

How to use GitHub with IntelliJ

Innova Lee(이상훈)
gcccompil3r@gmail.com

github.com



icleIntegrationRepo
n3/IntegrationProject
nc3502.github.io
linux-Zybo-Z7-10
bGLEDU
aLinux-Ultra96
linux-Zybo-Z7-10

Learn Git and GitHub

Using the Hello World guide, you can create a new repository, create a new branch, write comments,

[Read the guide](#)

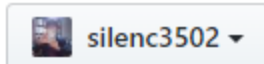


You've been added to the **koitt-er**

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner



/

Repository name *



Great repository names are short and memorable. Need inspiration? How about **effective-octo-parakeet**?

Description (optional)

KH Android



Public

Anyone can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.

Skip this step if you're importing an existing repository.



Initialize this repository with a README


This will let you immediately clone the repository to your computer.


Add .gitignore: **Java** ▼


Add a license: **MIT License** ▼



Create repository

 [silenc3502](#) / [SHL_KH_Android](#)

 **Code**

 Issues **0**

 Pull requests **0**

 Projects **0**

 Wiki

KH Android

[Manage topics](#)

 **1** commit

 **1** branch



Branch: master ▾

New pull request



silenc3502 Initial commit



[.gitignore](#)

Initial co



[LICENSE](#)

Initial co

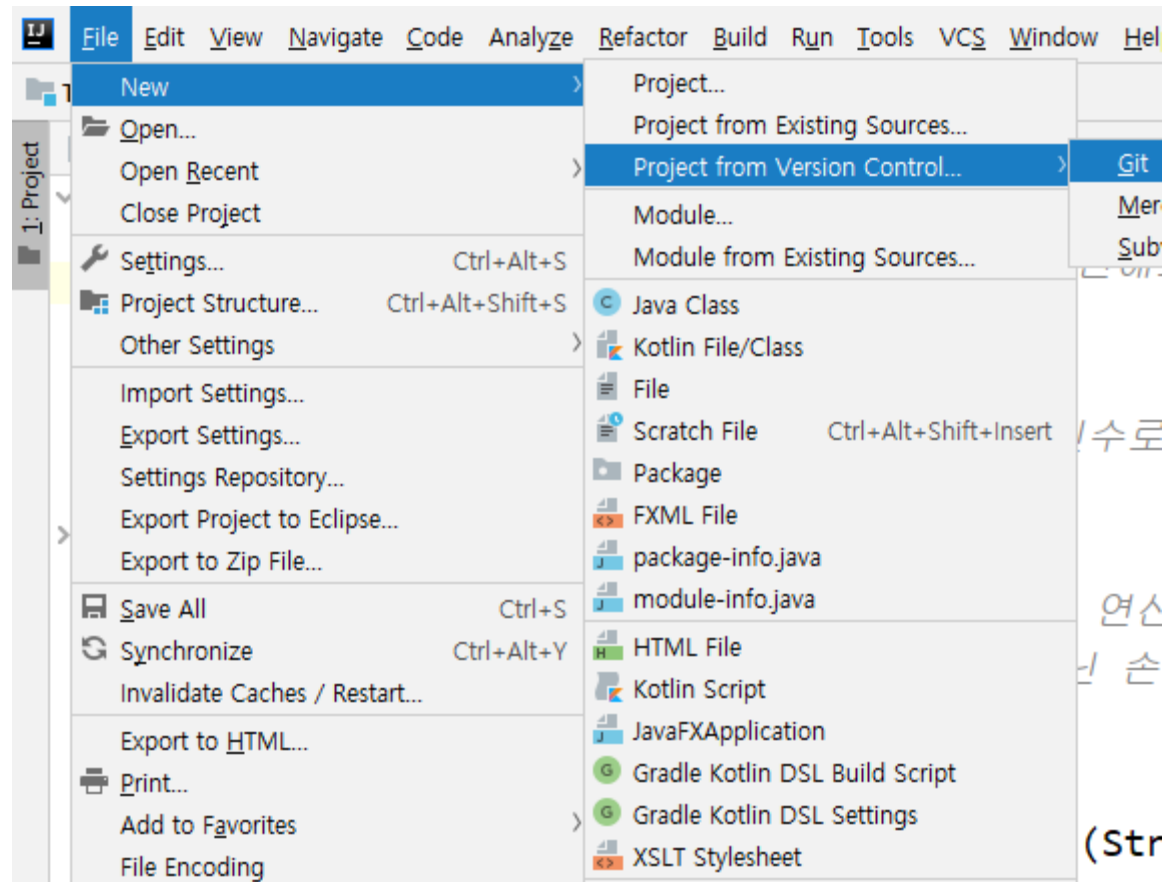


[README.md](#)

Initial co



[README.md](#)



속제 2-3.

Clone Repository

URL:

Directory:

[Log in to GitHub...](#) [?](#) [Clone](#) [Cancel](#) [Test](#)

```
int num1 = 3;  
int num2 = 7;
```

[Edit](#)

leases

1 contributor

MIT

[Create new file](#)

[Upload files](#)

[Find file](#)

[Clone or download](#)

Clone with HTTPS [?](#)

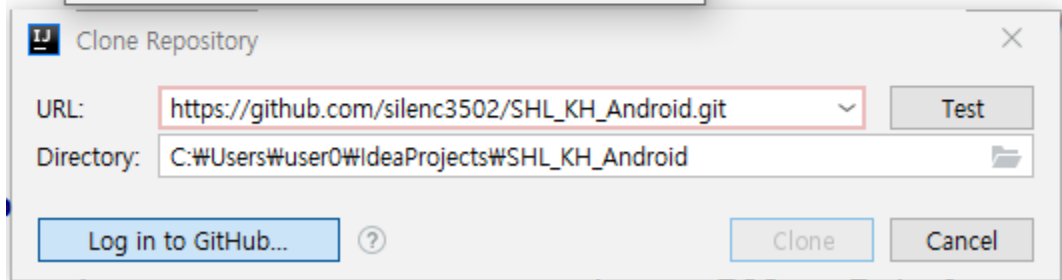
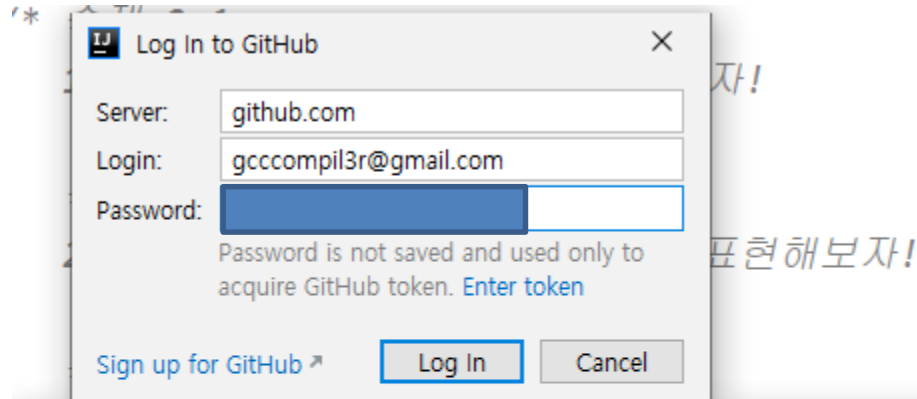
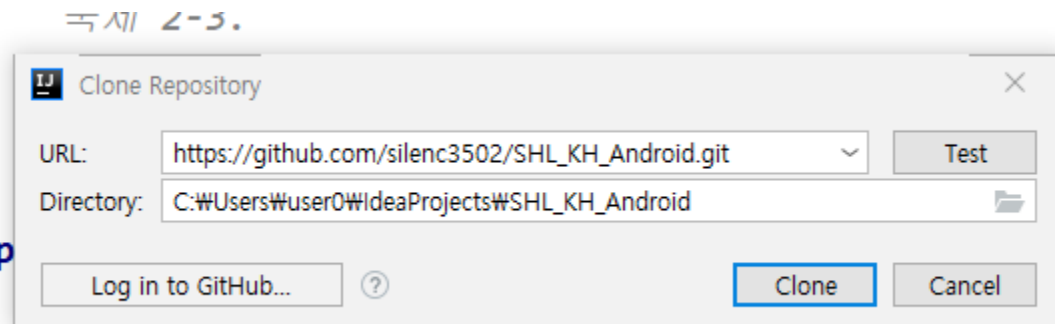
[Use SSH](#)

Use Git or checkout with SVN using the web URL.

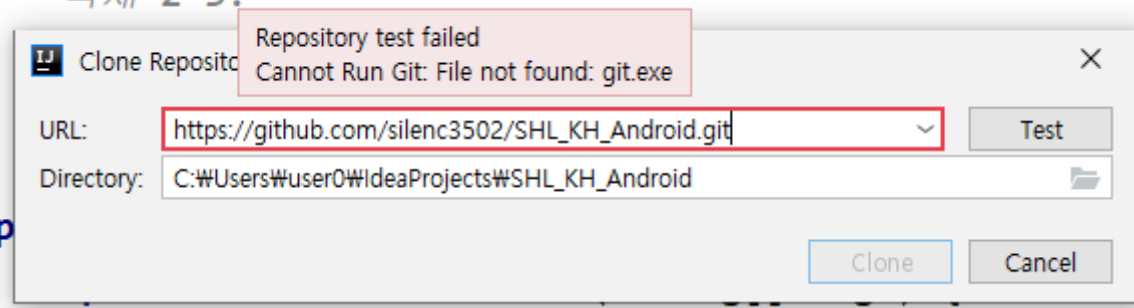


[Open in Desktop](#)

[Download ZIP](#)



속제 2-3.



gitforwindows.org

git for windows

[FAQ](#)

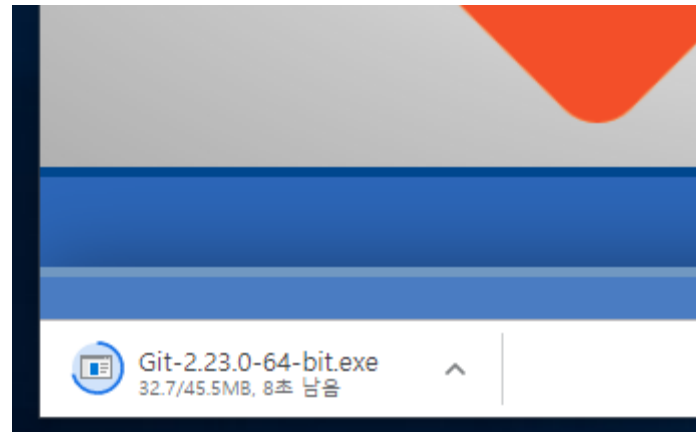
[REPO](#)

VERSION 2.23.0



We bring t
awesome
to Window

[Download](#)





Git 2.23.0 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.

