

OOP Inštalácia prostredí JAVA

Treba navštíviť nasledovný odkaz, a nainštalovať **JAVA 11**, podľa verzie operačného systému (v Návode je ukázaný Windows) -

<https://jdk.java.net/java-se-ri/11>

Videonávod - https://www.youtube.com/watch?v=49fsMujWP_A (treba ale myslieť na to, že chceme nainštalovať verziu 11 a nie 14, ako inštaluje prednášajúci)

jdk.java.net

GA Releases
JDK 17
JMC 8

Early-Access
Releases

JDK 19
JDK 18
Loom

Metropolis
Panama
Valhalla

Reference
Implementations

Java SE 17
Java SE 16
Java SE 15
Java SE 14
Java SE 13
Java SE 12
Java SE 11
Java SE 10
Java SE 9
Java SE 8
Java SE 7

Feedback
Report a bug
Archive

Java Platform, Standard Edition 11 Reference Implementations

The official Reference Implementation for Java SE 11 (JSR 384) is based solely upon open-source code available from the JDK 11 Project in the OpenJDK Community. This Reference Implementation applies to both the Final Release of JSR 384 (Sep 2018) and Maintenance Release 1 (Mar 2019).

The binaries are available under the GNU General Public License version 2, with the Classpath Exception.

These binaries are for reference use only!

These binaries are provided for use by implementers of the Java SE 11 Platform Specification and are for reference purposes only. This Reference Implementation has been approved through the Java Community Process. Production-ready binaries under the GPL are available from Oracle; and will be in most popular Linux distributions.

RI Binary (build 11+28) under the GNU General Public License version 2

- Linux/x64 Java Development Kit (sha256) 178.9 MB
- Windows/x64 Java Development Kit (sha256) 178.7 MB

RI Source Code

The source code of the RI binaries is available under the GPLv2 in a single zip file (sha256) 178.1 MB.

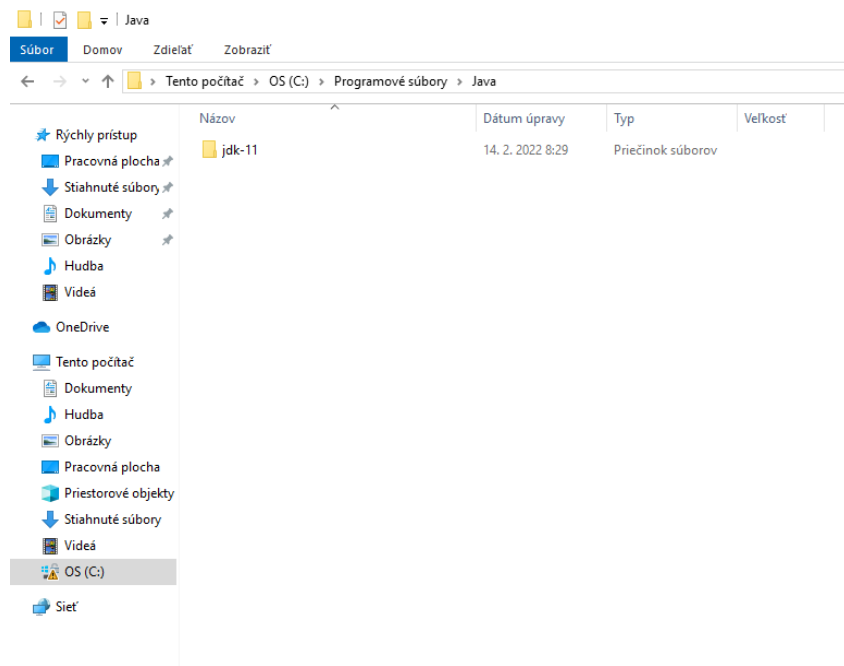
International use restrictions

Due to limited intellectual property protection and enforcement in certain countries, the JDK source code may only be distributed to an authorized list of countries. You will not be able to access the source code if you are downloading from a country that is not on this list. We are continuously reviewing this list for addition of other countries.

