

LITERATURLISTE • IDE

Debugging

Viele IDEs bieten Tools an, um den Code zu debuggen. Dabei lassen sich Haltepunkte definieren, an denen die Ausführung angehalten wird und der Zustand der Applikation (Variablen, Klassen etc.) untersucht werden kann.

<https://www.jetbrains.com/help/idea/debugging-code.html>

<https://www.eclipse.org/jetty/documentation/current/debugging-with-eclipse.html>

<https://netbeans.org/features/java/debugger.html>

<https://code.visualstudio.com/docs/editor/debugging>

Erscheinungsbild

In der Regel kann das Erscheinungsbild einer IDE – inklusive verwendeter Schriften und Farben – den individuellen Vorlieben und Bedürfnissen angepasst werden, z. B. um eine lesbare Darstellung für Beamer-Präsentationen anzubieten.

<https://www.jetbrains.com/help/idea/settings-appearance.html?section=Windows%20or%20Linux>

<http://www.eclipsecolorthemes.org>

<https://jaxenter.com/netbeans/the-complete-guide-to-tuning-the-appearance-of-netbeans>

<https://code.visualstudio.com/docs/getstarted/themes>

Git mit der IDE verwenden

Häufig bieten IDEs die Möglichkeit, Änderungen am Code direkt in ein Git-Repository zu pushen, ohne dass die Befehle zeitaufwändig in einer Shell eingetippt werden müssen.

<https://www.jetbrains.com/help/idea/using-git-integration.html?section=Windows%20or%20Linux>

<https://jaxenter.de/kolumne-eclipse-git-mit-eclipse-62447>

<https://netbeans.org/kb/docs/ide/git.html>

<https://www.microsoft.com/de-de/techwiese/know-how/visual-studio-code-07-git-als-quellcodeverwaltung-einsetzen.aspx>

IDE auswählen

Bei der Auswahl einer geeigneten IDE sollte neben den Punkten Systemvoraussetzungen / Kosten etc. vor allem darauf geachtet werden, ob sie geeignet ist, die verwendeten Programmiersprachen / Frameworks durch Syntax-Highlighting, Linting, Debugging, Compiling etc. zu unterstützen.

<https://www.it-talents.de/blog/it-talents/die-5-besten-ides-die-jedes-it-talent-kennen-sollte>

<https://www.heise.de/newsticker/meldung/Test-Microsoft-Visual-Studio-Code-ein-Editor-mit-integriertem-Debugger-4026553.html>

Klassen- und Methodenstruktur anzeigen lassen

Eine IDE kann die Struktur der verwendeten Klassen und Methoden aufschlüsseln.

<https://www.jetbrains.com/help/idea/viewing-structure-of-a-source-file.html>

<https://help.eclipse.org/neon/index.jsp?topic=%2Forg.eclipse.platform.doc>

[user%2Fconcepts%2Foutline.htm](#)

<https://stackoverflow.com/questions/4756770/where-is-the-class-outline-view-in-netbeans-for-php>

https://code.visualstudio.com/updates/v1_25#_outline-view

Projekt erstellen

Das simpelste Szenario ist es, wenn ein Quellcode lokal auf der Festplatte gespeichert wird und nicht mit verschiedenen Entwicklern gemeinsam am Code gearbeitet werden muss.

Arbeiten mehrere Entwickler gemeinsam an einer Code-Basis ist es notwendig, den Quellcode auf einem Server abzulegen, auf den alle beteiligten Entwickler mittels eines Tools wie z. B. Git, ihre Änderungen einspielen.

<https://www.jetbrains.com/help/idea/creating-and-managing-projects.html>

[https://help.eclipse.org/kepler/index.jsp?topic=%2Forg.eclipse.jdt.doc.](https://help.eclipse.org/kepler/index.jsp?topic=%2Forg.eclipse.jdt.doc.user%2FgettingStarted%2Fqs-ProjectConfiguration.htm)

[user%2FgettingStarted%2Fqs-ProjectConfiguration.htm](https://help.eclipse.org/kepler/index.jsp?topic=%2Forg.eclipse.jdt.doc.user%2FgettingStarted%2Fqs-ProjectConfiguration.htm)

<https://netbeans.org/kb/74/java/project-setup.html>

<https://www.microsoft.com/de-de/techwiese/know-how/node-js-projekt-mit-typescript-in-visual-studio-code-erstellen.aspx>

Shell

Die Shell – auch Terminal genannt – ist ein Kommandozeilen-Interpreter. Viele IDEs stellen eine integrierte Shell zur Verfügung.

<https://www.jetbrains.com/help/idea/settings-tools-terminal.html>

<https://marketplace.eclipse.org/content/easyshell>

<http://plugins.netbeans.org/plugin/24596/simple-shell>

<https://code.visualstudio.com/docs/editor/integrated-terminal>

Suche im Quellcode

Es gibt verschiedene Voreinstellungen bezüglich der Suche im Quellcode. So kann u. a. nach Dateinamen, Klassennamen oder einfachem Vorkommen im Code gesucht werden.

<https://www.jetbrains.com/help/idea/finding-and-replacing-text-in-file.html>

<https://www.codetrails.com/blog/shortcuts-searching-eclipse/>

<http://wiki.netbeans.org/GermanUsersguide>

<https://www.microsoft.com/de-de/techwiese/know-how/visual-studio-code-06-produktiv-ver-durch-den-richtigen-einsatz-der-suchfunktion.aspx>

Umbenennen (Refactoring) von Klassen, Variablen, Imports etc.

Möchte man im Code eine Klasse, Variable, Datei etc. per Hand umbenennen oder verschieben, kann dies durch die vielfältige Verschachtelung im Code sehr aufwändig und fehleranfällig werden. Eine gute IDE bietet daher eine automatisierte Umbenennung an, so dass die vorgenommenen Änderungen auf alle Vorkommnisse im Code übertragen werden.

<https://www.jetbrains.com/help/idea/refactoring-source-code.html>

[https://www.wikihow.com/Rename-Variables-in-Eclipse-\(Java\)](https://www.wikihow.com/Rename-Variables-in-Eclipse-(Java))

<http://wiki.netbeans.org/Refactoring>

<https://code.visualstudio.com/docs/editor/refactoring>