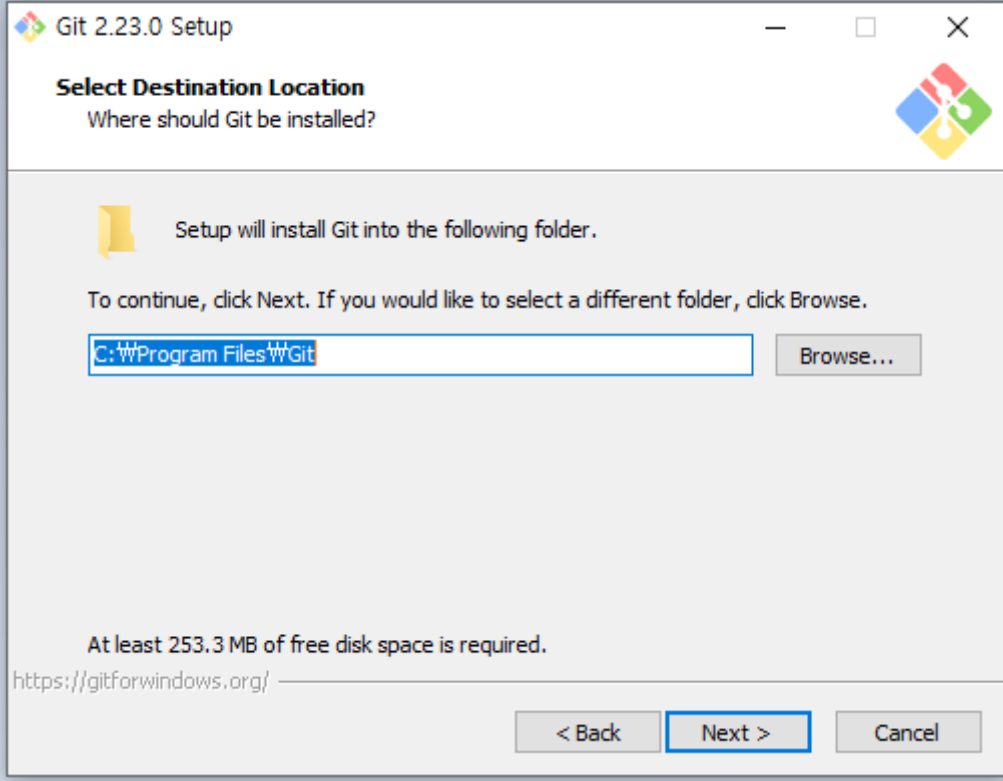
Preamble

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

<https://gitforwindows.org/>

Next >

Cancel



Git 2.23.0 Setup

Select Components
Which components should be installed?

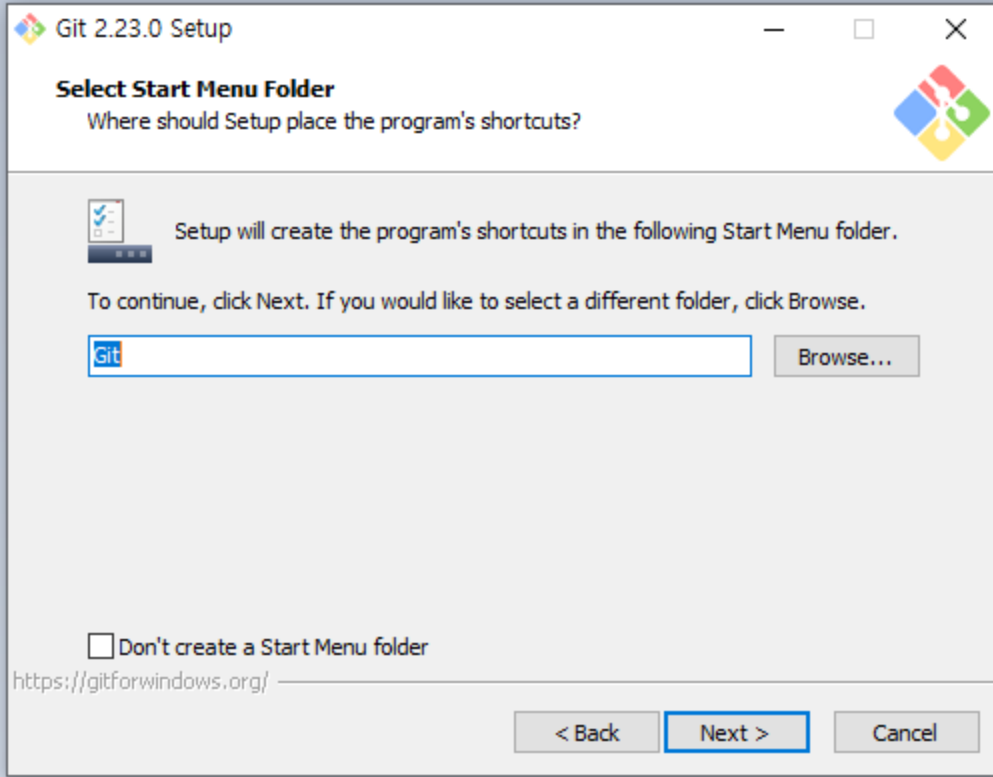
Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue.

- ☐ Additional icons
 - ☐ On the Desktop
- ☒ Windows Explorer integration
 - ☒ 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
- ☐ Use a TrueType font in all console windows
- ☐ Check daily for Git for Windows updates

Current selection requires at least 253.0 MB of disk space.

<https://gitforwindows.org/>

< Back Next > Cancel





Git 2.23.0 Setup



Choosing the default editor used by Git

Which editor would you like Git to use?



Use Vim (the ubiquitous text editor) as Git's default editor



The [Vim editor](#), while powerful, [can be hard to use](#). Its user interface is unintuitive and its key bindings are awkward.

Note: Vim is the default editor of Git for Windows only for historical reasons, and it is highly recommended to switch to a modern GUI editor instead.

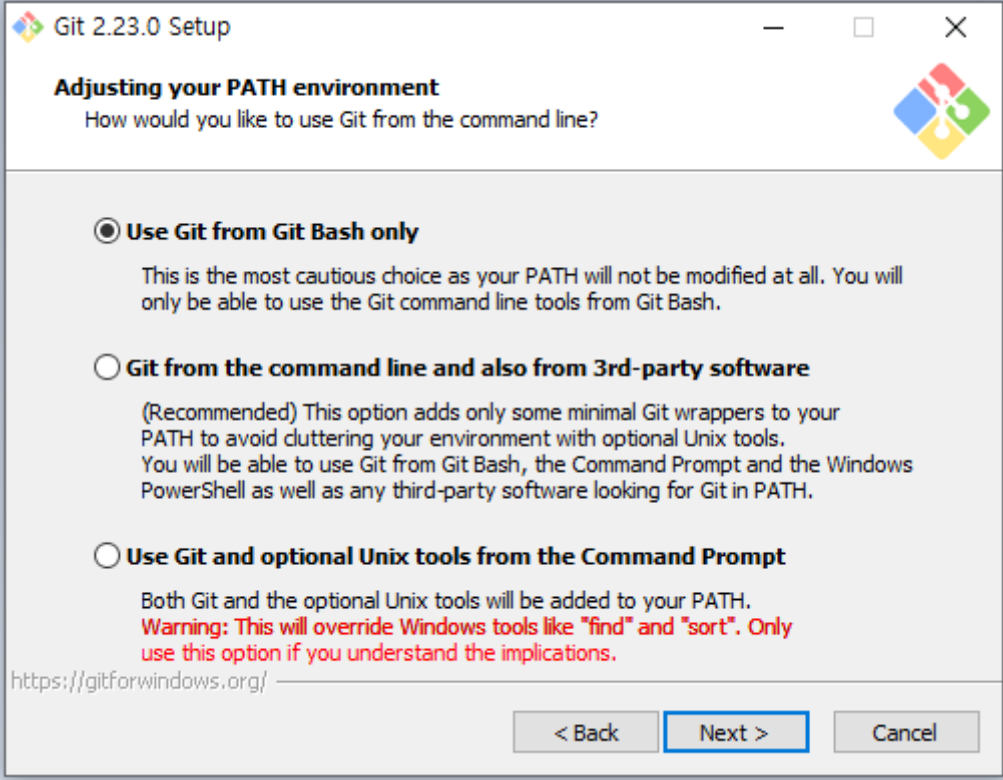
Note: This will leave the 'core.editor' option unset, which will make Git fall back to the 'EDITOR' environment variable. The default editor is Vim - but you may set it to some other editor of your choice.

<https://gitforwindows.org/>

< Back

Next >

Cancel





Git 2.23.0 Setup



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**

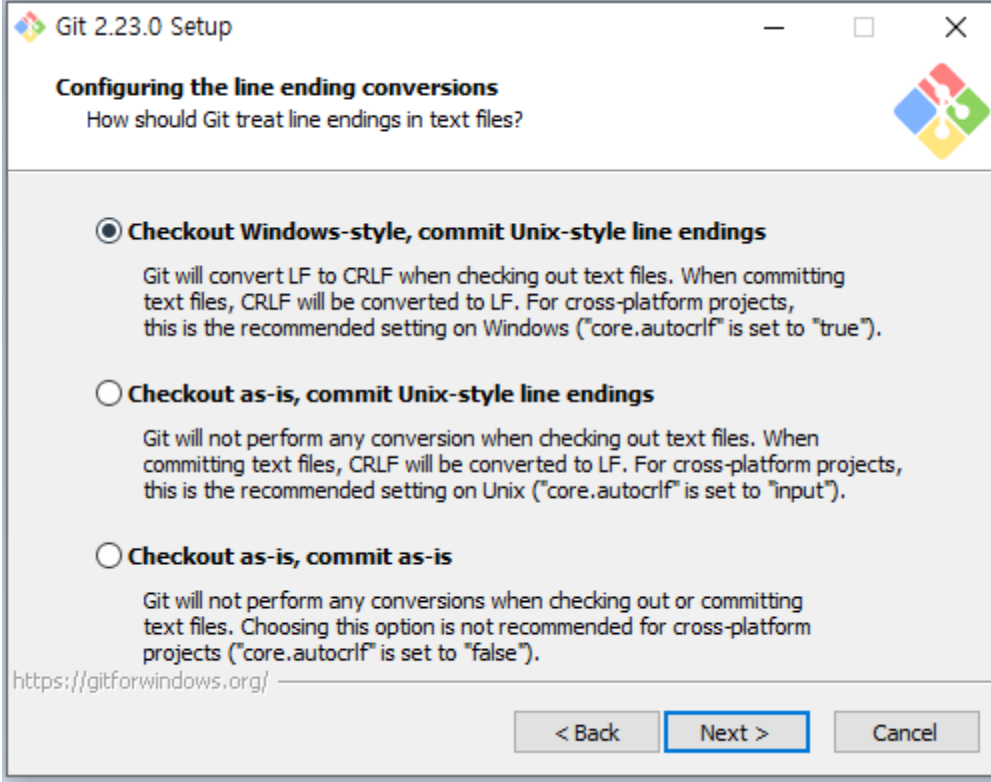
Server certificates will be validated using Windows Certificate Stores.
This option also allows you to use your company's internal Root CA certificates distributed e.g. via Active Directory Domain Services.

