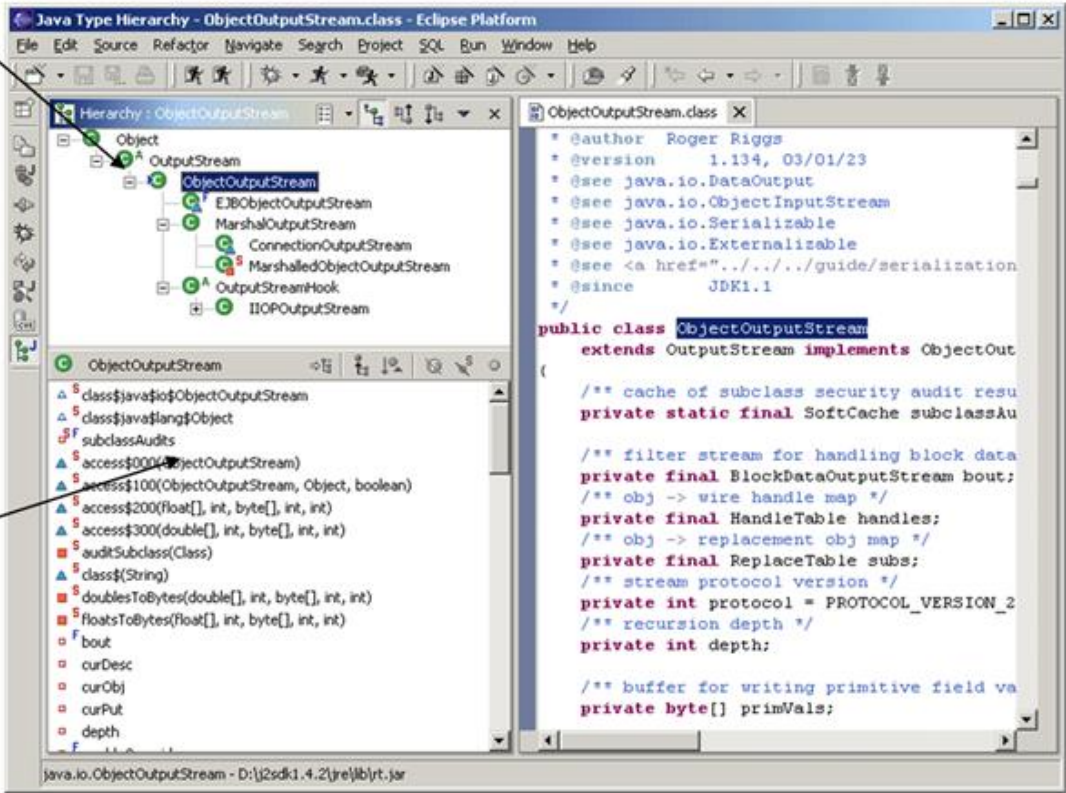
Java Type Hierarchy Perspektive

Hierarchy

Anzeige der Typhierarchie, d.h. Super- und Subklassen einer Klasse. Auch implementierte Interfaces können angezeigt werden.

Outline

Struktur der selktierten Klasse.





Debug Perspektive

Debug View

Laufende Programme, laufende Threads, Stacktraces.

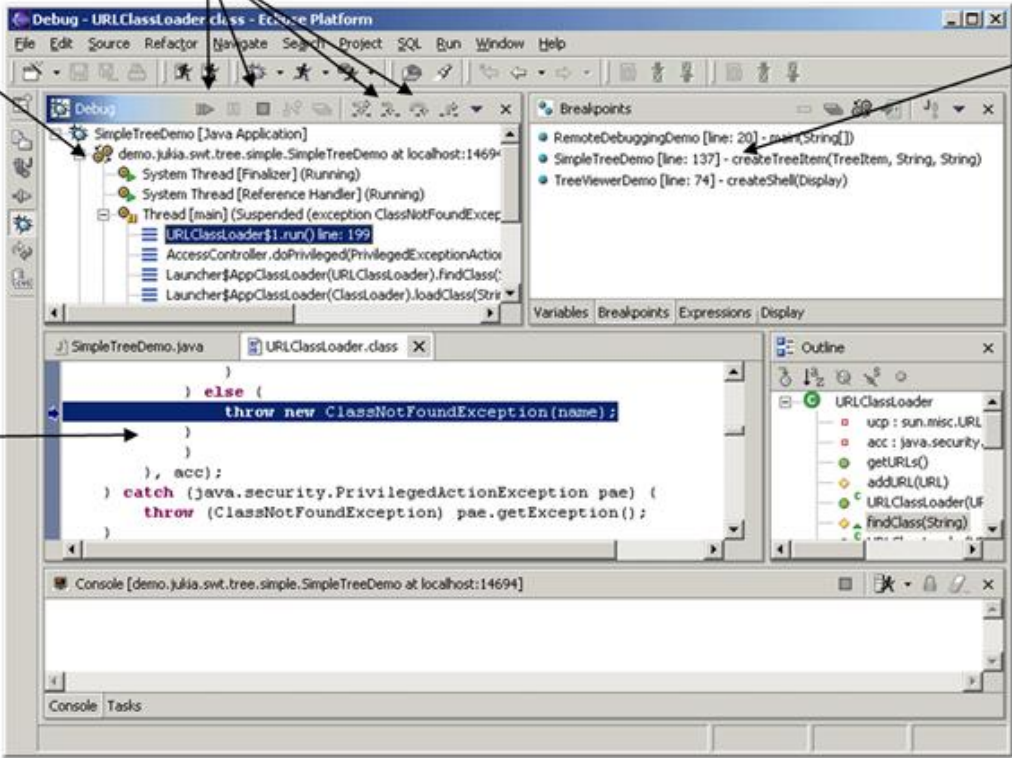
Source Code

Soweit verfügbar wird der Source Code angezeigt und die Stelle, an der sich der Debugger gerade befindet.

