



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

<https://gitforwindows.org/>

Next

Cancel



Select Components

Which components should be installed?



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
 - Open Git Bash here
 - Open 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

Current selection requires at least 340.3 MB of disk space.

<https://gitforwindows.org/>

Back

Next

Cancel



Select Start Menu Folder

Where should Setup place the program's shortcuts?



- —
- — Setup will create the program's shortcuts in the following Start Menu folder.
- —

To continue, click Next. If you would like to select a different folder, click Browse.

 Git[Browse...](#)

Don't create a Start Menu folder

<https://gitforwindows.org/>

[Back](#)[Next](#)[Cancel](#)



Choosing the default editor used by Git

Which editor would you like Git to use?



Use Visual Studio Code as Git's default editor



Visual Studio Code is an Open Source, lightweight and powerful editor running as a desktop application. It comes with built-in support for JavaScript, TypeScript and Node.js and has a rich ecosystem of extensions for other languages (such as C++, C#, Java, Python, PHP, Go) and runtimes (such as .NET and Unity).

(WARNING!) This will be installed only for this user.

Use this option to let Git use Visual Studio Code as its default editor.

<https://gitforwindows.org/>

Back

Next

Cancel



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.

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:

main

This setting does not affect existing repositories.

<https://gitforwindows.org/>

Back

Next

Cancel



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



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?



Fast-forward or merge

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.
This is the standard behavior of `git pull`.

<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



Configuring extra options

Which features would you like to enable?



Enable file system caching

File system data will be read in bulk and cached in memory for certain operations ("core.fscache" is set to "true"). This provides a significant performance boost.

Enable symbolic links

Enable [symbolic links](#) (requires the SeCreateSymbolicLink permission). Please note that existing repositories are unaffected by this setting.

<https://gitforwindows.org/>

Back

Install

Cancel

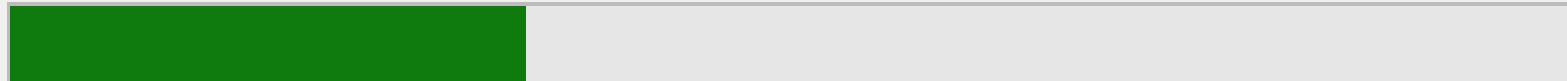
**Installing**

Please wait while Setup installs Git on your computer.



Extracting files...

C:\Program Files\Git\mingw64\libexec\p11-kit\trust-extract-compat



<https://gitforwindows.org/>

Cancel



Completing the Git Setup Wizard



Setup has finished installing Git on your computer. The application may be launched by selecting the installed shortcuts.

Click Finish to exit Setup.

Launch Git Bash

View Release Notes

Finish

Command Prompt - cmd

X + ▾

Microsoft Windows [Version 10.0.26100.5074]
(c) Microsoft Corporation. All rights reserved.

C:\Users\chadw>cmd

Microsoft Windows [Version 10.0.26100.5074]
(c) Microsoft Corporation. All rights reserved.

C:\Users\chadw>git --version
git version 2.51.0.windows.1

C:\Users\chadw>

1

General

Owner *



chadwickenebeli04

Repository name *

INFR3120ULab

INFR3120ULab is available.

Great repository names are short and memorable. How about [laughing-engine](#)?

Description

0 / 350 characters

2

Configuration

Choose visibility *

Choose who can see and commit to this repository

Public

Add README

READMEs can be used as longer descriptions. [About READMEs](#)

Off

Add .gitignore

.gitignore tells git which files not to track. [About ignoring files](#)

No .gitignore

Add license

Licenses explain how others can use your code. [About licenses](#)

No license

Create repository

Quick setup — if you've done this kind of thing before

 Set up in Desktop

or

HTTPS

SSH

<https://github.com/chadwickenebeli04/INFR3120ULab.git>



Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# INFR3120ULab" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/chadwickenebeli04/INFR3120ULab.git
git push -u origin main
```



...or push an existing repository from the command line

```
git remote add origin https://github.com/chadwickenebeli04/INFR3120ULab.git
git branch -M main
git push -u origin main
```



 **ProTip!** Use the URL for this page when adding GitHub as a remote.

```
chadw@Kenechi MINGW64 ~
$ git config --global user.email chadwick.enebeli@ontariotechu.net

chadw@Kenechi MINGW64 ~
$ git config --global user.name "chadwickenebeli04"