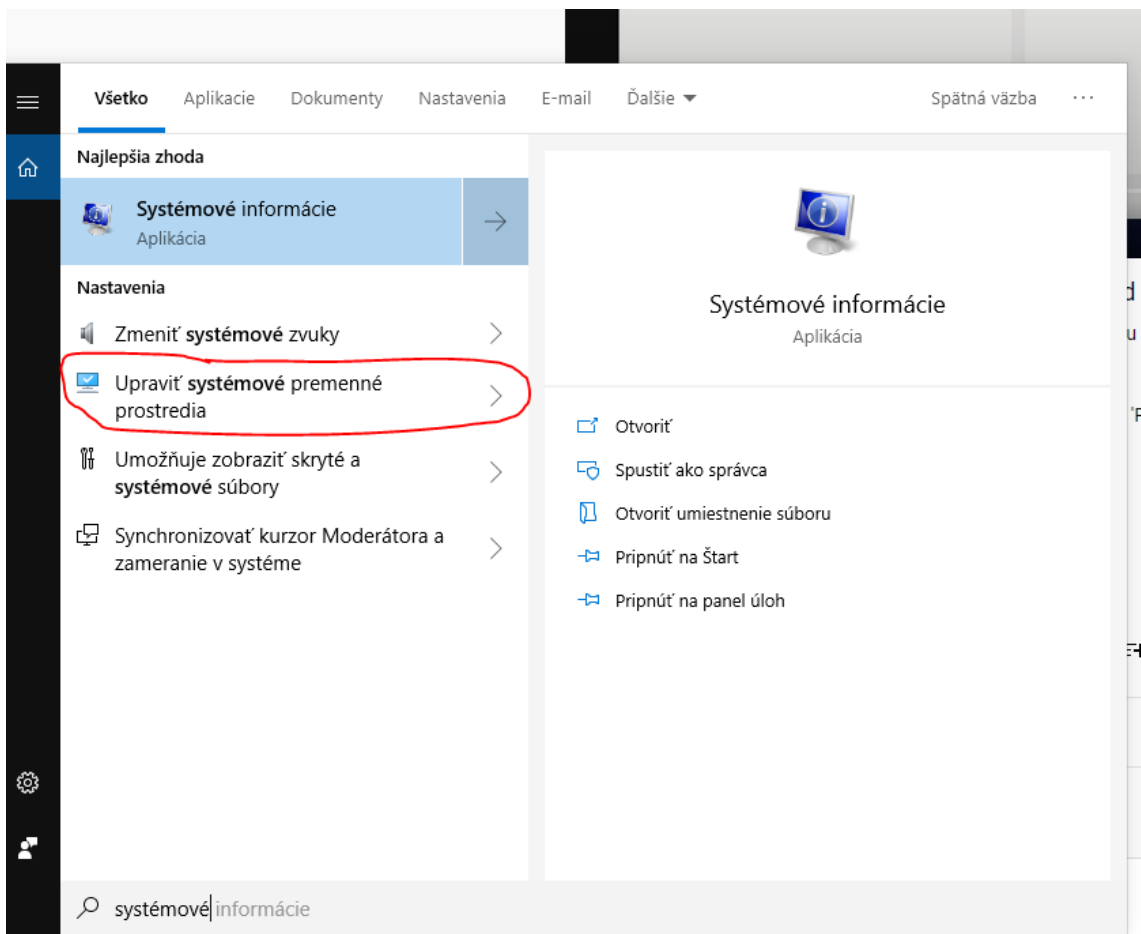


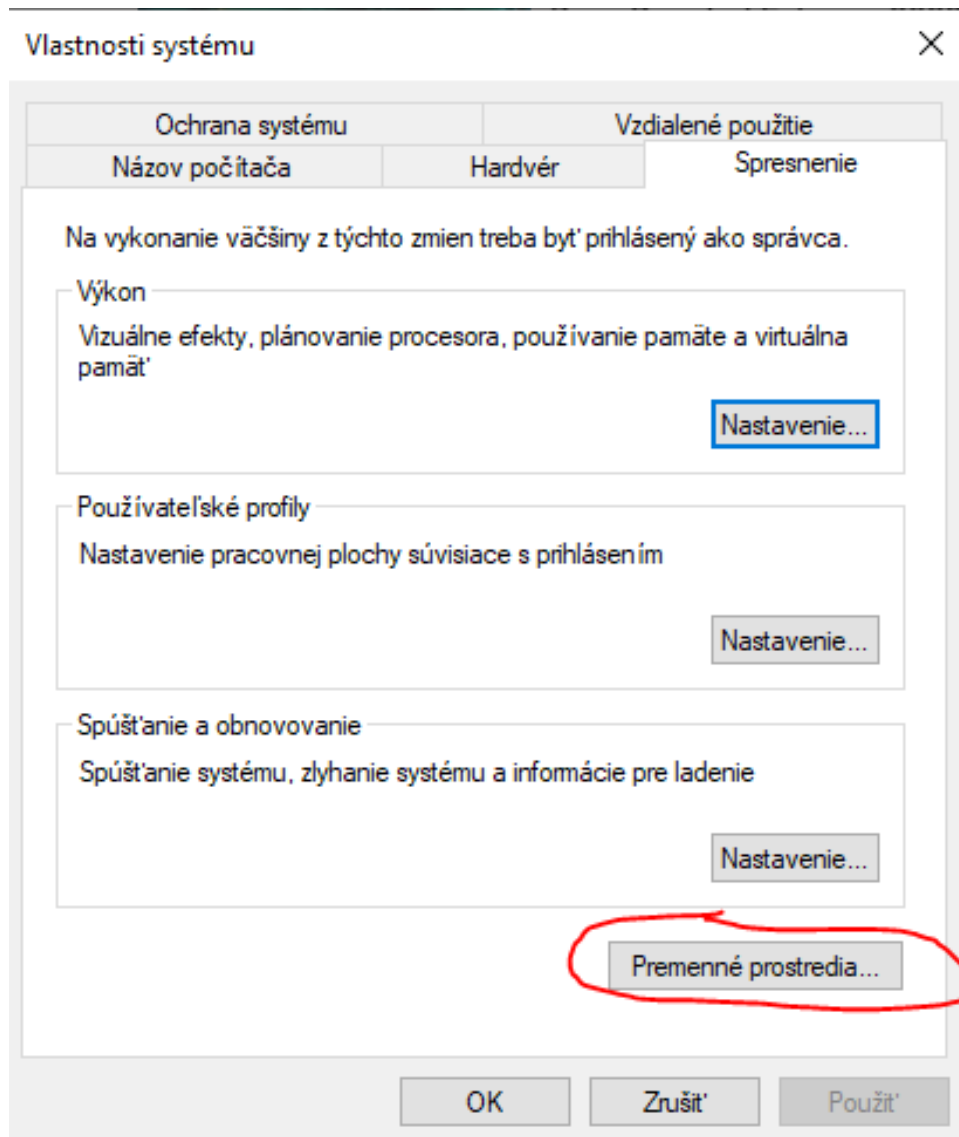
© 2022 Oracle Corporation and/or its affiliates
Terms of Use · Privacy · Trademarks