Debug Kontrolle

Breakpoints

Normale Breakpoints, bedingte Breakpoints, Exception-Breakpoints.

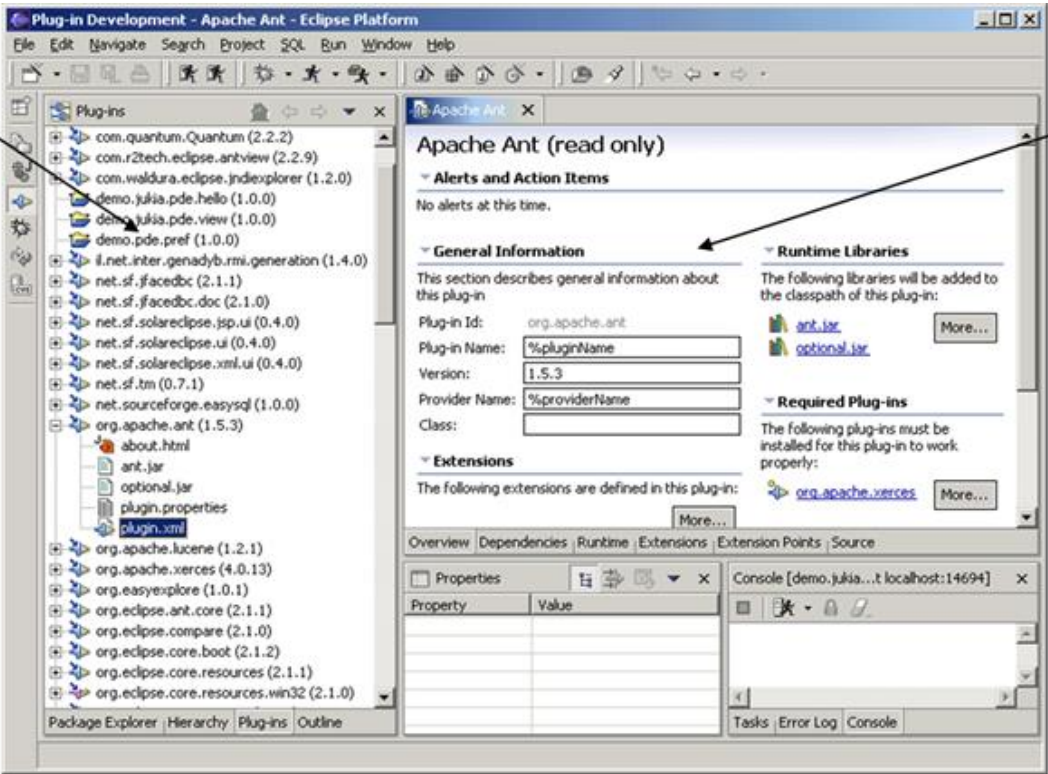




Plug-in Development Perspektive

Plug-ins

Anzeige der installierten Plug-ins und der Plug-in Projekte.



plugin.xml Editor

Spezieller Editor für das Editieren der *plugin.xml* Datei.



Beispiele für Views und Editoren



Standard Views

- Views
 - Navigator
 - Outline
 - Tasks
 - Console
 - Package Explorer
 - Hierarchy
 - Ant
 - Search
 - ...



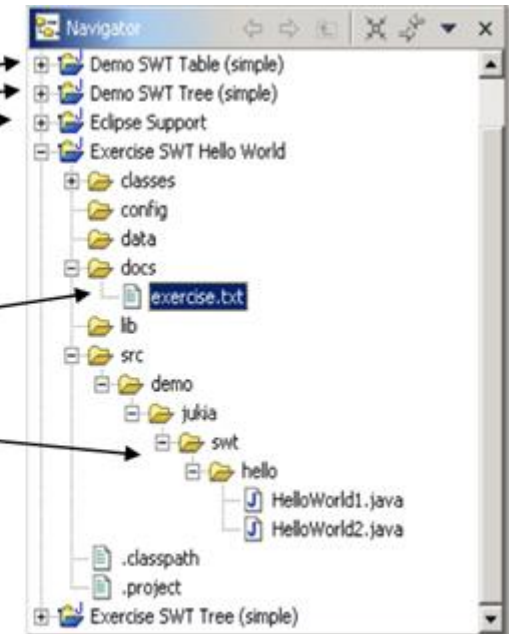
Navigator View

Projekte

Darstellung der eingerichteten Projekte.

Projekthalt

Unterhalb des Projektes werden alle Verzeichnisse und Dateien dargestellt. Sie entsprechen genau dem, was sich unterhalb des Projektordners auf der Festplatte befindet.





Outline View

- Outline
 - Strukturdarstellung (z.B. Java Klasse oder Interface)

Package

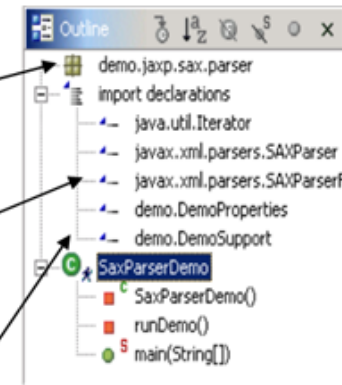
Anzeige des Packages, in dem die Klasse liegt.

