



Python Basic

Заняття 3: GIT.

Структура заняття

- Що таке git?
- Навіщо він потрібен ?
- Які можливості надає git?
- Як встановити git?
- Що таке репозиторій?
- Як створити репозиторій?
- Що таке коміт і комітити?
- Які є стани у репозиторію?
- Що таке github?
- Що краще gui чи cli?
- Навіщо потрібні файли readme.md і .gitignore?

git це

- Абсолютний лідер за популярністю серед сучасних систем управління версіями.
- Розподілена система керування версіями.
- Створено Лінусом Торвальдсом.
- Відкритий вихідний код.
- Безкоштовний.


git дуже корисний якщо...

- виявляється, що завдання, що вирішується, більше не актуально;
- потрібно внести виправлення до більш ранньої версії програми;
- помилка знайшлася під час роботи над новим завданням;
- повернути будь-які зміни.

Можливості git

- повернення до будь-якої попередньої версії коду;
- перегляд історії змін;
- паралельна робота над проектом;
- бекуп коду.


Інсталювання


 **git** --distributed-is-the-new-centralized


Search entire site...

About
Documentation
Downloads
GUI Clients
Logos
Community

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

Logos

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



Latest source Release
2.37.1
[Release Notes](#) (2022-07-04)
[Download for Windows](#)

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](#).



git

--distributed-is-the-new-centralized



Search entire site...

About

Documentation

Downloads

GUI Clients

Logos

Community

Download for Windows

[Click here to download](#) the latest **(2.37.1) 64-bit** version of **Git for Windows**. This is the most recent [maintained build](#). It was released **5 days ago**, on 2022-07-12.

Other Git for Windows downloads

Standalone Installer



Do you want to allow this app to make changes to your device?



Git for Windows

Verified publisher: Johannes Schindelin

File origin: Network drive

[Show more details](#)

Yes

No



Git 2.37.1 Setup



Information

Please read the following important information before continuing.



When you are ready to continue with Setup, click Next.

GNU General Public License

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place - Suite 330, Boston, MA 02111-1307, USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

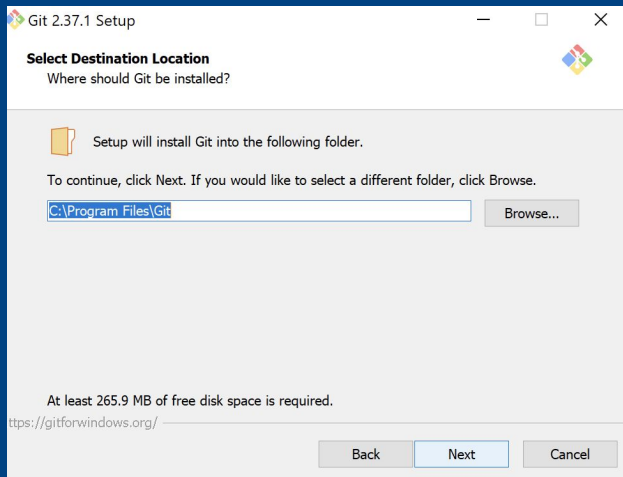
Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--

<https://gitforwindows.org/>

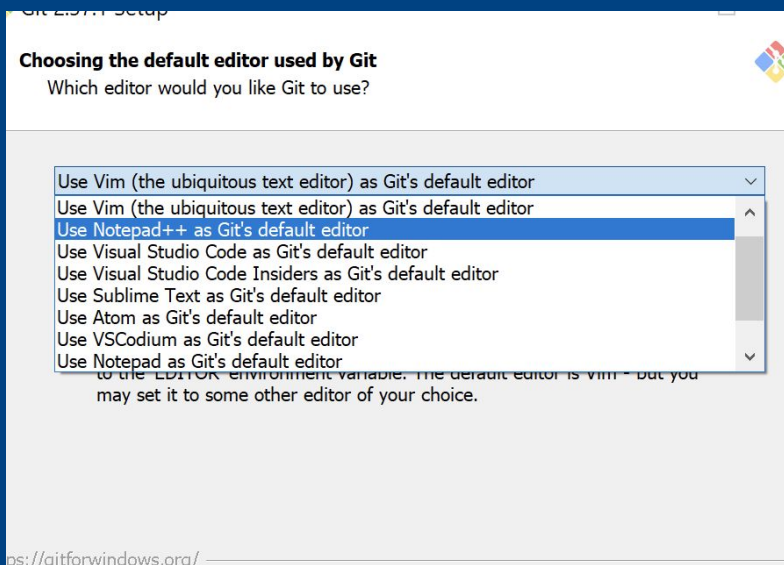
Next

Cancel



Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue.

- ☒ Additional icons
 - ☒ On the Desktop
- ☒ Windows Explorer integration
 - ☒ Git Bash Here
 - ☒ Git GUI Here
- ☒ Git LFS (Large File Support)
- ☒ Associate .git* configuration files with the default text editor
- ☒ Associate .sh files to be run with Bash
- ☐ Check daily for Git for Windows updates
- ☒ (NEW!) Add a Git Bash Profile to Windows Terminal



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.

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/>

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").

Що таке репозиторій?

Це місце, де зберігається ваш код та вся інформація про його зміни. Репозиторії можуть бути у вас на комп'ютері, на комп'ютерах ваших колег і на віддаленому сервері.

Як отримати репозиторій

- `git init` - створити локальний репозиторій
- `git clone <url>` - клонувати існуючий репозиторій
-

Що таке коміт і комітити?

Англійською `commit` означає «фіксувати». Git-коміт - це операція, яка бере всі підготовлені зміни та відправляє їх до репозиторію як єдине ціле.