Windows: Po stiahnutí treba zip rozbaľiť, a priečinok **jdk-11** nakopírovať do cesty **C:/Program Files/Java** (ak tam priečinok Java nemáte, vytvorte ho)

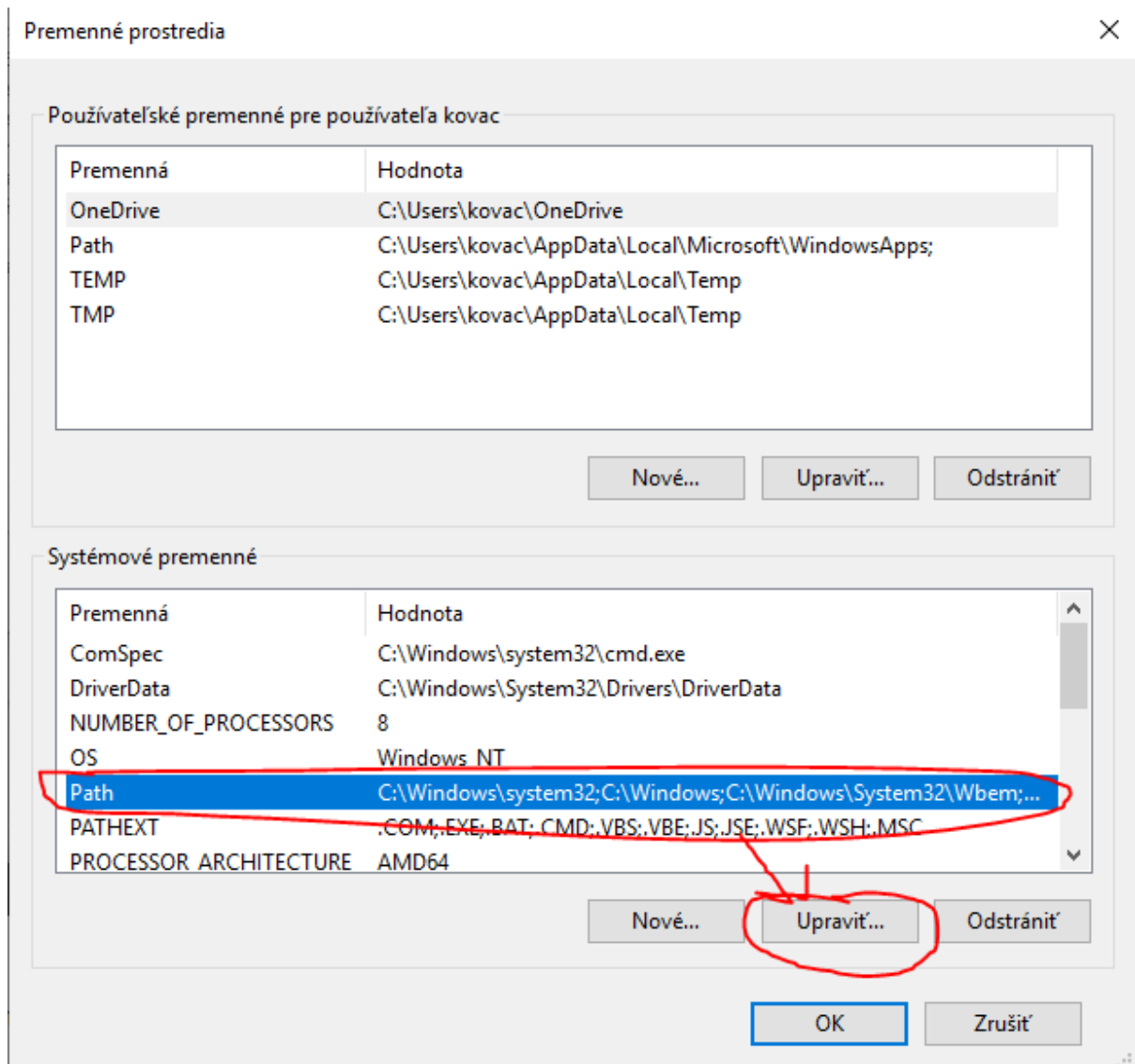


Po nakopírovaní priečinku jdk-11 potrebujeme povedať Windowsu, kde má Javu hľadať, to spravíme pomocou systémových premenných (Environment variables), vo vyhľadávaní otvoríme **Upraviť systémové premenné prostredia** a v otvorenom okne klikneme na **Premenné protredia...**