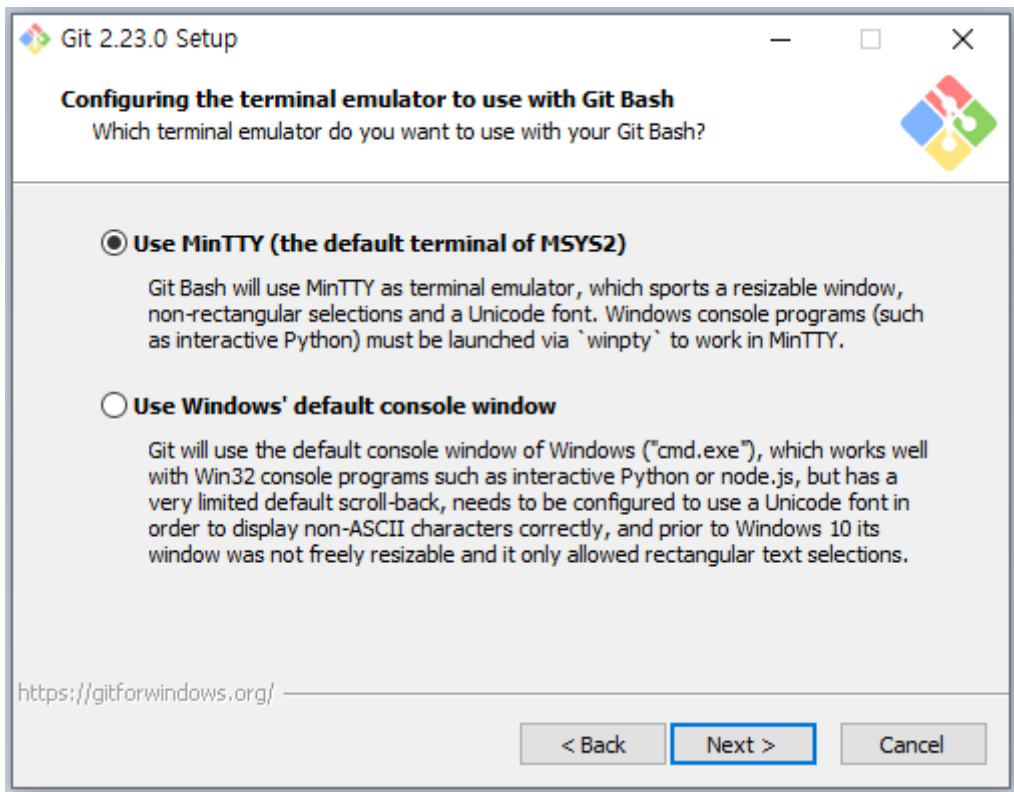
<https://gitforwindows.org/>

< Back

Next >

Cancel





Git 2.23.0 Setup

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



Git 2.23.0 Setup



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 Git Credential Manager**

The [Git Credential Manager for Windows](#) provides secure Git credential storage for Windows, most notably multi-factor authentication support for Visual Studio Team Services and GitHub. (requires .NET framework v4.5.1 or later).

☐ **Enable symbolic links**

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

<https://gitforwindows.org/>

< Back

Next >

Cancel



Git 2.23.0 Setup



Configuring experimental options

Which bleeding-edge features would you like to enable?



☐ **Enable experimental, built-in add -i/-p**

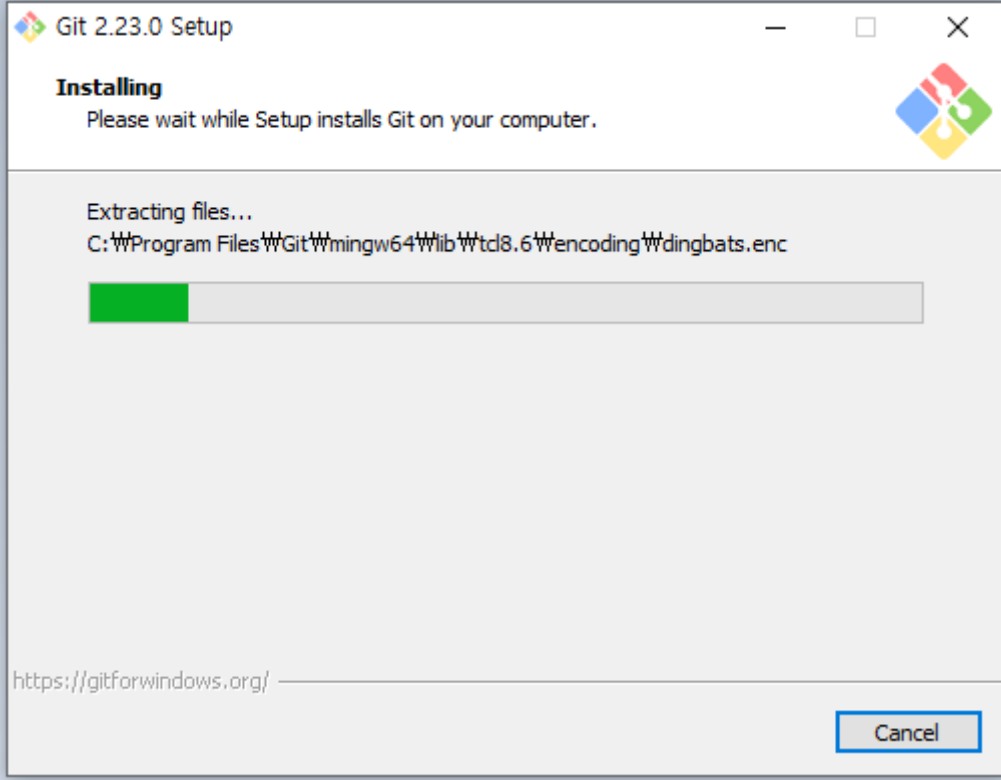
(NEW!) Use the experimental built-in interactive add ("git add -i" or "git add -p"). This makes it faster (especially the startup!), but it is not yet considered robust.

<https://gitforwindows.org/>

< Back

Install

Cancel



Completing the Git Setup Wizard

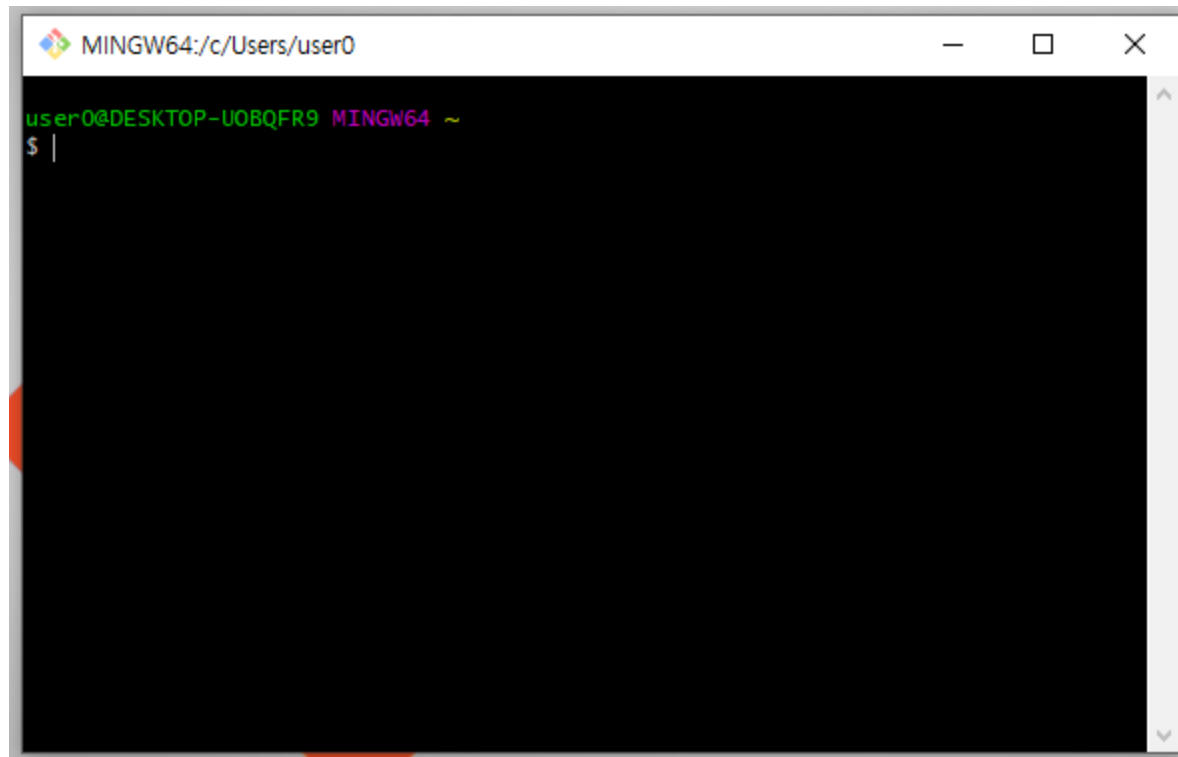
Setup has finished installing Git on your computer. The application may be launched by selecting the installed shortcuts.