chadw@Kenechi MINGW64 ~
$
```

Command Prompt

X + ▾

Microsoft Windows [Version 10.0.26100.5074]
(c) Microsoft Corporation. All rights reserved.

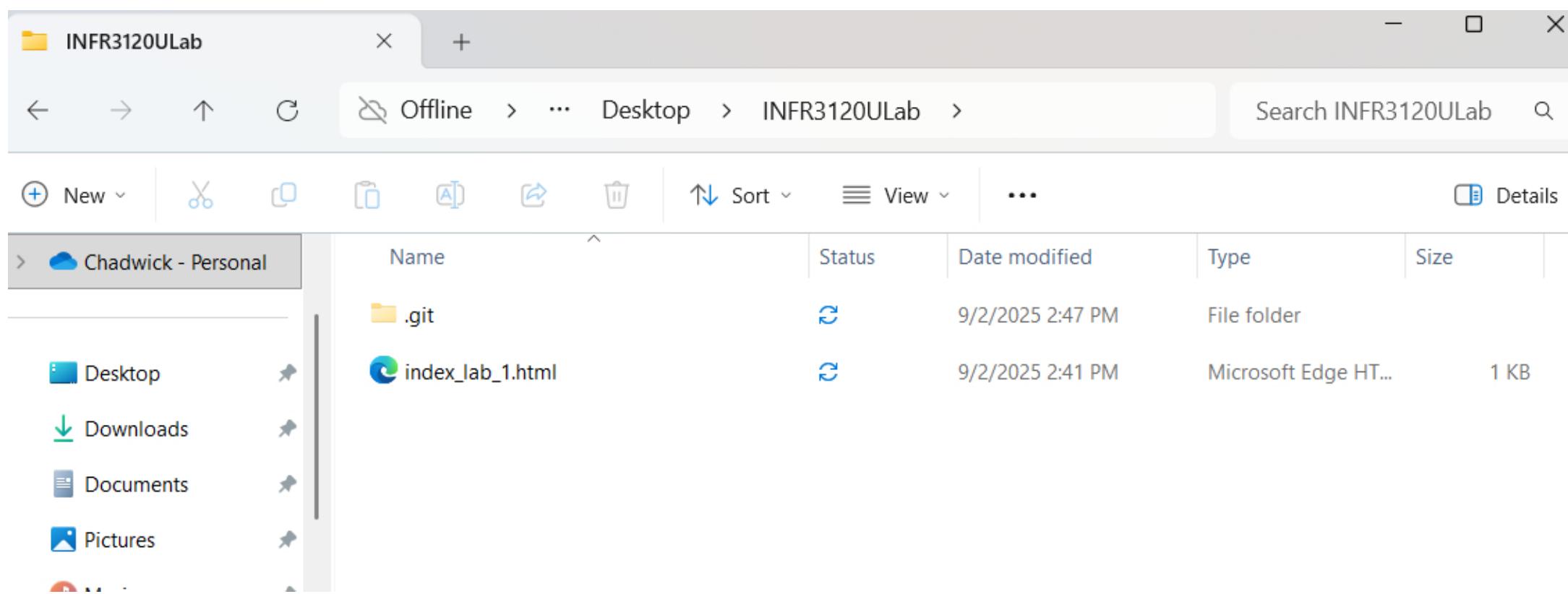
C:\Users\chadw>git --version
git version 2.51.0.windows.1

C:\Users\chadw>cd
C:\Users\chadw

C:\Users\chadw>cd
C:\Users\chadw

C:\Users\chadw>cd C:\Users\chadw\OneDrive\Desktop\INFR3120ULab
C:\Users\chadw\OneDrive\Desktop\INFR3120ULab>

```
C:\Users\chadw\OneDrive\Desktop\INFR3120ULab>git init
Initialized empty Git repository in C:/Users/chadw/OneDrive/Desktop/INFR3120ULab/.git/
C:\Users\chadw\OneDrive\Desktop\INFR3120ULab>
```



Connect to GitHub

X

GitHub

Sign in

[Browser/Device](#)

[Token](#)

[Sign in with your browser](#)

[Sign in with a code](#)

Don't have an account? [Sign up](#)

```
C:\Users\chadw\OneDrive\Desktop\INFR3120ULab>echo "# INFR3120ULab" >> README.md

C:\Users\chadw\OneDrive\Desktop\INFR3120ULab>git init
Reinitialized existing Git repository in C:/Users/chadw/OneDrive/Desktop/INFR3120ULab/.git/

C:\Users\chadw\OneDrive\Desktop\INFR3120ULab>git add README.md

C:\Users\chadw\OneDrive\Desktop\INFR3120ULab>git commit -m "first commit"
[main (root-commit) e5f1ac8] first commit
 1 file changed, 1 insertion(+)
 create mode 100644 README.md

C:\Users\chadw\OneDrive\Desktop\INFR3120ULab>git branch -M main

C:\Users\chadw\OneDrive\Desktop\INFR3120ULab>git remote add origin https://github.com/chadwickenebeli04/INFR3120ULab.git
error: remote origin already exists.

C:\Users\chadw\OneDrive\Desktop\INFR3120ULab>git push -u origin main
info: please complete authentication in your browser...
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 237 bytes | 237.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/chadwickenebeli04/INFR3120ULab.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.

C:\Users\chadw\OneDrive\Desktop\INFR3120ULab>
```

No packages published

chadwickenebeli04 / INFR3120ULab

Type [I] to search

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

INFR3120ULab Public

Pin Watch 0 Fork 0 Star 0

main 1 Branch 0 Tags

Go to file Add file Code

chadwickenebeli04 first commit e5f1ac8 · 1 minute ago 1 Commit

README.md first commit 1 minute ago

README

INFR3120ULab

About

No description, website, or topics provided.

Readme Activity 0 stars 0 watching 0 forks

Releases

No releases published Create a new release

Packages

No packages published Publish your first package