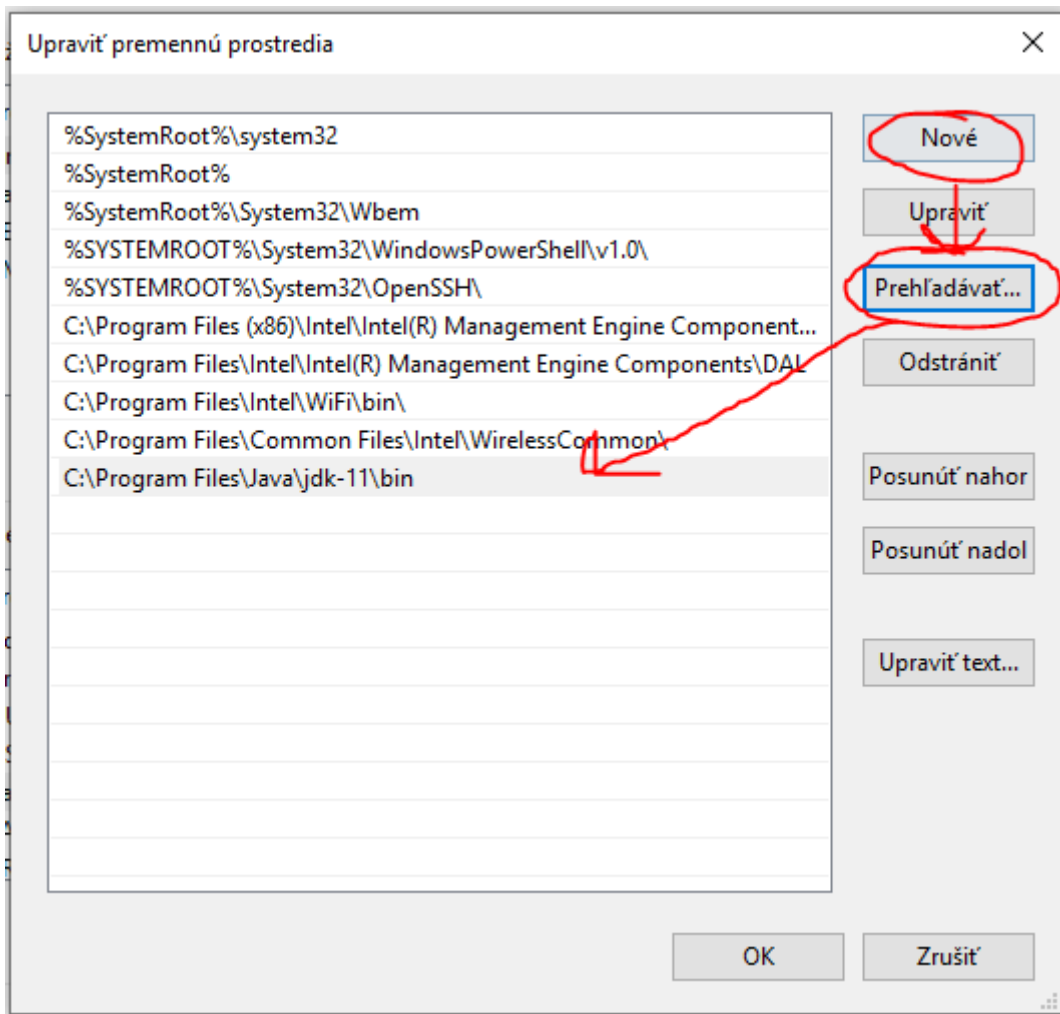




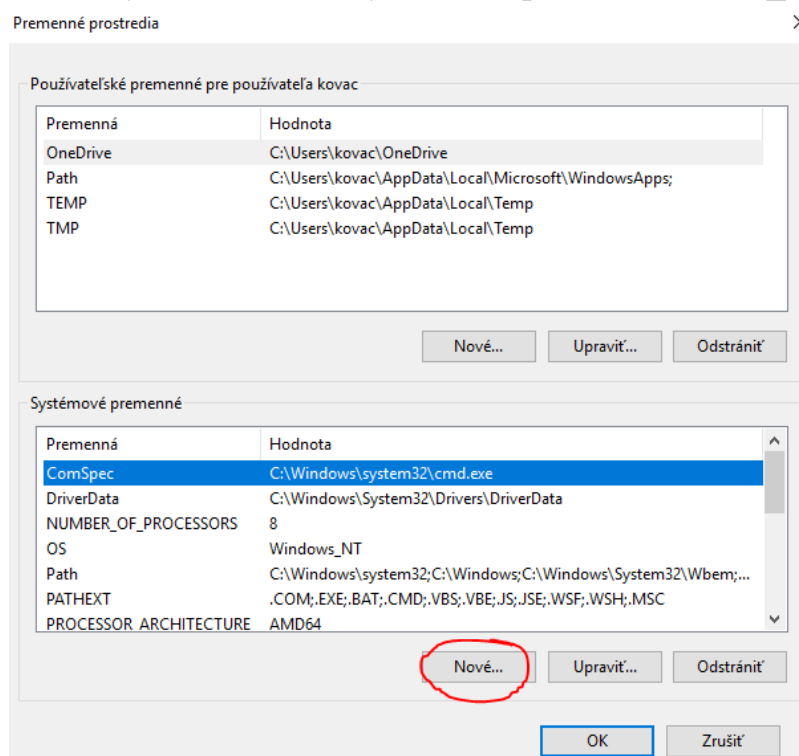
Zvolíme položku **Path** a klikneme **upraviť**