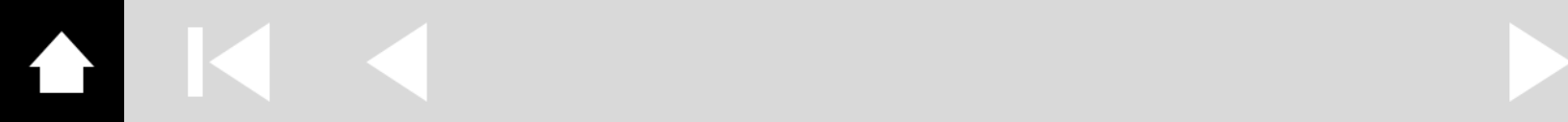
Imports

Anzeige der importierten Klassen und Interfaces.

Klasse / Interface

Darstellung der Klasse / des Interfaces mit Attributen und Methoden.





Tasks View

- Tasks
 - Fehler, Warnungen, Infos und TODOs.
 - Doppelklick öffnet die entsprechende Resource im Editor.

Typ
Fehler, Warnung, Info, TODO etc.

Meldung

Resource
Resource auf die sich die Meldung bezieht.

Meldung
Projekt und Unterordner, in dem sich die Resource befindet.

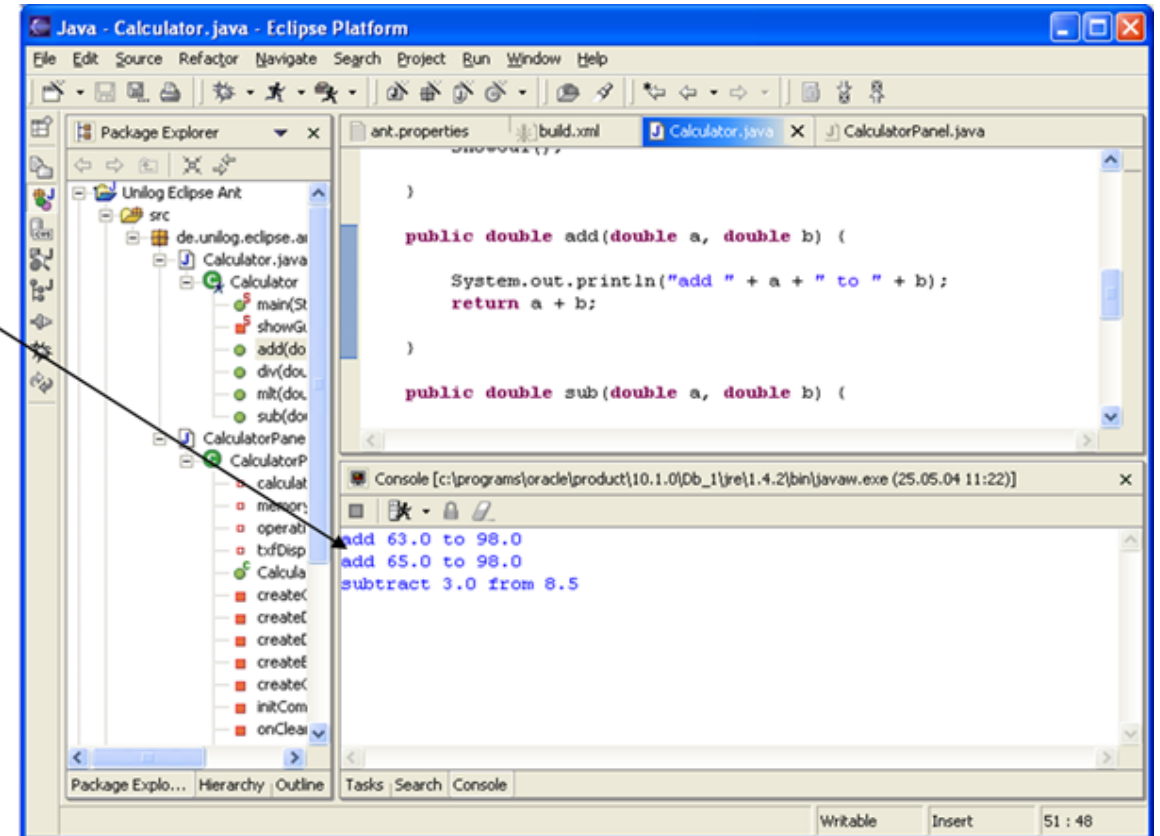
Zeilennummer

	✓ !	Description	Resource	In Folder	Loca...
		RMI builder: Successfully compiled 1 clas...	Demo JSE RMI		
	i	RMI builder: Successfully compiled 1 clas...	Demo RMI Regular		
	✗	Syntax error on token "}", "{" expected	ClipboardDemo.java	Demo AWT Clipboard/src/d...	line 572
	✗	Syntax error on keyword "void"; "interfa...	ClipboardDemo.java	Demo AWT Clipboard/src/d...	line 582
	✗	Unmatched bracket	ClipboardDemo.java	Demo AWT Clipboard/src/d...	line 627
	✓	TODO Auto-generated method stub	ListClipboardOwner.java	Demo AWT Clipboard/src/d...	line 29
	✓	TODO Auto-generated method stub	DemoContentProvider.java	Demo JFaces Viewer (Tree...	line 74