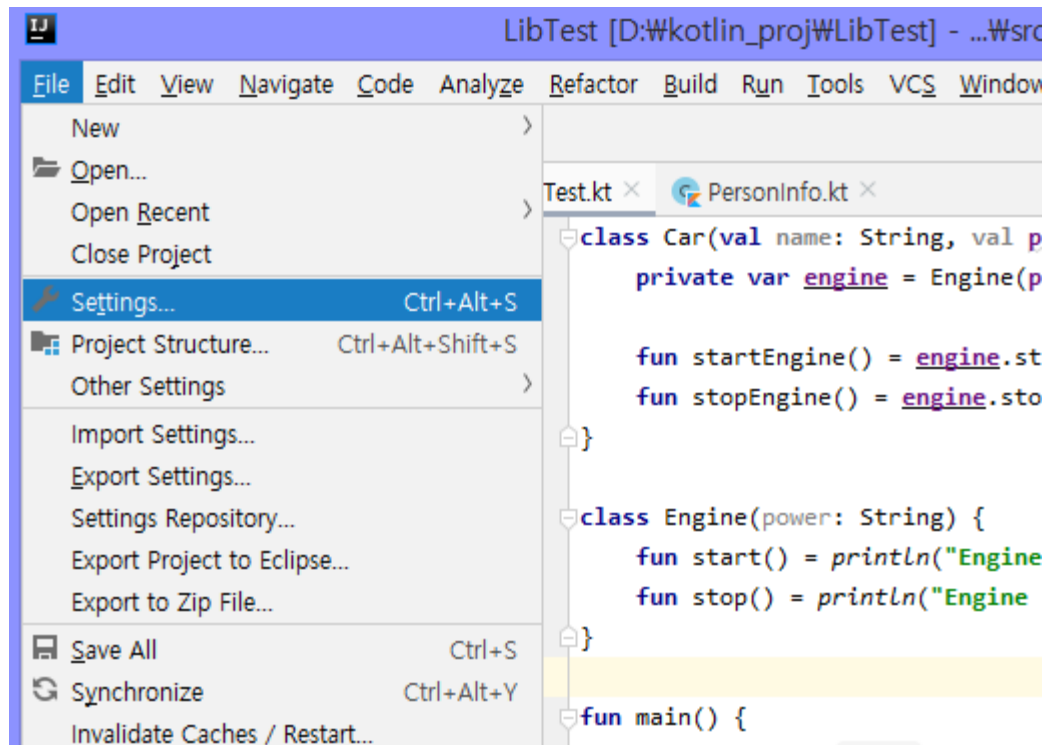
Click Finish to exit Setup.

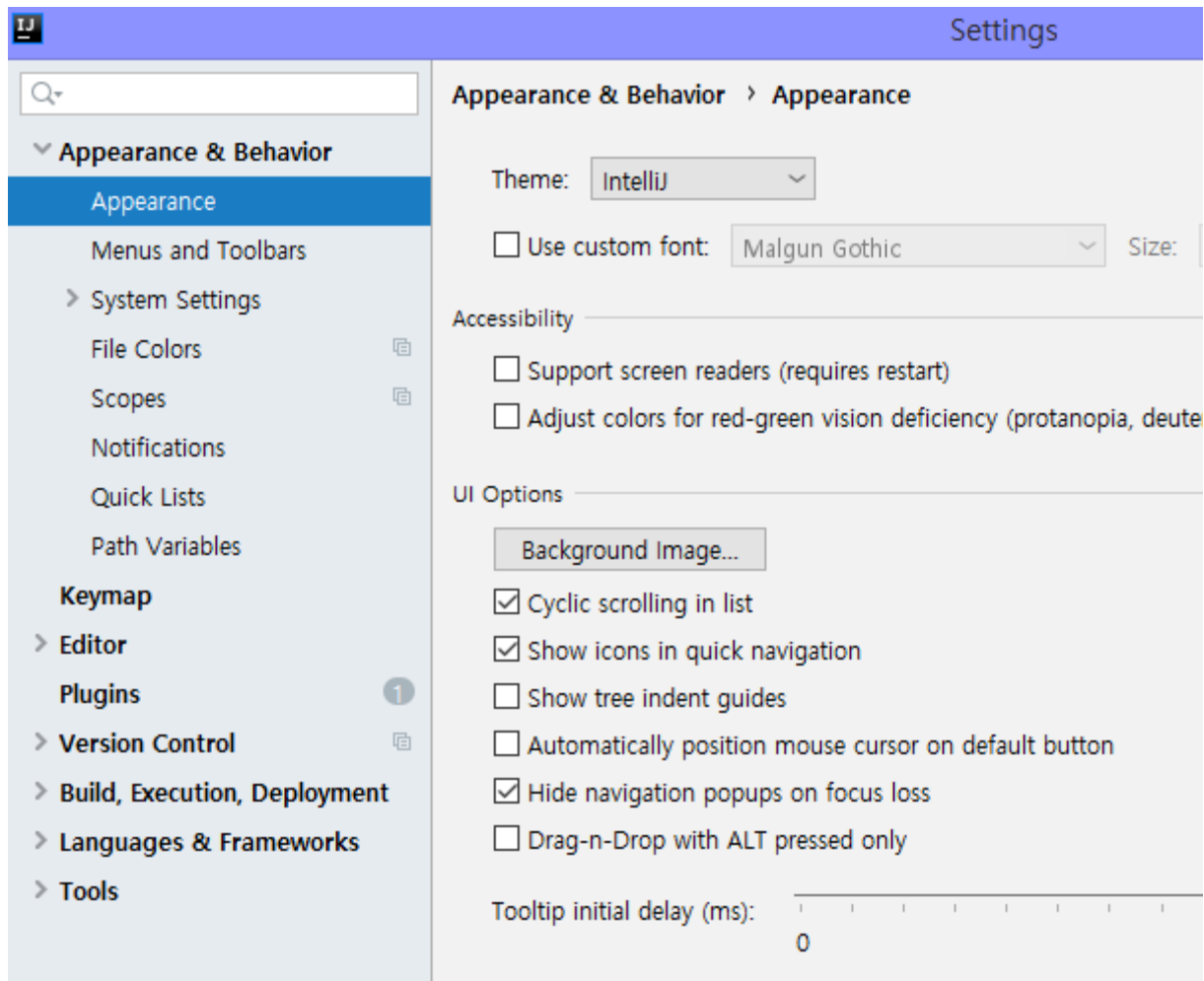
- ☒ Launch Git Bash
- ☐ View Release Notes

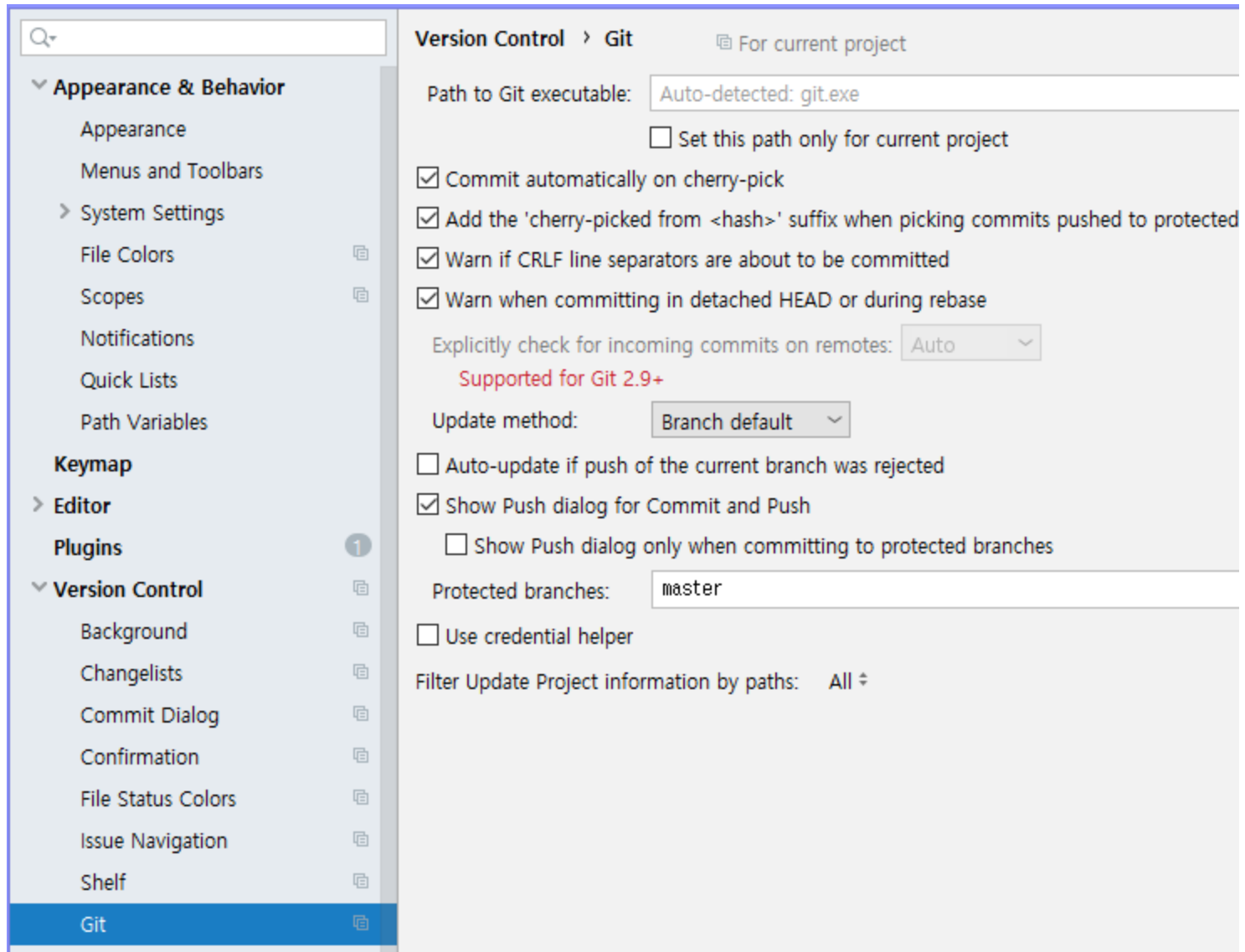


Finish









Path to Git executable: Auto-detected: git.exe



Test

☐ Set this path only for current project☒ Commit automatically☒ Add the 'cherry-pick' command☒ Warn if CRLF line endings☒ Warn when committing