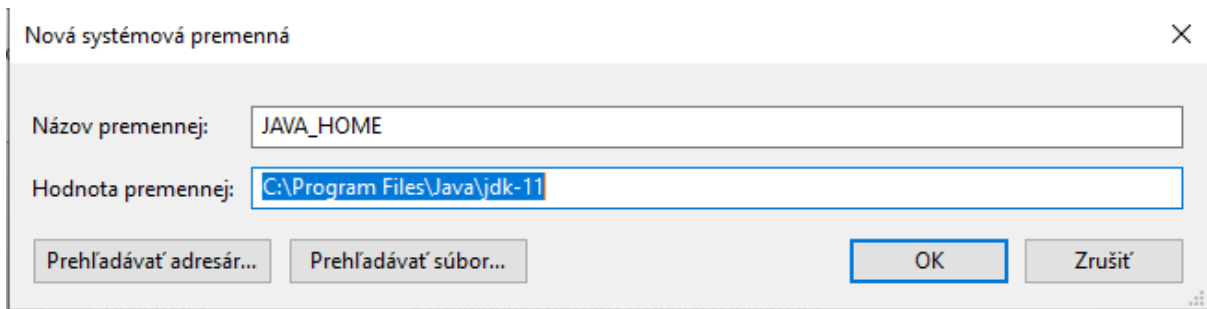


Následne sa pozrieme, či už pre danú verziu javy nemáme systémovú cestu, ak nie, tak klikneme na nový a následne na prehľadávať, a vyberieme cestu kde sme skopírovali priečinok jdk-11 a k danej ceste klikneme ešte podpriečinok bin (cestu vidno na obrázku)



Následne si vytvoríme novú systémovú premennú JAVA_HOME





Tejto premennej nastavíme cestu k priečinku **jdk-11**, následne potvrdím všetky okná, a mali by sme mať nainštalovanú Javu.

Kontrola: Pre kontrolu, či máme javu správne nainštalovanú, treba otvoriť príkazový riadok, a po zadaní príkazu **java** alebo **javac** by sme mali vidieť help k danému príkazu

```
Microsoft Windows [Version 10.0.17763.379]
(c) 2018 Microsoft Corporation. Všetky práva vyhradené.

C:\Users\kovac>java
Usage: java [options] <mainclass> [args...]
        (to execute a class)
or java [options] -jar <jarfile> [args...]
        (to execute a jar file)
or java [options] -m <module>[/<mainclass>] [args...]
   java [options] --module <module>[/<mainclass>] [args...]
        (to execute the main class in a module)
or java [options] <sourcefile> [args]
        (to execute a single source-file program)

Arguments following the main class, source file, -jar <jarfile>,
-m or --module <module>/<mainclass> are passed as the arguments to
main class.

where options include:

  -cp <class search path of directories and zip/jar files>
  -classpath <class search path of directories and zip/jar files>
  --class-path <class search path of directories and zip/jar files>
               A ; separated list of directories, JAR archives,
               and ZIP archives to search for class files.
  -p <module path>
  --module-path <module path>...
               A ; separated list of directories, each directory
               is a directory of modules.
  --upgrade-module-path <module path>...
```

Vo videonávode je ukázaný aj malý príklad, z kompilovaním java súborov cez príkazový riadok. https://www.youtube.com/watch?v=49fsMujWP_A

GIT

Otvoríme stránku <https://git-scm.com/downloads>, a vyberieme si operačný systém (my si ukážeme Windows)

git --local-branching-on-the-cheap

Search entire site...

About

Documentation

Downloads

GUI Clients

Logos

Community

The entire **Pro Git book** written by Scott Chacon and Ben Straub is available to [read online for free](#). Dead tree versions are available on [Amazon.com](#).

Downloads

macOS **Windows** Linux/Unix

Older releases are available and the [Git source repository](#) is on GitHub.

GUI Clients

Git comes with built-in GUI tools (**git-gui**, **gitk**), but there are several third-party tools for users looking for a platform-specific experience.

[View GUI Clients →](#)

Logos

Various Git logos in PNG (bitmap) and EPS (vector) formats are available for use in online and print projects.

[View Logos →](#)

Git via Git

If you already have Git installed, you can get the latest development version via Git itself:

```
git clone https://github.com/git/git
```

You can also always browse the current contents of the git repository using the [web interface](#).

</> About this site
Patches, suggestions, and comments are welcome.

Git is a member of Software Freedom Conservancy

A klikneme na najnovšiu verziu.

Download for Windows

[Click here to download](#) the latest (2.35.1) 64-bit version of **Git for Windows**. This is the most recent [maintained build](#). It was released **13 days ago**, on 2022-02-01.