Console View

Aus- und Eingaben,
die normalerweise
auf die Konsole
gehen oder von der
Konsole kommen





Package Explorer View

Projekt

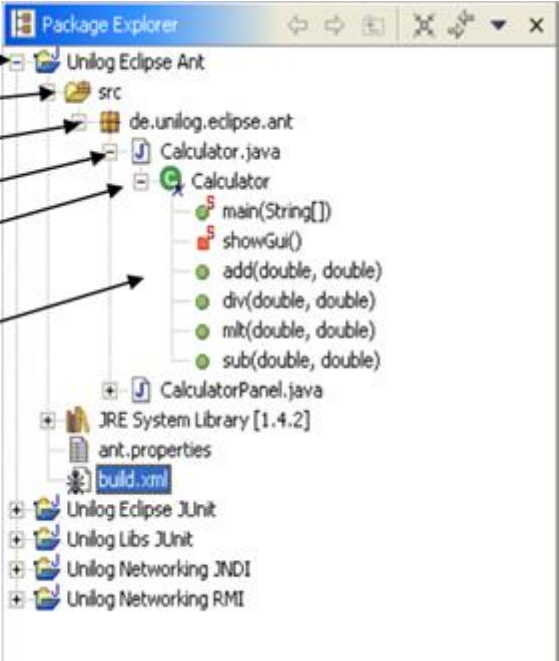
Quellcode Ordner

Package

Java-Datei

Klasse

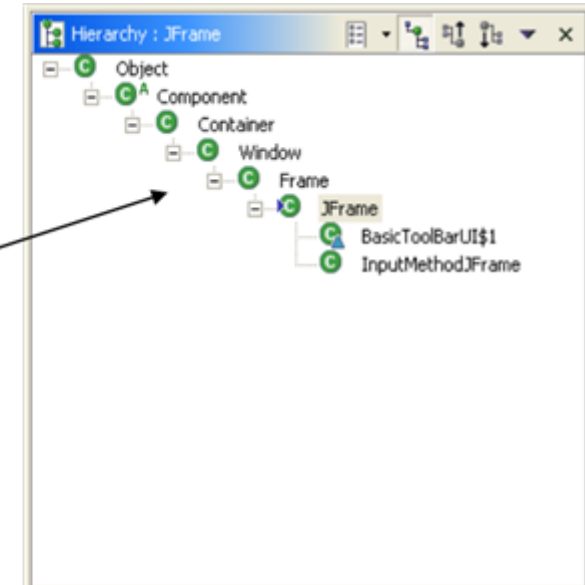
Methoden





Hierarchy View

Hierarchiebaum
der Klasse
JFrame





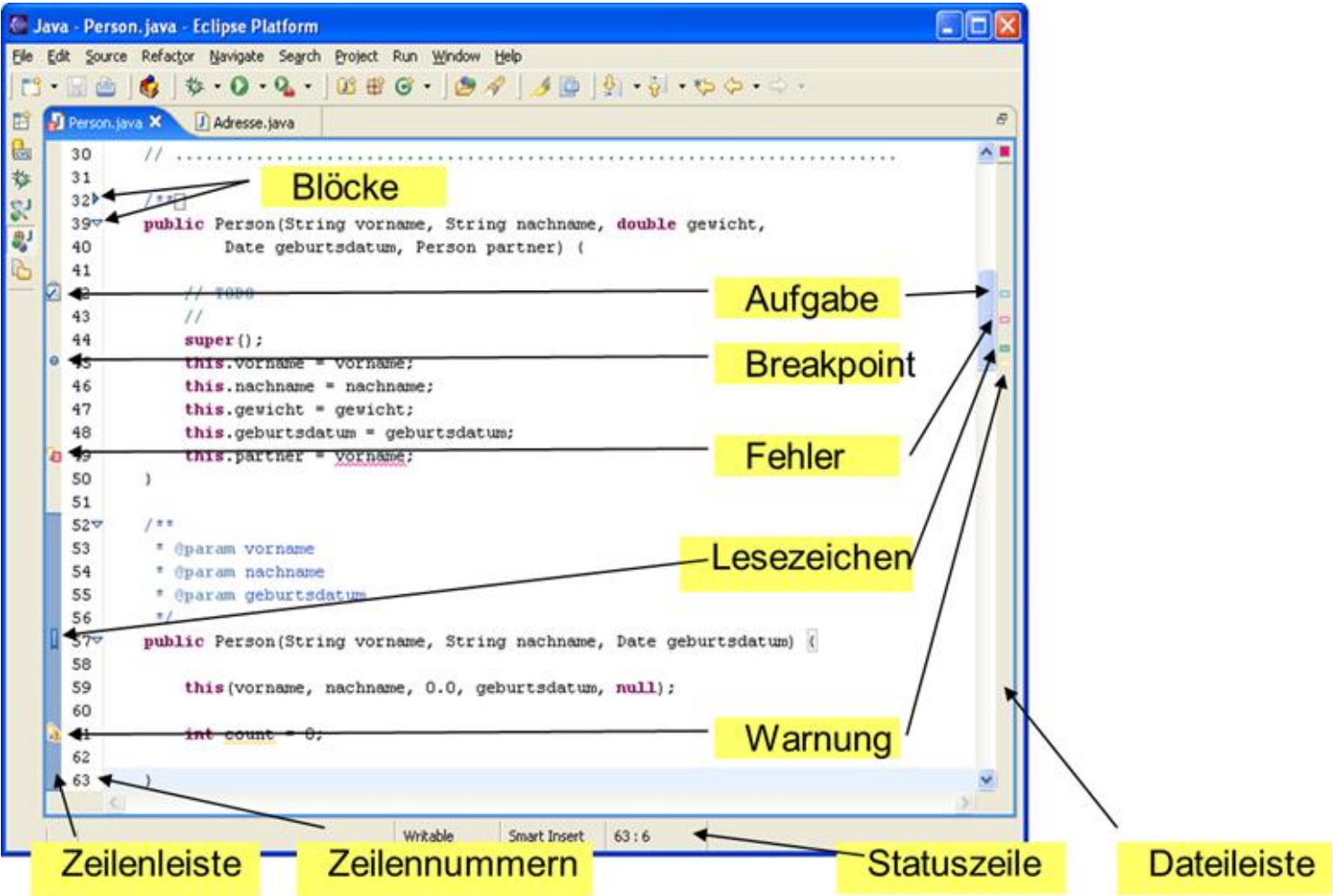
Standard Editoren

Standard-Editoren:

- Text-Editor
- Java-Editor
