

IntelliJ portable (mit Git und JDK)

Installationsanleitung

gute Notfall-Alternative:
online-java.com/

1. IntelliJ herunterladen

- <https://www.jetbrains.com/de-de/idea/>
- herunterladen
- Community Edition
- zip-Datei (statt exe)



Herunterladen IntelliJ IDEA

Windows macOS Linux

Ultimate

Für Webentwicklung und Entwicklung in Unternehmen

Herunterladen .exe

Kostenlose 30-Tage-Testversion

Community

Für die Entwicklung mit JVM und Android

Herunterladen .exe

Kostenlos, entwickelt auf JetBrains Source

Windows (.exe)
Windows (.zip)

2. git herunterladen

- <https://git-scm.com/>
- downloads
- Windows
- vorgeschlagenen Download abbrechen, stattdessen
- 64-bit Git for Windows Portable



Downloads

macOS Windows
Linux/Unix

Git for Windows Portable ("thumbdrive edition")
32-bit Git for Windows Portable.
64-bit Git for Windows Portable.

3. JDK herunterladen

- <https://bell-sw.com/pages/downloads/>
- Download now
- JDK 11 LTS
- Windows - Full JDK
- Download ZIP

ALL
VERSIONS

JDK 8
LTS

JDK 11
LTS

JDK 17
LTS / CURRENT

Windows

☒ x86 ☐ ARM

Package ? Full JDK

Liberica Full JDK 11.0.12+7 x86 64 bit for Windows

Download MSI, 258.30Mb Checksum: SHA1

Download ZIP, 262.48Mb Checksum: SHA1

4. extrahieren

- IntelliJ-Zip
 - auf Serververzeichnis oder Stick
 - z.B. U:\Eigene Dateien\Progs\IntelliJ
- JDK
 - in IntelliJ-Verzeichnis hinein
 - z.B. U:\Eigene Dateien\Progs\IntelliJ
 - erstellten Ordner „jdk...“ in „jdk“ umbenennen
- Git
 - in IntelliJ-Unter-Verzeichnis hinein
 - z.B. U:\Eigene Dateien\Progs\IntelliJ\git

5. properties-Datei von IntelliJ anpassen

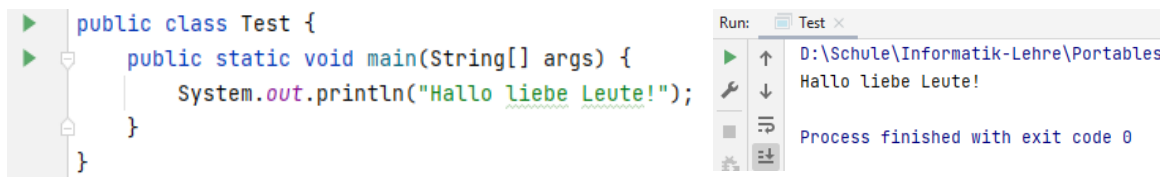
- in bin-Verzeichnis von IntelliJ - idea.properties öffnen mit Editor
- alle Pfadangaben „...path=\${...}“ durch „...path=\${idea.home.path}“ ersetzen, Hashtags in der entsprechenden Zeile entfernen und alle Pfade in Unterverzeichnis von .IdealC packen (wenn nicht schon geschehen), z.B. wird aus
„# idea.plugins.path=\${idea.config.path}/plugins“ folgendes
„idea.plugins.path=\${idea.home.path}/.IdealC/plugins“

6. IntelliJ anpassen

- im entsprechenden bin-Verzeichnis „idea64.exe“ Doppelklick
- „User Agreement“ zustimmen
- keine Konfigurationsdateien importieren
- „Customize - All settings...“
- „Version Control - Git“
- „Path to Git executable“ anpassen - auf den in 4. ausgewählten Pfad bzw. die davon ausgehende Weiterführung des Pfades nach „bin/git.exe“
- Apply - OK

7. Erstes Projekt in IntelliJ

- „Projects - New Project“ oder „Get from VCS“
- einmal beispielhaft „New Project“
- „Java“ belassen, „Project SDK“ suchen im unter 4. angegebenen Pfad zur JDK
- „Next“ - „Next“
- Projektname und Speicherort (Server oder Stick) auswählen - „Finish“
- offene Hinweisfenster schließen
- Rechtsklick auf „src“-Ordner des Projekts - „New“ - „Java Class“ - Namen vergeben
- Programm kann beginnen ...
 - „main“ bewirkt z.B. das schnelle Erzeugen der Hauptmethode:
 - `public static void main(String[] args) {`
 - `}`
 - „sout“ ist die Kurzform für ein Ausgabestatement - `System.out.println();`
 - Füllen mit Text: `System.out.println(„Hallo liebe Leute!“);`
 - Ausführen geht über Play-Button am linken Rand - „Run ...“
 - Ausgabe im unteren Bereich in der Konsole (Reiter „Run“) zu sehen



8. weitere nützliche Einstellungen

- „File - Settings“ - „Appearance & Behaviour“ - „System Settings“ - „Confirm before exiting the IDE“ und „Reopen projects on startup“ abwählen - „Apply“
- „Editor - General - Code Folding“ - „File header“, „imports“, „One-line ...“ und „Generic constructor ...“ abwählen - „Apply“
- „Editor - Natural Languages“ - Deutsch hinzufügen über „+“
- „Plugins - Installed“ - ausschalten: „Android, Build Tools, Code Coverage, HTML and XML, IDE Settings, Groovy, Plugin Development, Test Tools, Touch Integration, Themes, Tools Integration, Mercurial, Perforce ..., Subversion, Code With me, Eclipse ..., Jet Brains Marketplace ..., Kotlin, Space“ - „Apply“