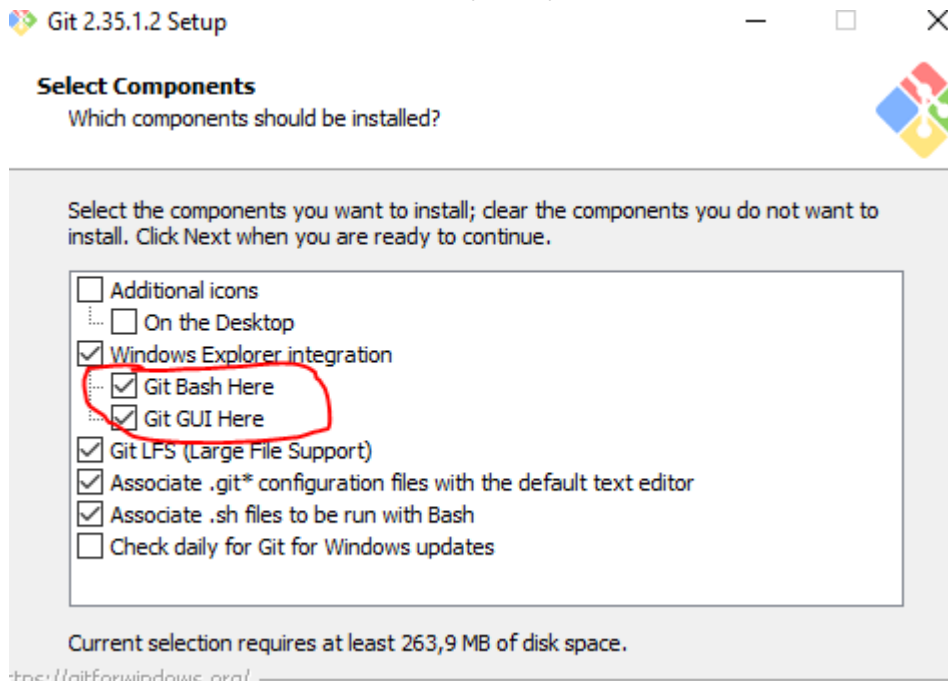
Other Git for Windows downloads

Standalone Installer

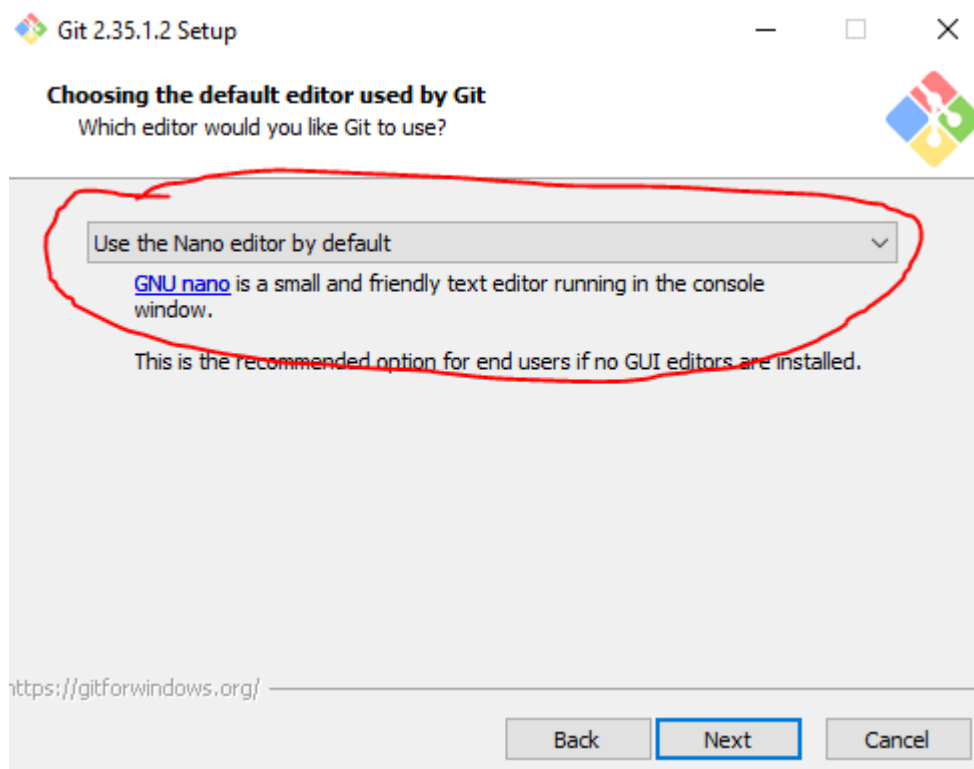
[32-bit Git for Windows Setup.](#)

[64-bit Git for Windows Setup.](#)

Pri inštalácii určite treba mať vybratý Git Bash



Pri výbere defaultného editora to nechám na vás, ja som si osobne vybral Nano



Tak isto nechávam za seba rozhodovať git, že sa hlavná vetva (branch) bude volať master (ak by s tým mal niekto problém, kľudne si zvolte druhú možnosť a premenujte si ju)

Adjusting the name of the initial branch in new repositories

What would you like Git to name the initial branch after "git init"?

 **Let Git decide**

Let Git use its default branch name (currently: "master") for the initial branch in newly created repositories. The Git project [intends](#) to change this default to a more inclusive name in the near future.

 Override the default branch name for new repositories

NEW! Many teams already renamed their default branches; common choices are "main", "trunk" and "development". Specify the name "git init" should use for the initial branch:

This setting does not affect existing repositories.

<https://gitforwindows.org/>

Back

Next

Cancel

Ďalšie nastavenie nechávam na **Recommended****Adjusting your PATH environment**

How would you like to use Git from the command line?

 **Use Git from Git Bash only**

This is the most cautious choice as your PATH will not be modified at all. You will only be able to use the Git command line tools from Git Bash.

 Git from the command line and also from 3rd-party software

(Recommended) This option adds only some minimal Git wrappers to your PATH to avoid cluttering your environment with optional Unix tools. You will be able to use Git from Git Bash, the Command Prompt and the Windows PowerShell as well as any third-party software looking for Git in PATH.