- plugin.xml Editor
- ...



Java Editor





Suchen mit Eclipse

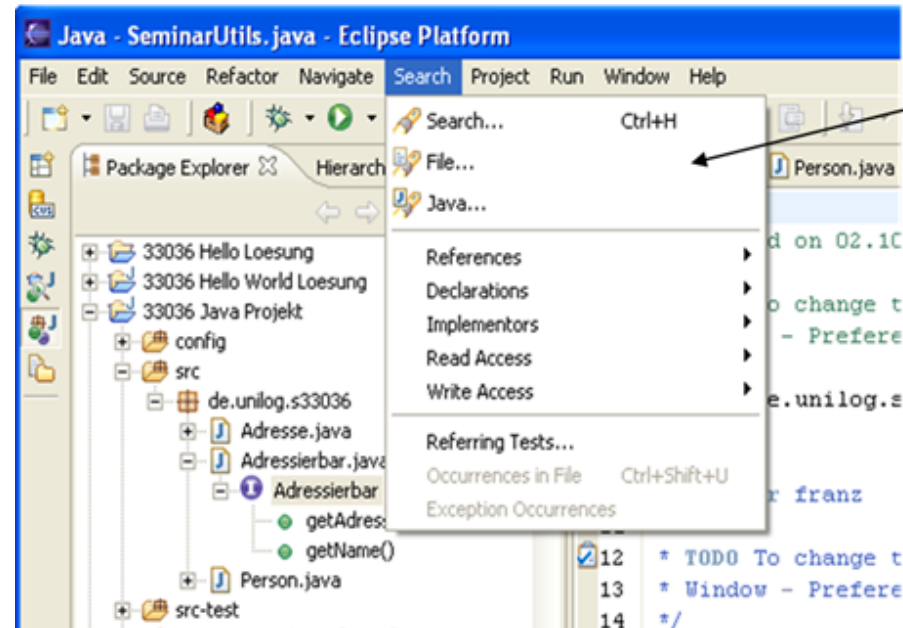


Inhalt

- Suche
 - Java Suche
 - Volltextsuche
 - Suche im Hilfesystem
 - Suche in der Dartei
- Working Sets
 - Definition
 - Anwendung
 - Verwaltung



Suchmenü

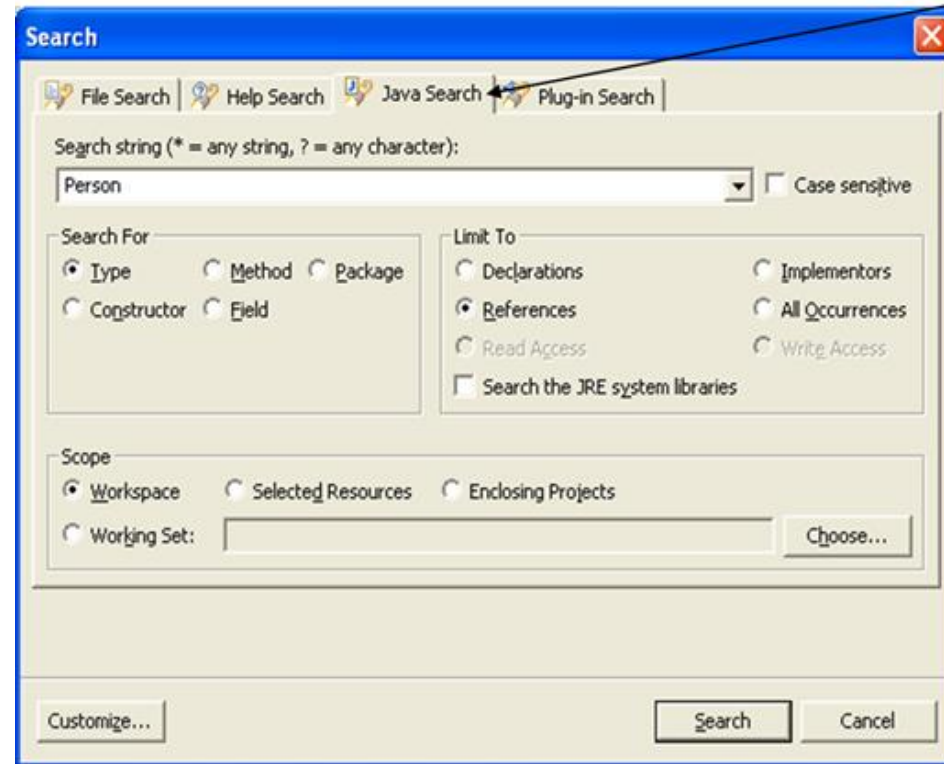


Menü Search

- Suchdialog
- Volltext Suche
- Java Suche (Deklarationen, Implementierungen, Referenzen usw.)
- Suche im Hilfesystem