Explicitly check for

Supported for Git

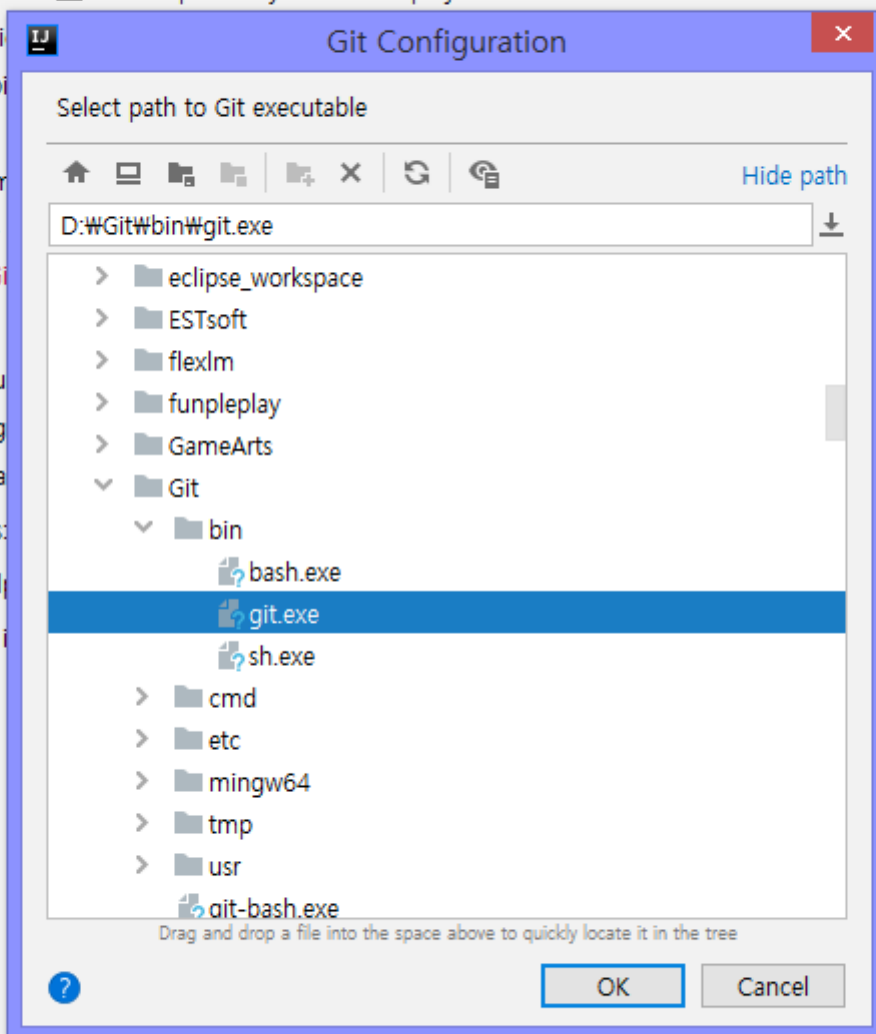
Update method:

☐ Auto-update if pushed☒ Show Push dialog☐ Show Push dialog

Protected branches:

☐ Use credential helper

Filter Update Project



Executable:

☐ Set this path only for current project

Automatically on cherry-pick

Cherry-picked from '<hash>' suffix when picking commits pushed to protected branches

File line separators are about to be committed

Committing in detached HEAD or during rebase

Link for incoming commits on remotes:

for Git 2.9+

id:

if push of the current branch was rejected

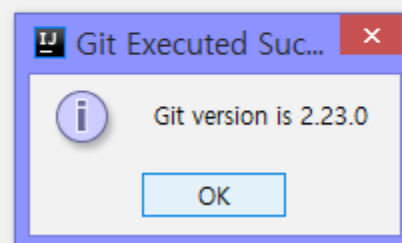
Dialog for Commit and Push

Show dialog only when committing to protected branches

Branches:

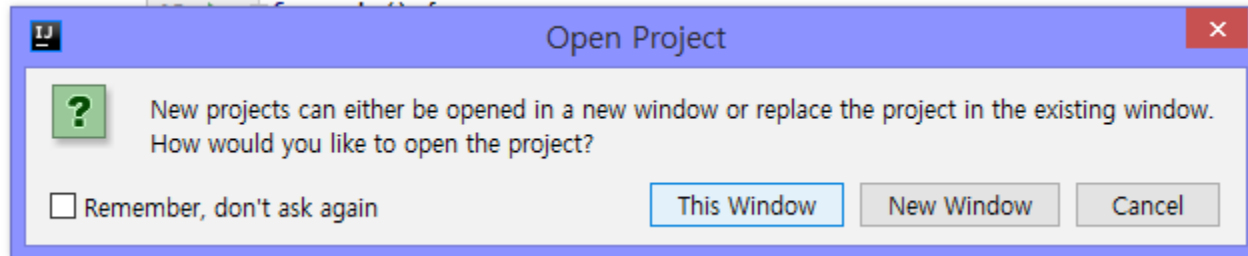
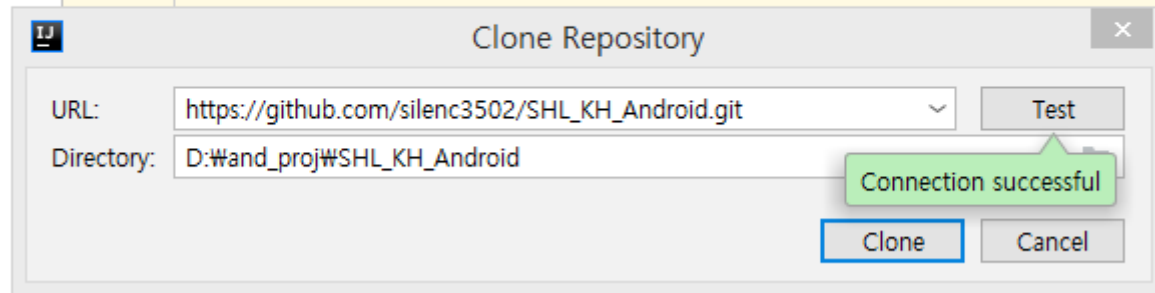
Global helper

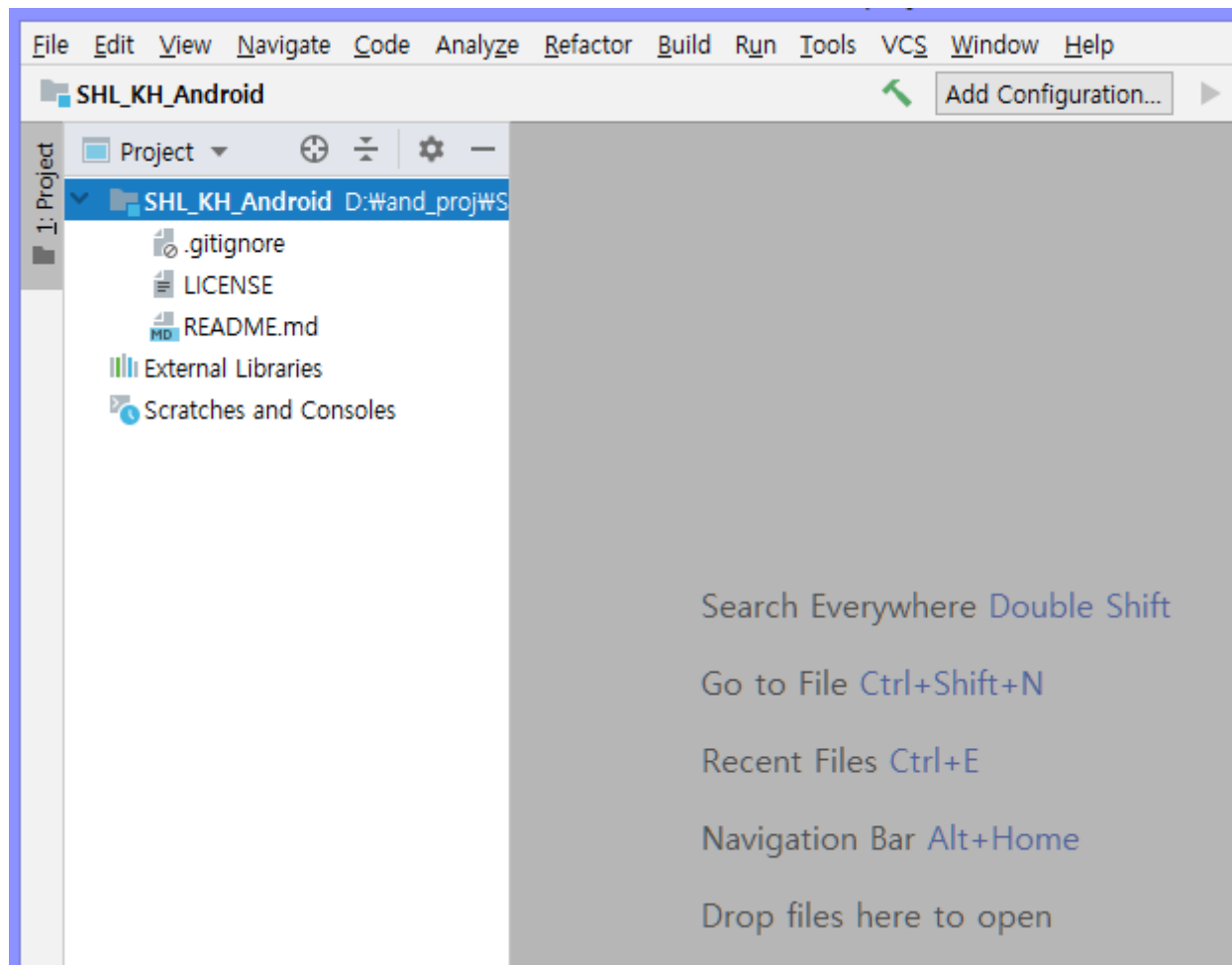
Show object information by paths:

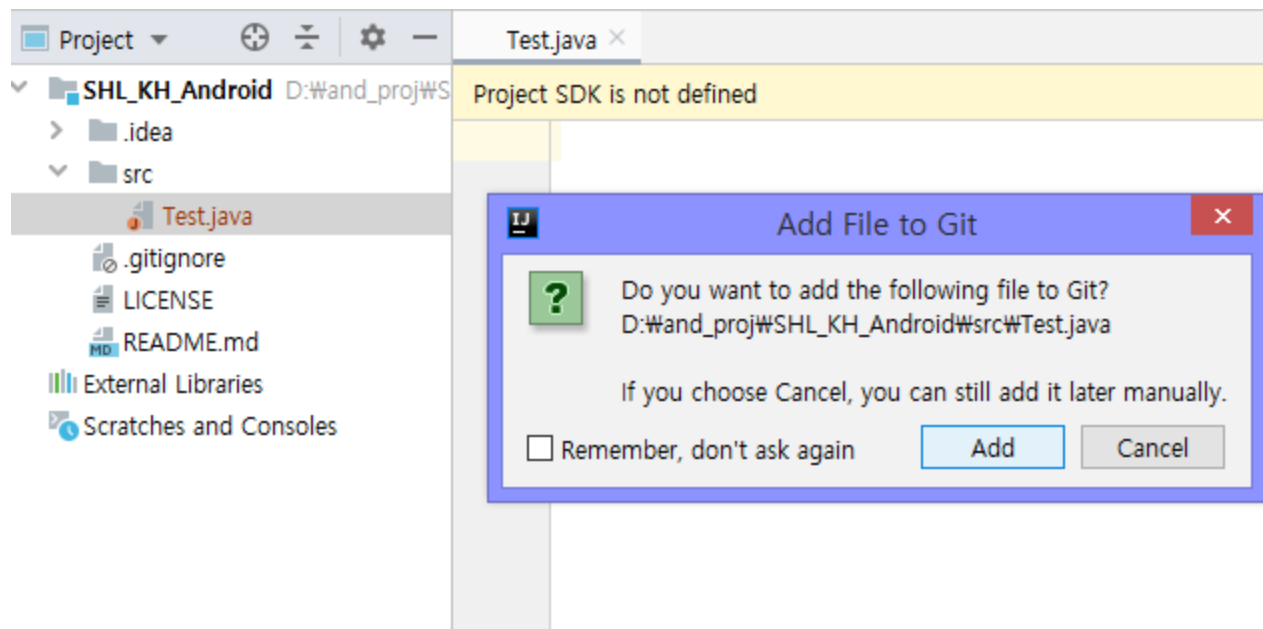


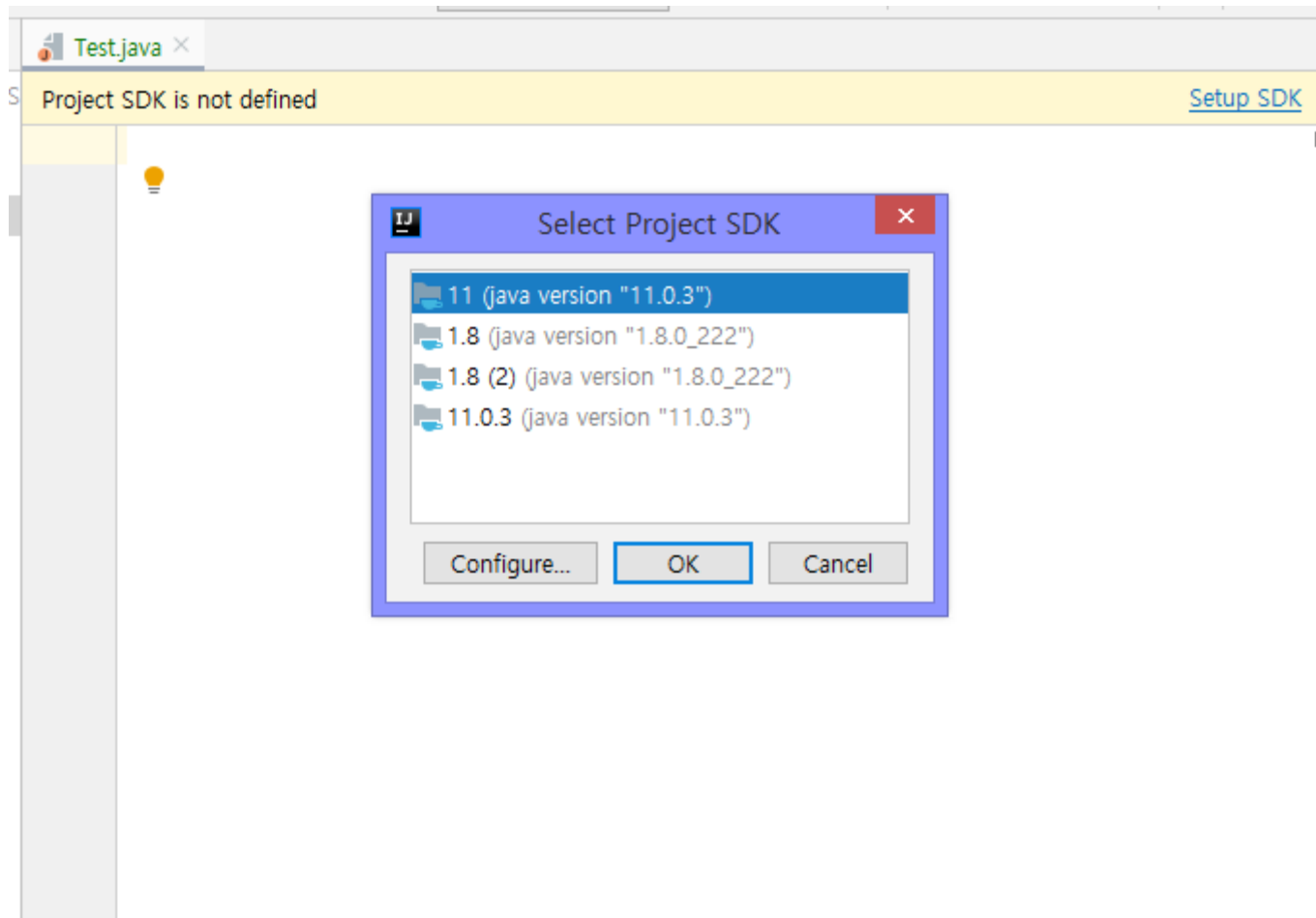
les

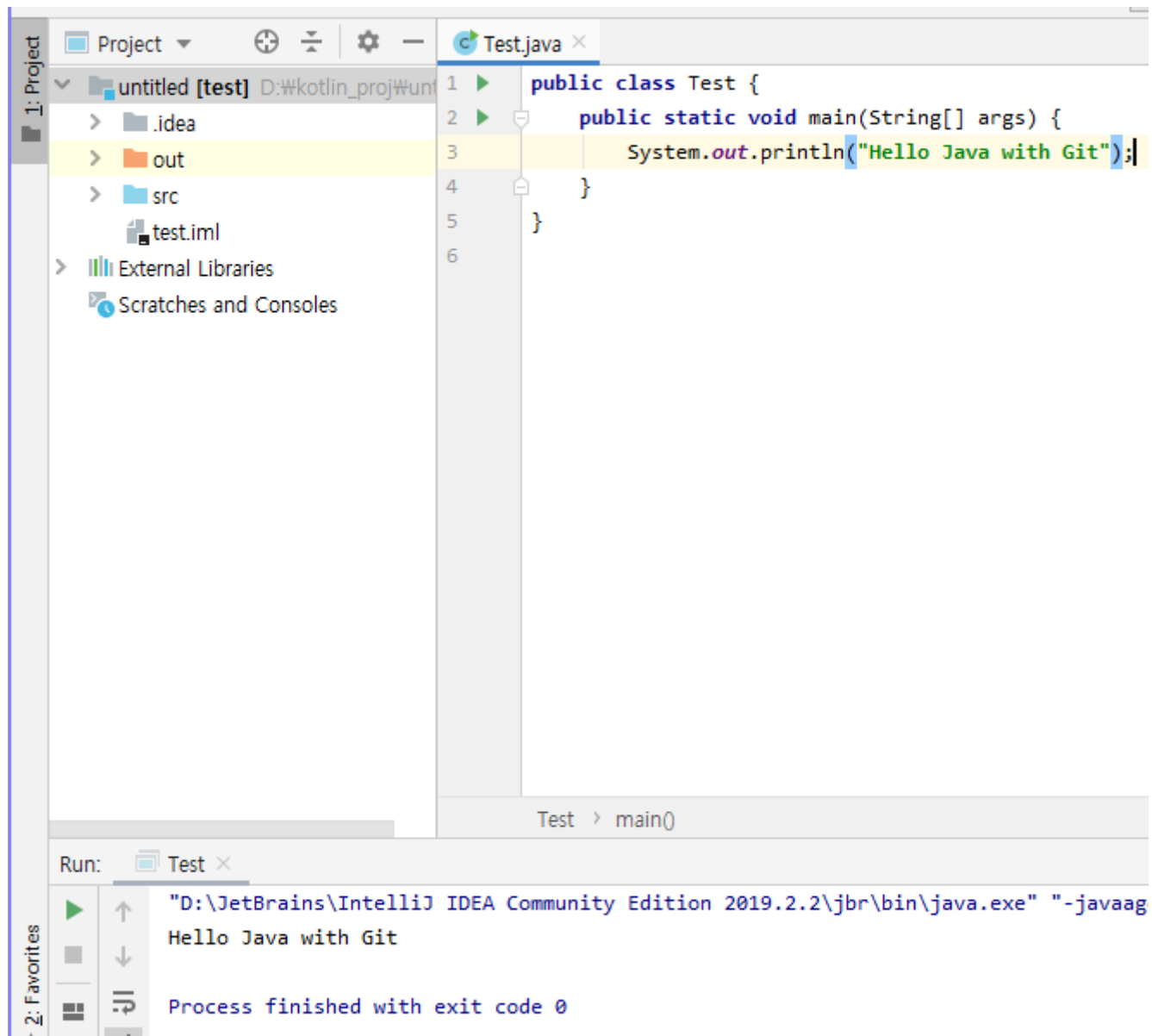
```
9      fun start() = println("Engine has been started")
10     fun stop() = println("Engine has been stopped")
11 }
12
```