 Use Git and optional Unix tools from the Command Prompt

Both Git and the optional Unix tools will be added to your PATH.
Warning: This will override Windows tools like "find" and "sort". Only use this option if you understand the implications.

<https://gitforwindows.org/>

Back

Next

Cancel

SSH a HTTPS nastavenia tiež nechávam na predvolenom

Choosing the SSH executable

Which Secure Shell client program would you like Git to use?



Use bundled OpenSSH

This uses ssh.exe that comes with Git.

Use external OpenSSH

NEW! This uses an external ssh.exe. Git will not install its own OpenSSH (and related) binaries but use them as found on the PATH.

<https://gitforwindows.org/>

Back Next Cancel

Choosing HTTPS transport backend

Which SSL/TLS library would you like Git to use for HTTPS connections?



Use the OpenSSL library

Server certificates will be validated using the ca-bundle.crt file.

Use the native Windows Secure Channel library

Server certificates will be validated using Windows Certificate Stores. This option also allows you to use your company's internal Root CA certificates distributed e.g. via Active Directory Domain Services.

<https://gitforwindows.org/>

Back Next Cancel

Configuring the line ending conversions

How should Git treat line endings in text files?

 **Checkout Windows-style, commit Unix-style line endings**

Git will convert LF to CRLF when checking out text files. When committing text files, CRLF will be converted to LF. For cross-platform projects, this is the recommended setting on Windows ("core.autocrlf" is set to "true").

 Checkout as-is, commit Unix-style line endings

Git will not perform any conversion when checking out text files. When committing text files, CRLF will be converted to LF. For cross-platform projects, this is the recommended setting on Unix ("core.autocrlf" is set to "input").

 Checkout as-is, commit as-is

Git will not perform any conversions when checking out or committing text files. Choosing this option is not recommended for cross-platform projects ("core.autocrlf" is set to "false").

<https://gitforwindows.org/>

Back

Next

Cancel

Configuring the terminal emulator to use with Git Bash

Which terminal emulator do you want to use with your Git Bash?

 **Use MinTTY (the default terminal of MSYS2)**

Git Bash will use MinTTY as terminal emulator, which sports a resizable window, non-rectangular selections and a Unicode font. Windows console programs (such as interactive Python) must be launched via `winpty` to work in MinTTY.

 Use Windows' default console window

Git will use the default console window of Windows ("cmd.exe"), which works well with Win32 console programs such as interactive Python or node.js, but has a very limited default scroll-back, needs to be configured to use a Unicode font in order to display non-ASCII characters correctly, and prior to Windows 10 its window was not freely resizable and it only allowed rectangular text selections.

<https://gitforwindows.org/>

Back

Next

Cancel

Choose the default behavior of `git pull`

What should `git pull` do by default?



Default (fast-forward or merge)

This is the standard behavior of `git pull`: fast-forward the current branch to the fetched branch when possible, otherwise create a merge commit.

Rebase

Rebase the current branch onto the fetched branch. If there are no local commits to rebase, this is equivalent to a fast-forward.

Only ever fast-forward

Fast-forward to the fetched branch. Fail if that is not possible.

<https://gitforwindows.org/>

Back

Next

Cancel

Choose a credential helper

Which credential helper should be configured?



Git Credential Manager

Use the [cross-platform Git Credential Manager](#).
See more information about the future of Git Credential Manager [here](#).

None

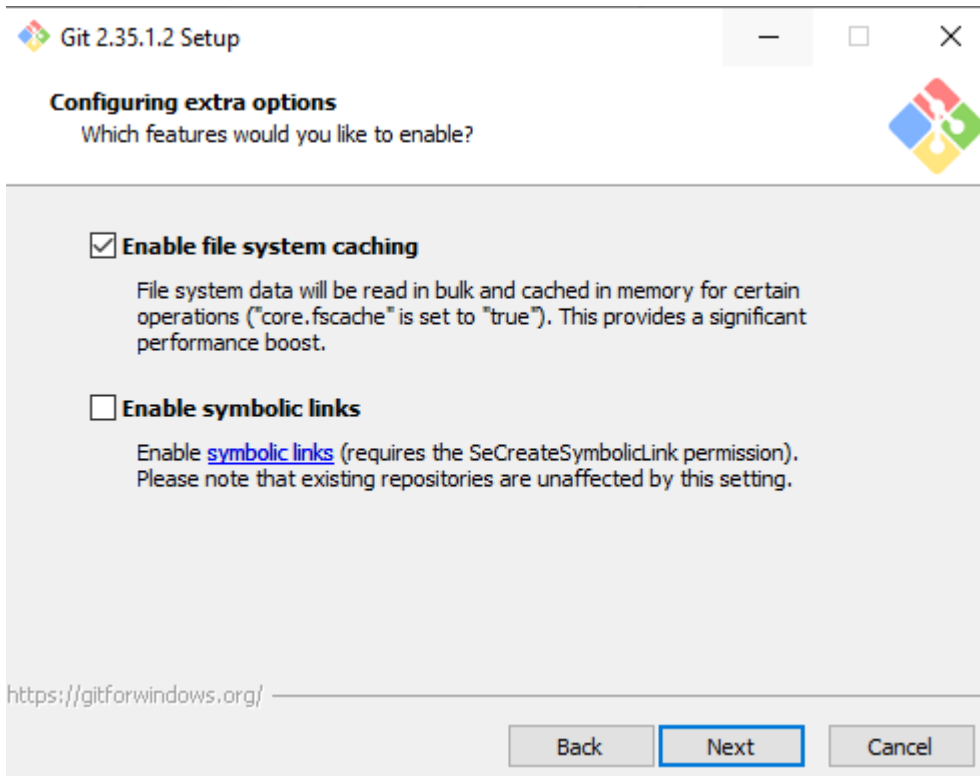
Do not use a credential helper.

