

# GIT

버전 관리의 새로운 표준

Git은 전 세계 개발자들이 사용하는 강력한 버전 관리 시스템입니다.

프로젝트의 변경 사항을 효과적으로 추적하고 관리하는 방법을 함께 알아보겠습니다.

# 1.GIT 개요

GIT은 소프트웨어 개발에서 가장 널리 사용되는 버전 관리 시스템입니다.

개발자들의 협업을 원활하게 하고, 코드의 변경 사항을 효과적으로 추적하며, 프로젝트의 전체 히스토리를 안전하게 보관합니다.

## Git이란?

**Git**은 컴퓨터 파일을 안전하게 저장하고, 수정한 기록을 기억해 주는 도구입니다.

마치 파일을 관리하는 "**스마트 노트북**"이라고 할 수 있습니다.

여러 사람이 같이 작업할 때도 누가 어떤 부분을 고쳤는지 정리해 주고, 실수했을 때 다시 고칠 수 있게 도와줍니다.

# 학교 과제 파일 관리

1. 학교에서 친구들과 함께 보고서를 만들고 있습니다.
2. 각자 작성한 내용을 통합해서 매번 파일에 저장하는데, 파일 이름이 이렇게 됩니다:
  - 보고서\_v1.docx
  - 보고서\_v2\_final.docx
  - 보고서\_v3\_final\_FINAL.docx
3. 시간이 지날수록 파일 이름이 복잡해지고,
4. 누가 어떤 부분을 고쳤는지 찾기도 힘듭니다.
4. 만약 실수로 잘못 고쳤다면 다시 원래대로 돌리기가 어렵습니다

## Git은 이런 문제를 해결합니다.

1. 파일의 수정 기록을 자동으로 관리합니다.
2. 실수했을 때 이전 상태로 되돌릴 수 있습니다.
3. 여러 사람이 동시에 작업해도 파일이 겹치거나 지워지지 않게 해 줍니다.

# Git의 작동 방법

## 1. 저장소 만들기

- Git은 "저장소(repository)"라는 곳에 파일을 저장합니다.
- 저장소는 파일의 수정 기록을 안전하게 보관하는 "금고"라고 생각하면 됩니다.

## 2. 수정 내용 저장하기

- 파일을 수정하면 Git에게 "이 내용 저장해!"라고 알려줘야 합니다.
- 이를 **커밋(commit)**이라고 합니다.
- 커밋은 "이 시점에서 저장하자"는 표시입니다.

## 3. 필요하면 되돌리기

- 실수하거나 잘못 고친 파일이 있다면, Git을 통해 이전 상태로 되돌릴 수 있습니다.

# Git의 사용 이유

## 1. 파일을 안전하게 보관하기 위해

- 실수로 파일을 삭제해도 Git은 이전 버전을 기억하고 있어서 복구할 수 있습니다.

## 2. 여러 사람이 동시에 작업하기 위해

- 친구들과 협업할 때도 서로의 파일을 덮어쓰지 않게 관리해 줍니다.

## 3. 수정 기록을 남기기 위해

- 누가 언제 무엇을 수정했는지 기록이 남아서, 문제가 생겼을 때 쉽게 확인할 수 있습니다.

# Git의 파일 저장 위치

## 1. 여러분의 컴퓨터

- Git은 기본적으로 여러분의 컴퓨터에 파일을 저장합니다.

## 2. GitHub 같은 클라우드

- GitHub은 Git 저장소를 인터넷에 저장해 주는 서비스입니다.
- 마치 Google Drive처럼, 언제 어디서나 접근할 수 있고 팀과 공유할 수 있습니다.

# Git과 GitHub의 차이점

## 1. **Git:** 여러분의 컴퓨터에서 작동하는 프로그램.

→ 파일을 기록하고 관리합니다.

## 2. **GitHub:** Git 저장소를 인터넷에 올릴 수 있는 서비스.

→ 팀원들과 파일을 공유하고, 어디서나 접근할 수 있습니다.

## 2. Git 설치


설치 시 기본 옵션을 유지하는 것을 권장하며, 특별한 설정이 필요한 경우 고급 옵션에서 조정할 수 있습니다.



# Git 설치

<https://git-scm.com/> 접속합니다

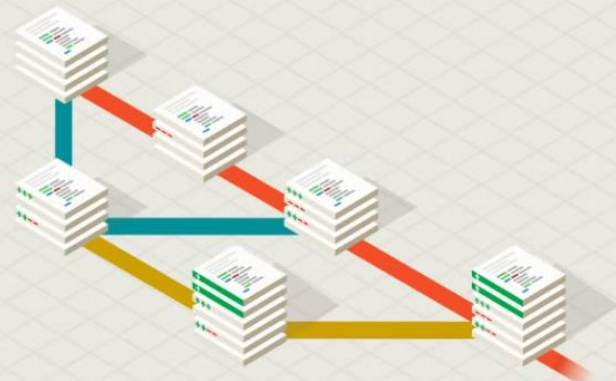
Downloads 를 선택합니다


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

Type / to search entire site...