Java Suche



Dialog Java Search

- Suche nach Typen, Feldern, Methoden, Konstruktoren, Paketen
- Suche nach Deklaration und/oder Referenzen
- Suchbereich (Workspace, ausgewählte Ressourcen, abhängige Projekte, Working Sets)

Menü

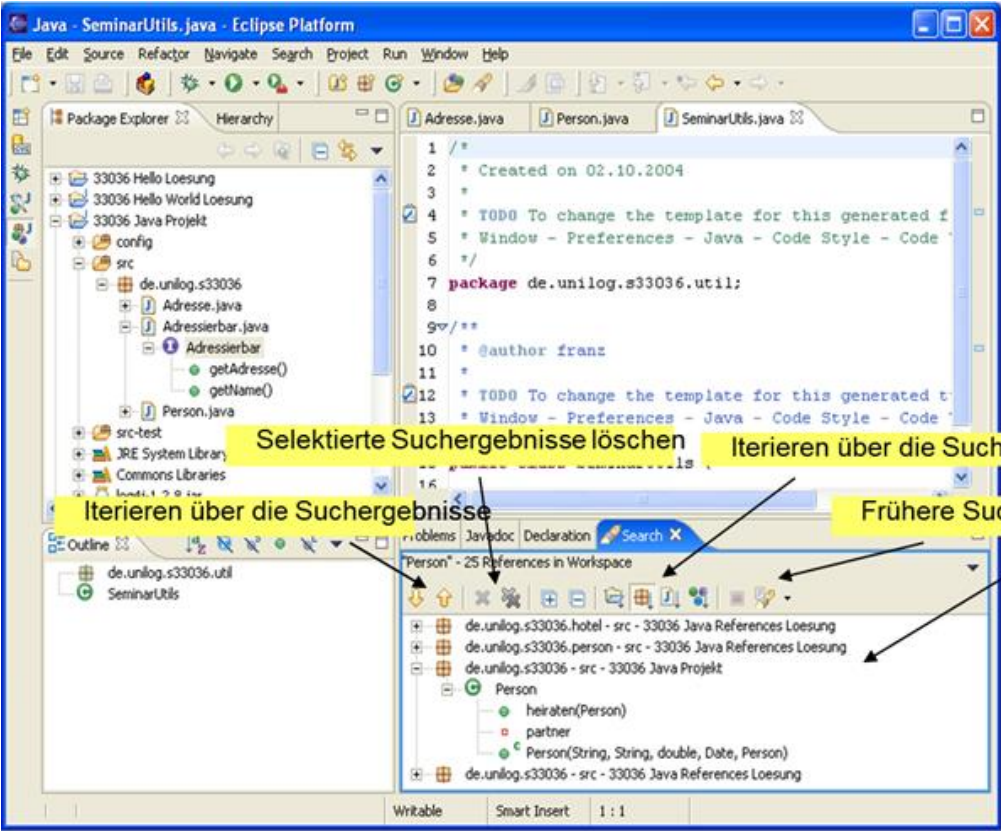
Search
 > Search ...
 > File ...
 > Java ...
 etc.

Popupmenü

> References
 > Declarations
 > Implementors



Suchergebnisse



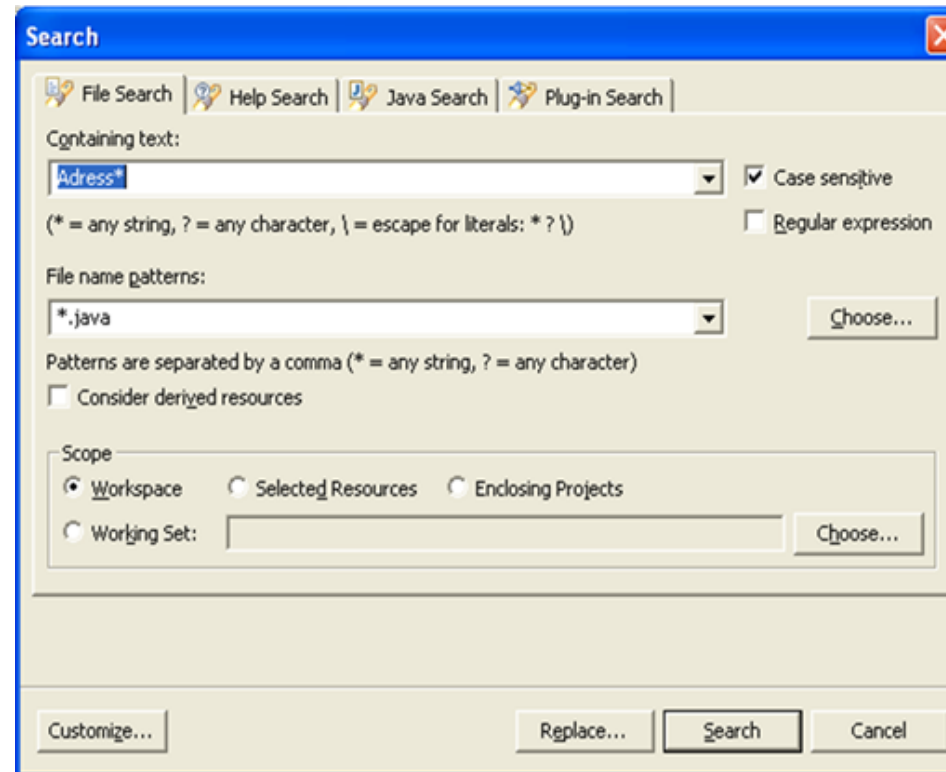
Ergebnisse

Die Ergebnisse werden in einem eigenen View angezeigt.