<https://gitforwindows.org/>

Back

Next

Cancel



nasleduje ešte jedna strana kde necháme všetko tak ako nám to predvolene povoľuje a inštalácia prebehne sama.

Následne na zoznámenie z Gitom je dobré si prečítať nasledujúce stránky About - <https://docs.github.com/en/get-started/using-git/about-git>

Na tejto stránke si treba hlavne pozrieť **Basic Git Commands**

Gitignore - <https://git-scm.com/docs/gitignore>

súbor nachádzajúci sa v priečinkoch repozitára, stránka vysvetľuje jeho obsah

Najlepšie je si nastaviť autorizáciu vzdialených repozitárov pomocou SSH kľúčov, návody sú na nasledujúcich linkoch (pre nastavenie SSH kľúča a a pracovanie so vzdialenými repozitármi treba mať vytvorený účet na platforme GitHub, preto by sme vás poprosili aby ste si účet vytvorili pomocou stuba mailov, a aby ste si nastavili profil s vaším menom a priezviskom, alebo s ais menom, aby sme vás neskôr pri zadaniach vedeli identifikovať)

<https://docs.github.com/en/authentication/connecting-to-github-with-ssh/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent>

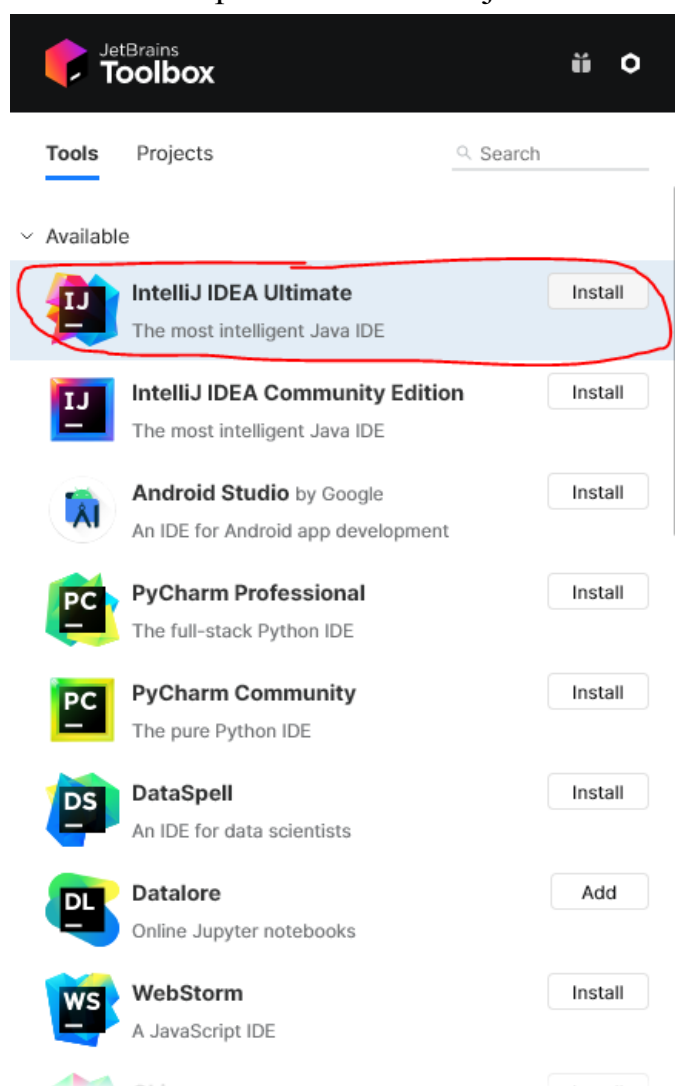
<https://docs.github.com/en/authentication/connecting-to-github-with-ssh/adding-a-new-ssh-key-to-your-github-account>

IDE

Odporúčame používať IntelliJ IDEA ako IDE pre tento predmet, avšak môžete použiť IDE aké len chcete (iné IDE nájdete na webovej stránke predmetu) Najprv sa treba registrovať ako študent, aby ste mali zadarmo licenciu k JetBrains produktom - <https://www.jetbrains.com/shop/eform/students>

Po úspešnej registrácii následne odporúčam stiahnuť toolbox - <https://www.jetbrains.com/toolbox-app/>

Po nainštalovaní Toolboxu ho spustíte a nainštalujete **IntelliJ IDEA Ultimate**



Po nainštalovaní je možné že budete musieť spustiť IntelliJ IDEU cez menu vo windowse (aspon mne ju nešlo na klik v toolboxe spustit). Po spustení od vás bude chcieť importovať nastavenia, pokiaľ ste IntelliJ IDEU ešte nepoužívali, kľudne dajte možnosť **Do not import**. Následne sa vám zobrazí okno, cez ktoré budete musieť potvrdiť svoju identitu. Najjednoduchšie je to cez prihlásenie sa, po prihlásení treba kliknúť na **Activate** a môžete okno zavrieť cez tlačidlo **Close**.