Git is a **free and open source** distributed version control system designed to handle everything from small to very large projects with speed and efficiency.

Git is **easy to learn** and has a **tiny footprint with lightning fast performance**. It outclasses SCM tools like Subversion, CVS, Perforce, and ClearCase with features like **cheap local branching**, convenient **staging areas**, and **multiple workflows**.




**About**


The advantages of Git compared to other source control systems.

**Documentation**


Command reference pages, Pro Git book content, videos and other material.

**Downloads**


GUI clients and binary releases for all major platforms.

**Community**


Get involved! Bug reporting, mailing list, chat, development and more.





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




Latest source Release  
**2.47.1**  
Release Notes (2024-11-25)  
**Download for Windows**

 **Windows GUIs**


 **Tarballs**

 **Mac Build**

 **Source Code**

# Git 설치

64 bit 윈도우용 프로그램을 다운로드 받습니다

 **git** --fast-version-control

Type / to 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](#).

## Download for Windows

[Click here to download](#) the latest (2.47.1) **64-bit** version of **Git for Windows**. This is the most recent [maintained build](#). It was released **5 days ago**, on 2024-11-25.

### Other Git for Windows downloads

**Standalone Installer**  
[32-bit Git for Windows Setup.](#)  
**64-bit Git for Windows Setup.**


**Portable ("thumbdrive edition")**  
[32-bit Git for Windows Portable.](#)  
**64-bit Git for Windows Portable.**

**Using winget tool**  
Install [winget tool](#) if you don't already have it, then type this command in command prompt or Powershell.  
`winget install --id Git.Git -e --source winget`

The current source code release is version **2.47.1**. If you want the newer version, you can build it from [the source code](#).


### Now What?

Now that you have downloaded Git, it's time to start using it.




#### Read the Book

Dive into the Pro Git book and learn at your own pace.



#### Download a GUI

Several free and commercial GUI tools are available for the Windows platform.