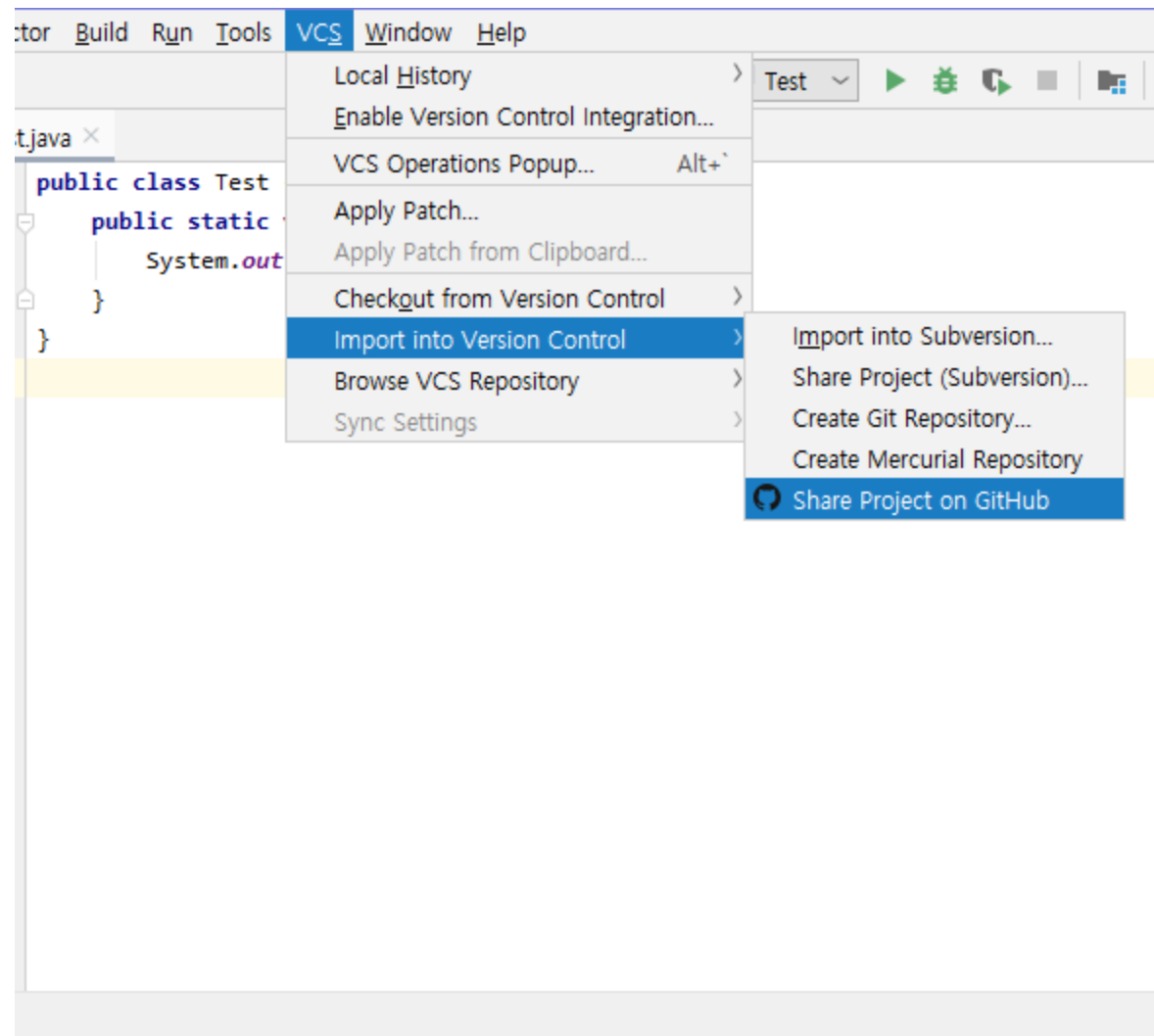


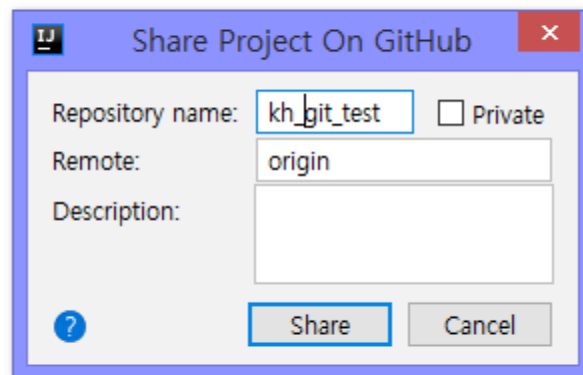












A screenshot of a Windows-style dialog box titled "Share Project On GitHub". The dialog has a blue title bar with a small icon on the left and a red close button on the right. The main content area is light gray and contains three input fields: "Repository name:" with the text "kh_git_test", "Remote:" with the text "origin", and "Description:" which is an empty text area. To the right of the "Repository name:" field is a checkbox labeled "Private" which is currently unchecked. At the bottom left of the dialog is a blue circular help icon with a white question mark. At the bottom right are two buttons: "Share" and "Cancel".

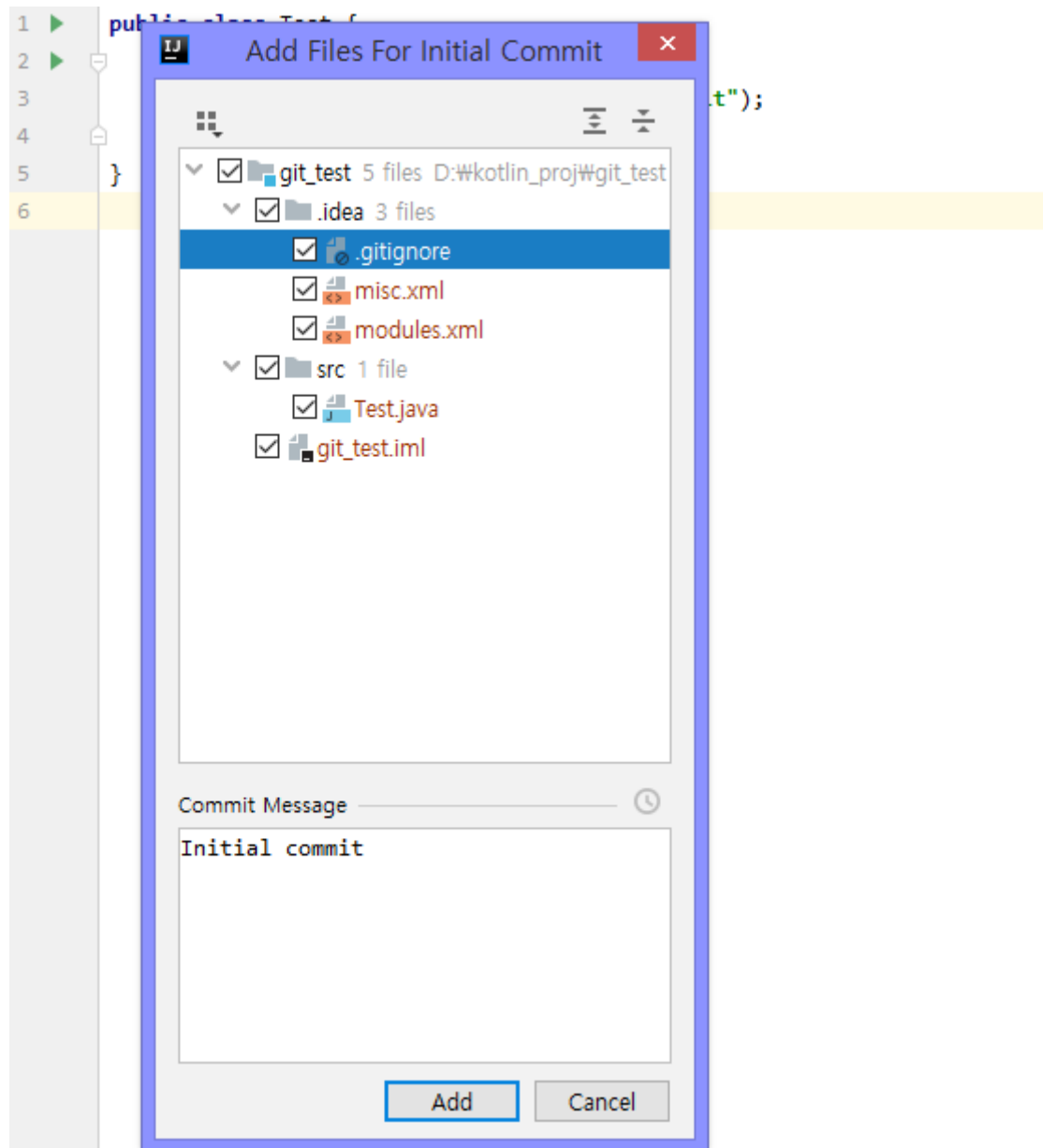
Share Project On GitHub

Repository name: kh_git_test ☐ Private

Remote: origin

Description:

? Share Cancel



```
user@TorAleksander MINGW64 ~  
$ git config --global user.email "gcccompil3r@gmail.com"  
  
user@TorAleksander MINGW64 ~  
$ git config --global user.name "silenc3502"  
  
user@TorAleksander MINGW64 ~  
$ |
```



Can't finish GitHub sharing process

Successfully created project '[kh_git_test](#)' on GitHub, but initial commit failed:

*** Please tell me who you are. Run git config --global user.email "you@example.com" git config --global user.name "Your Name" to set your account's default identity. Omit --global to set the identity only in this repository. unable to auto-detect email address (got 'user@TorAleksander.(none)')



Using the Hello World guide, you'll start a branch, writ

[Read the guide](#)

[/ kh_git_test](#)