Durch Doppelklick kann die gefundene Codestelle geöffnet werden.



Volltextsuche



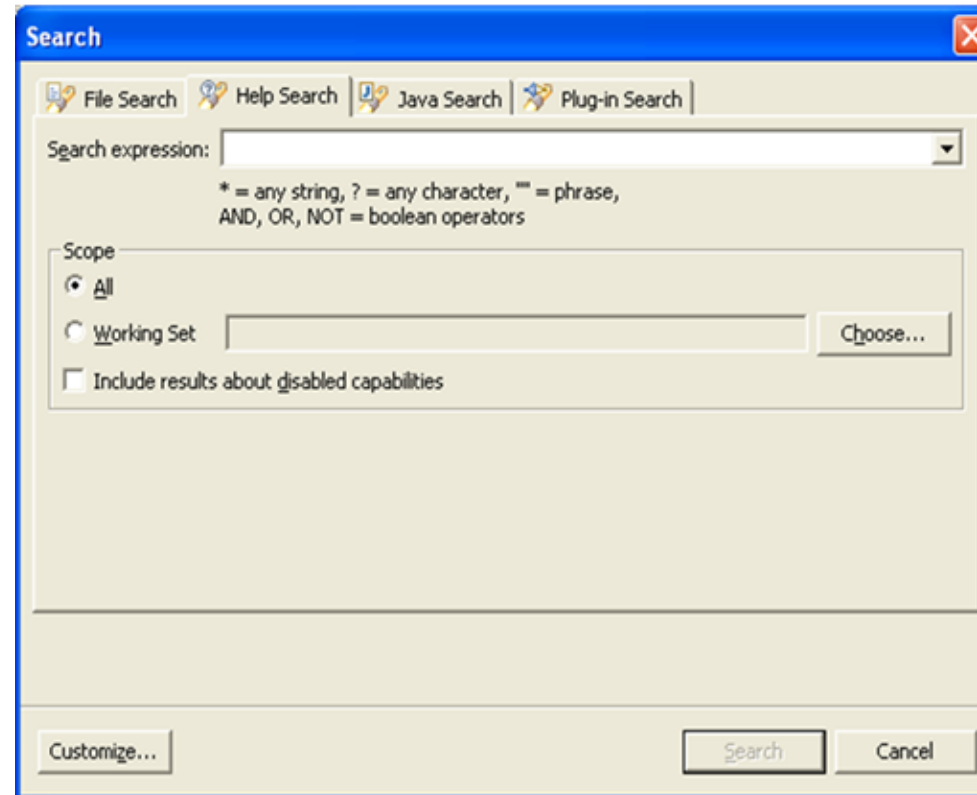
Dialog File Search

- Textmuster (auch Regular Expressions)
- Dateinamensmuster
- Suchbereiche

Volltextsuche ist langsamer als die Java Suche!