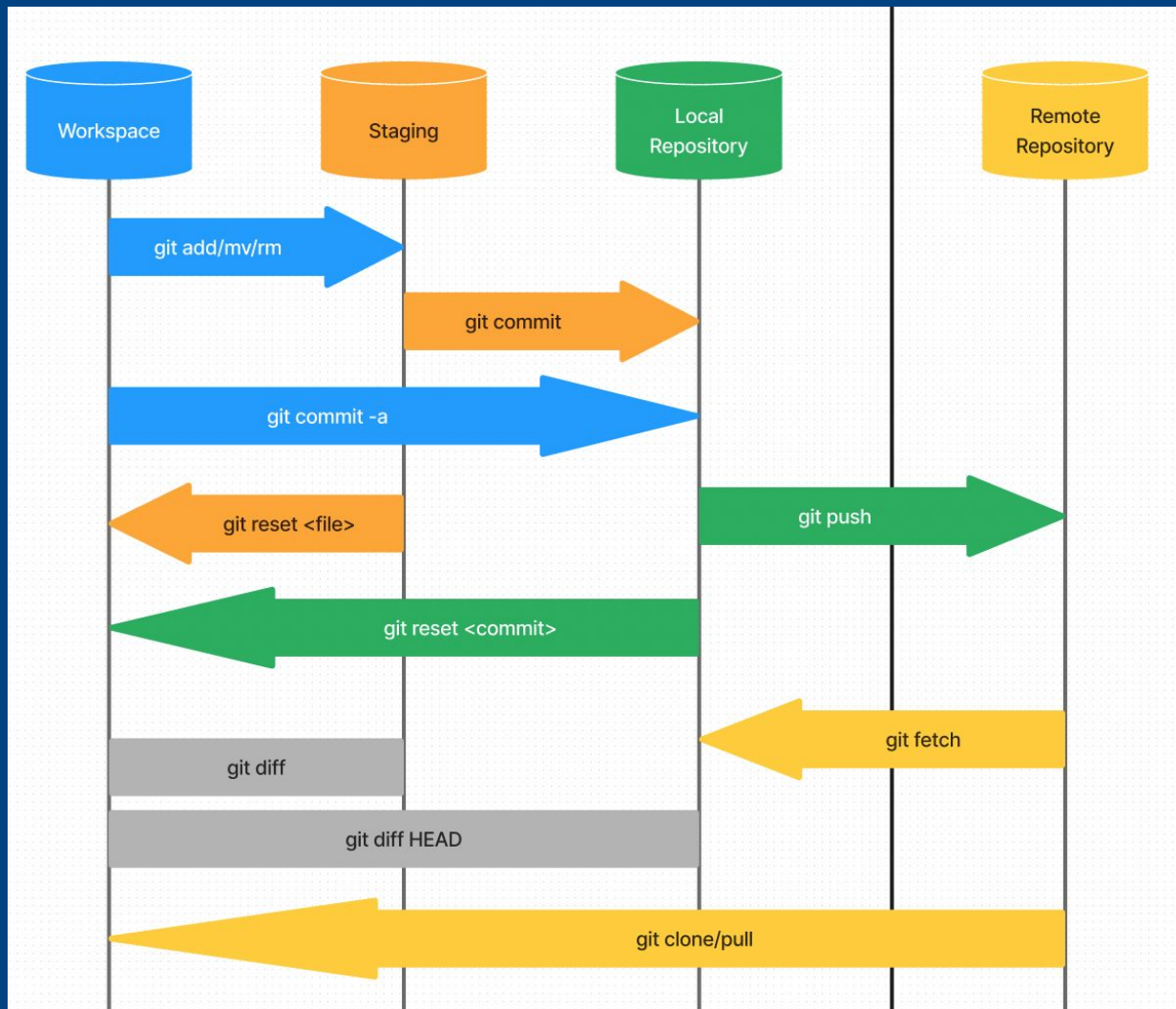
Як правило, робочий процес є циклом: коміт — зміна файлів — коміт.

| | COMMENT | DATE |
|---|------------------------------------|--------------|
| ○ | CREATED MAIN LOOP & TIMING CONTROL | 14 HOURS AGO |
| ○ | ENABLED CONFIG FILE PARSING | 9 HOURS AGO |
| ○ | MISC BUGFIXES | 5 HOURS AGO |
| ○ | CODE ADDITIONS/EDITS | 4 HOURS AGO |
| ○ | MORE CODE | 4 HOURS AGO |
| ○ | HERE HAVE CODE | 4 HOURS AGO |
| ○ | AAAAAAAAA | 3 HOURS AGO |
| ○ | ADKFJSLKDFJSDKLFJ | 3 HOURS AGO |
| ○ | MY HANDS ARE TYPING WORDS | 2 HOURS AGO |
| ○ | HAAAAAAAAAANDS | 2 HOURS AGO |

AS A PROJECT DRAGS ON, MY GIT COMMIT
MESSAGES GET LESS AND LESS INFORMATIVE.

Стани репозиторію

- Змінено (modified)
- Індексований (staged)
- Збережено до комміту (committed)



GitHub

Існують різні способи зберігання та використання репозиторію: виділяють локальні, централізовані та розподілені системи контролю версій.

У локальних системах контролю версій репозиторій зберігається та використовується на одному пристрої, але працювати з такою системою може лише один розробник. У разі централізованої системи репозиторій зберігається одному сервері.

GUI vs CLI

- GUI - Graphical User Interface
- CLI - Command Line Interface

README.md & .gitignore