[Issues](#) 0

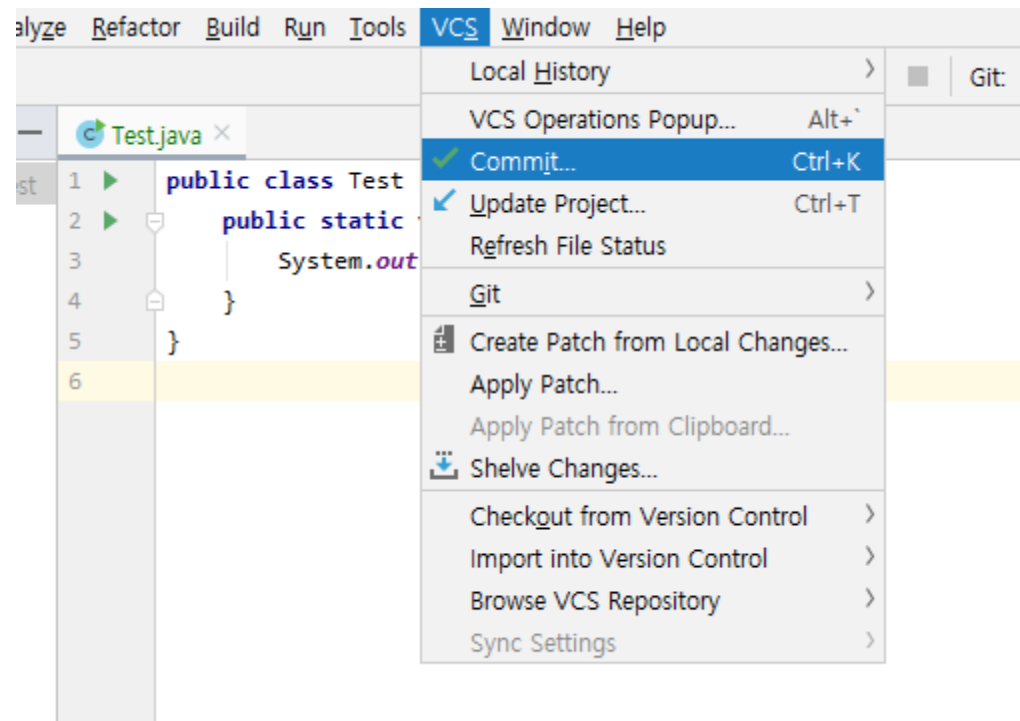
[Pull requests](#) 0

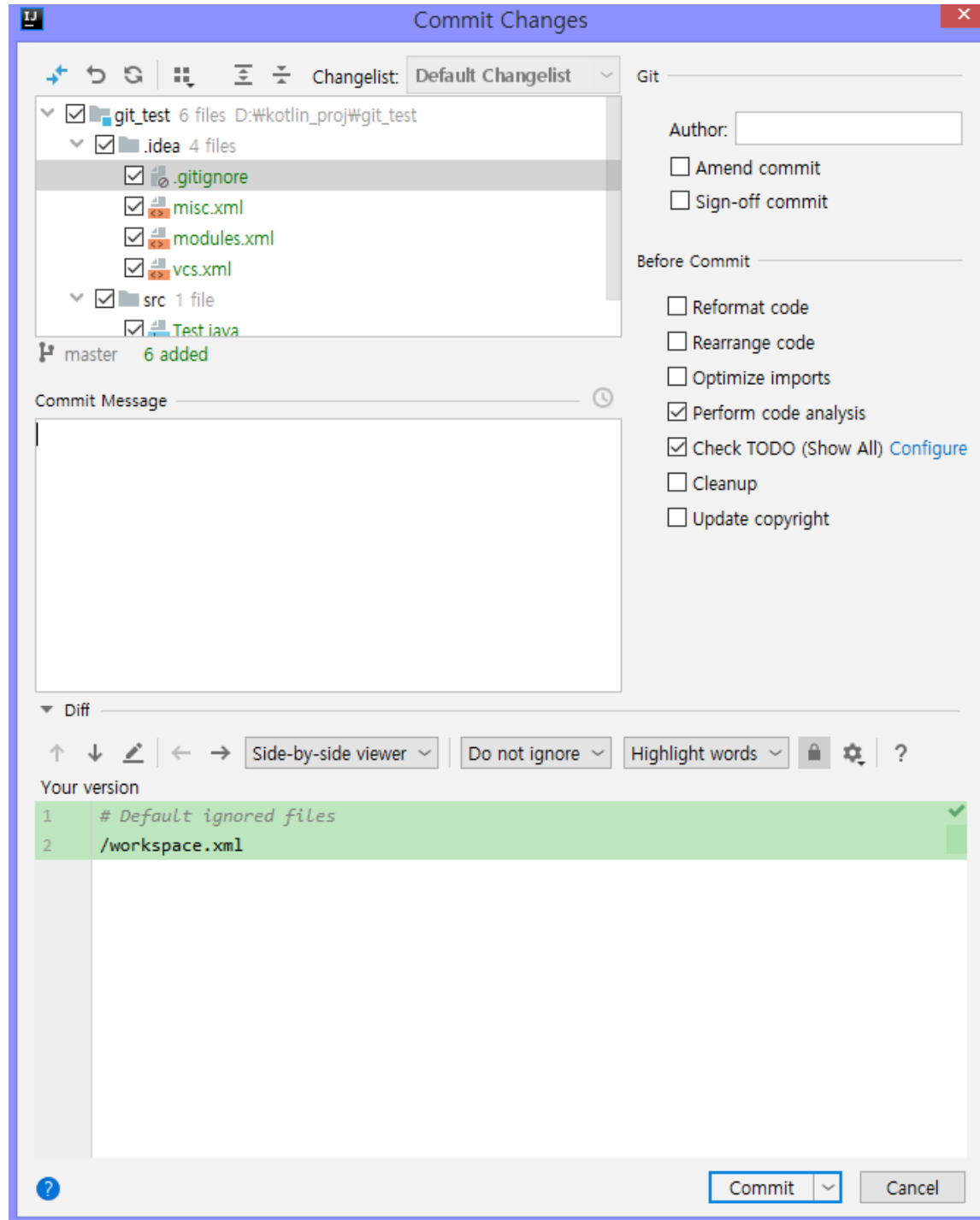
[Projects](#) 0

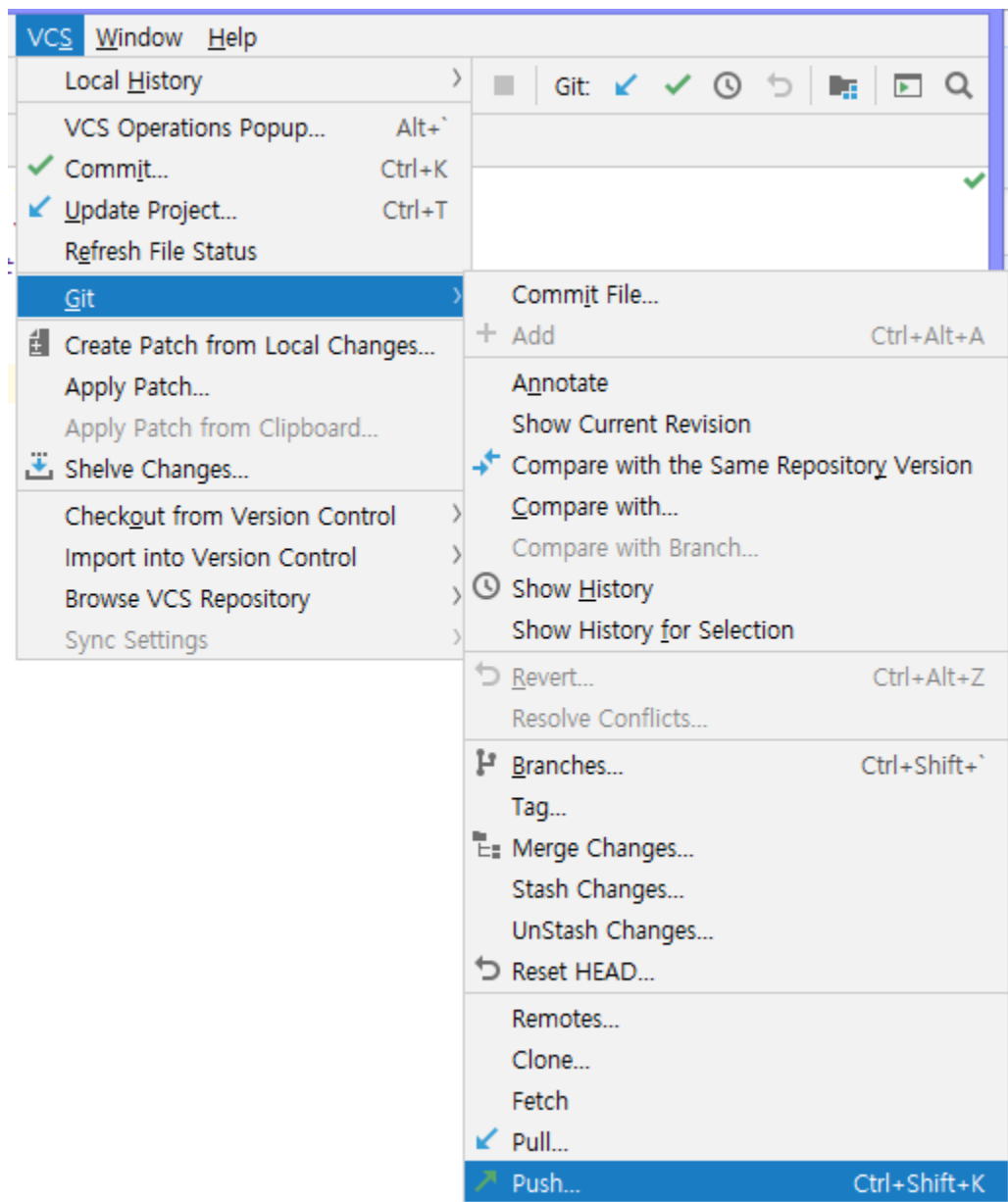
[Wiki](#)

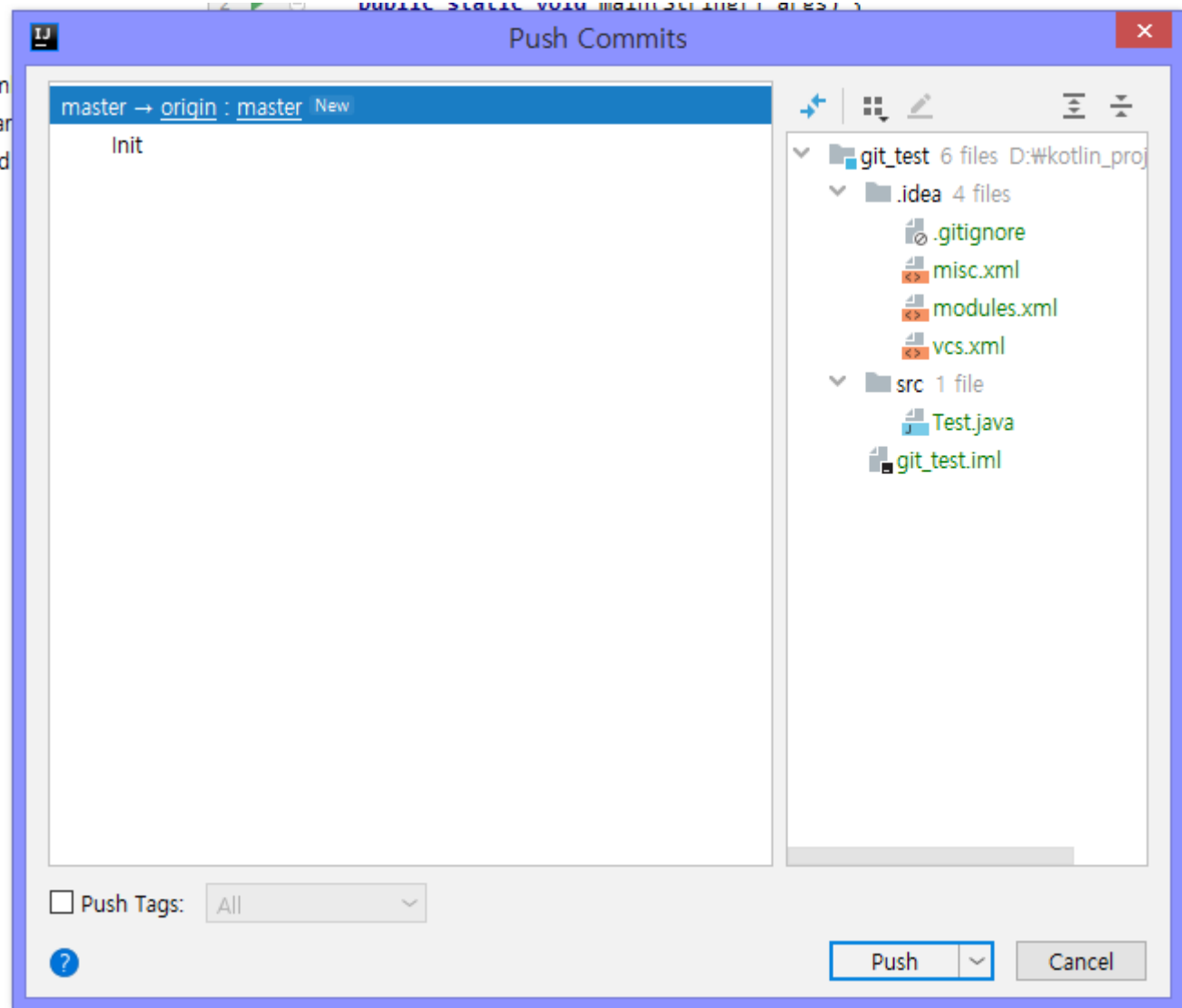
[Security](#)

Setup — if you've done this kind of thing before









silenc3502 / kh_git_test

↔ Code

! Issues 0

🔗 Pull requests 0

📁 Projects 0

📖 Wiki

🛡️ Security

⌵ description, website, or topics provided.

[image topics](#)

🕒 1 commit

🔗 1 branch

Branch: master ▼

New pull request



silenc3502 Init

📁 .idea

📁 src

📄 git_test.iml