Suche in der Hilfe

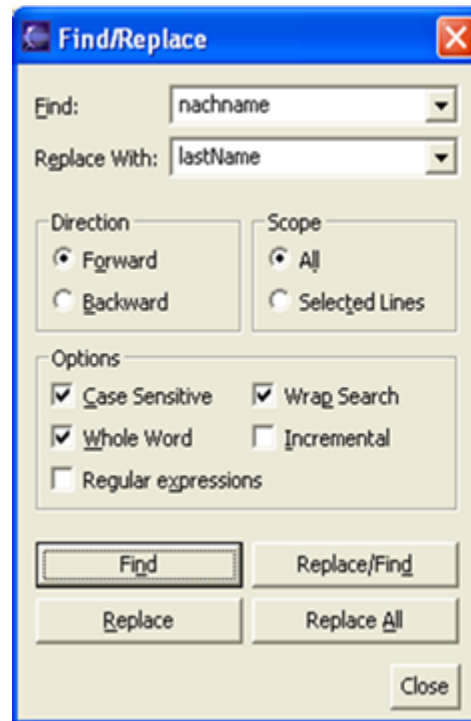


Dialog Help Search

- Schlüsselwörter
- Suchbereiche



Suche im Quelltext

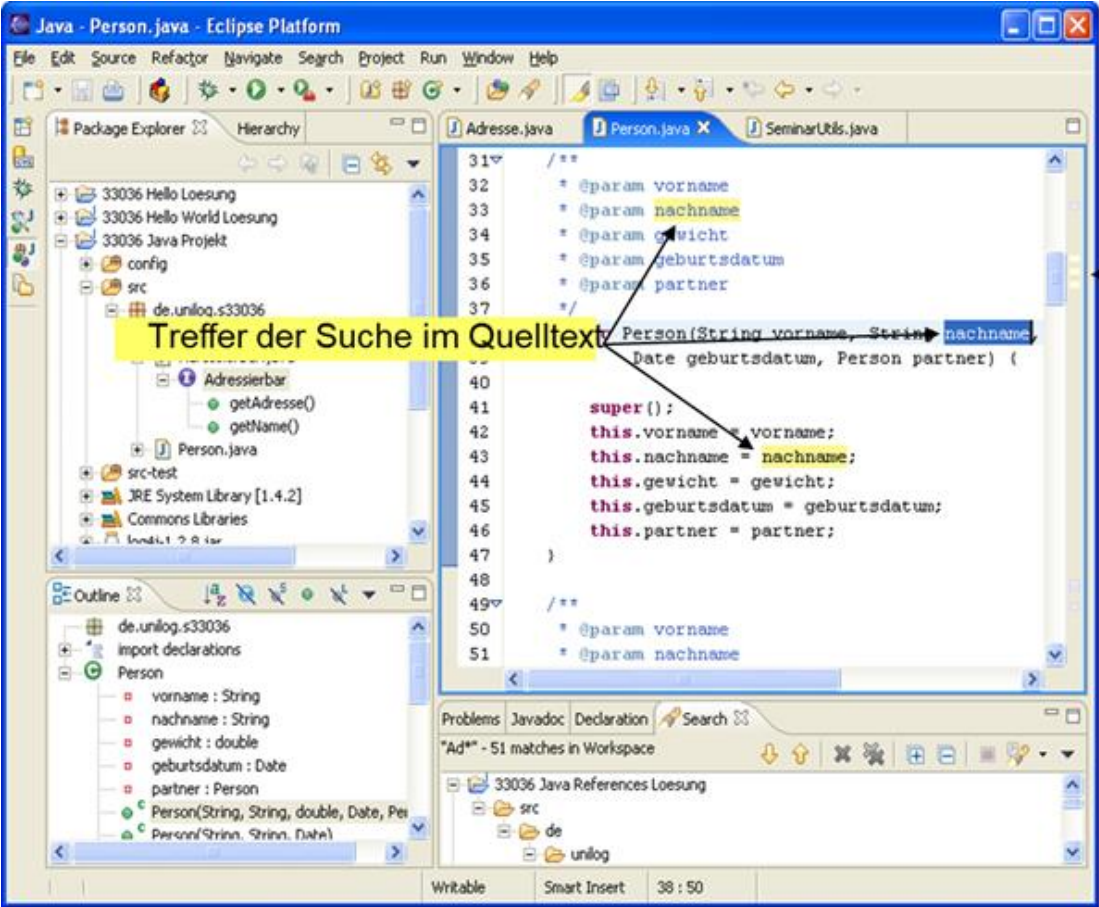


Suchen / Ersetzen

- Suchrichtung (vorwärts/rückwärts)
- Suchbereich (ganze Datei oder selektierter Bereich)
- Optionen
- Suchen / Ersetzen



Fundstellen

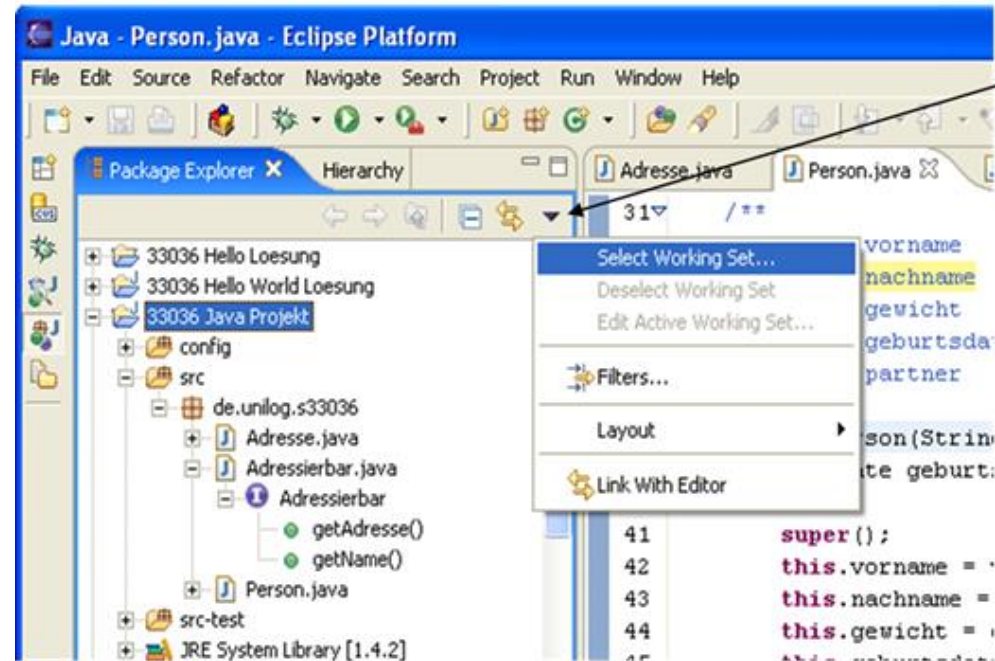




Working Sets



Working Set definieren



Package Explorer

- Working Sets definieren, löschen, editieren und auswählen.
- Angezeigte Projekte einschränken

Working Set

- Auswahl von mehreren Projekten oder auch nur von Teilen eines Projekts.
- Benutzerdefinierte Suchbereiche