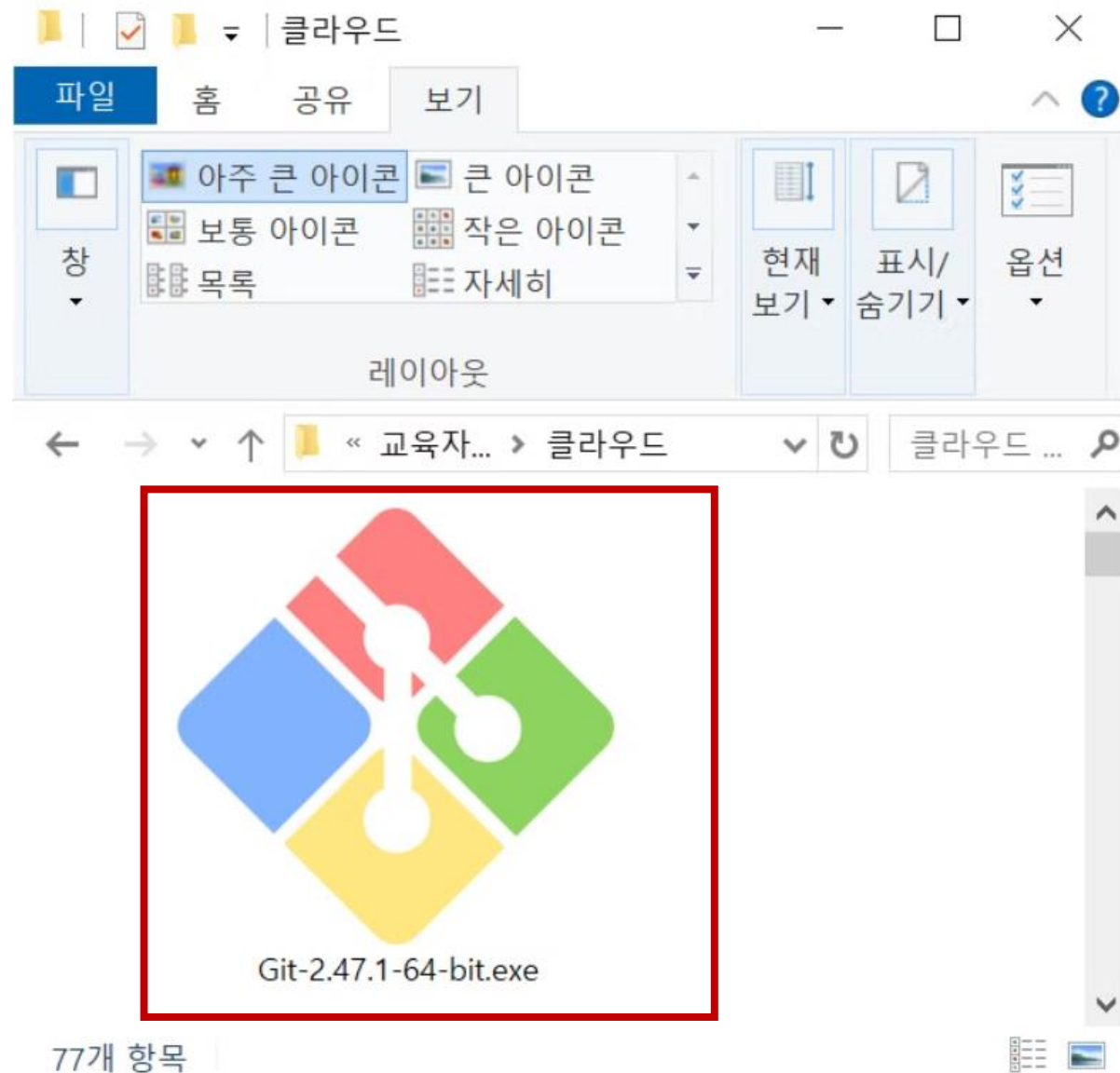


#### Get Involved

A knowledgeable Git community is available to answer your questions.

# Git 설치

다운로드 받은 설치 파일을 더블클릭합니다



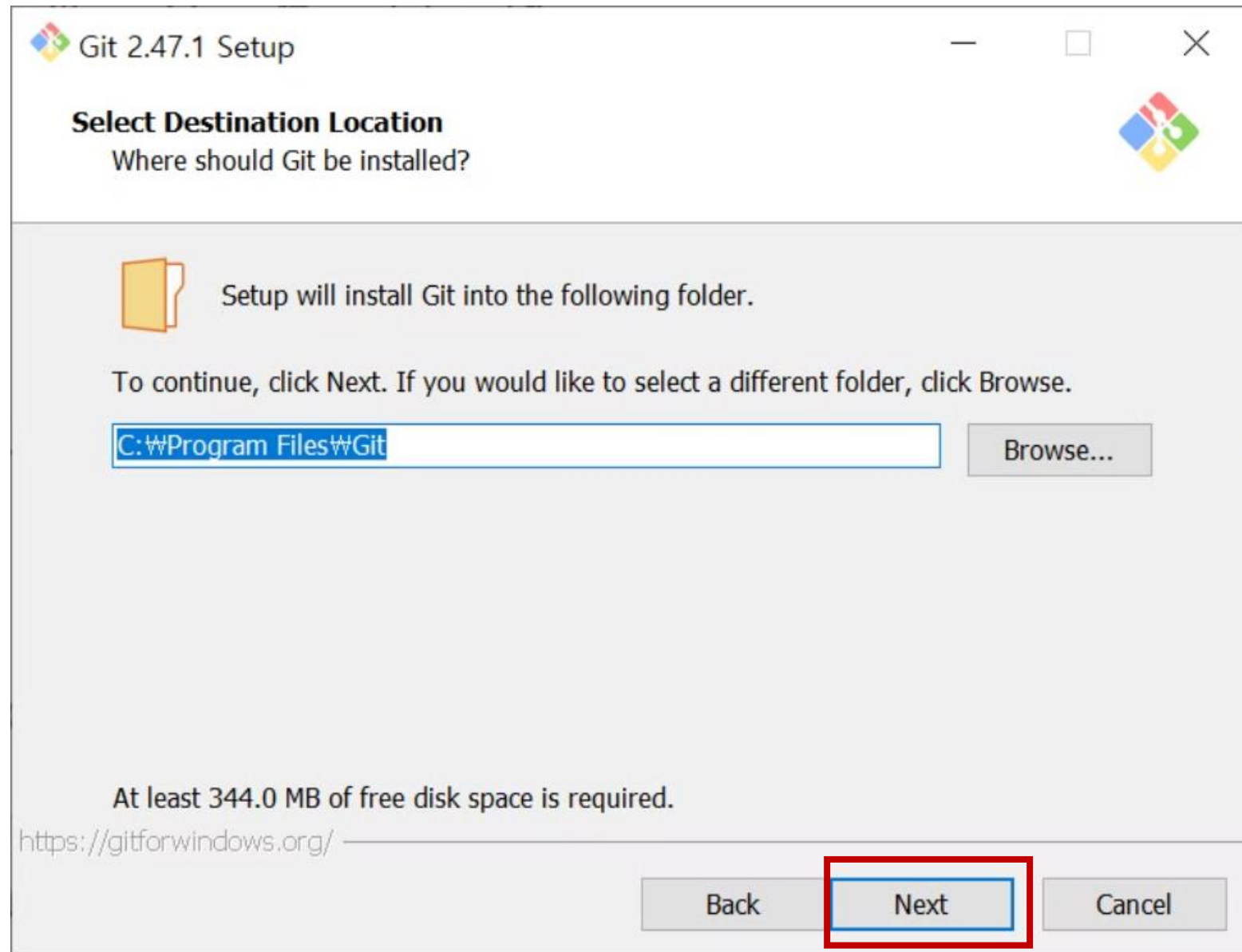
# Git 설치

기본 설정으로 설치합니다



# Git 설치

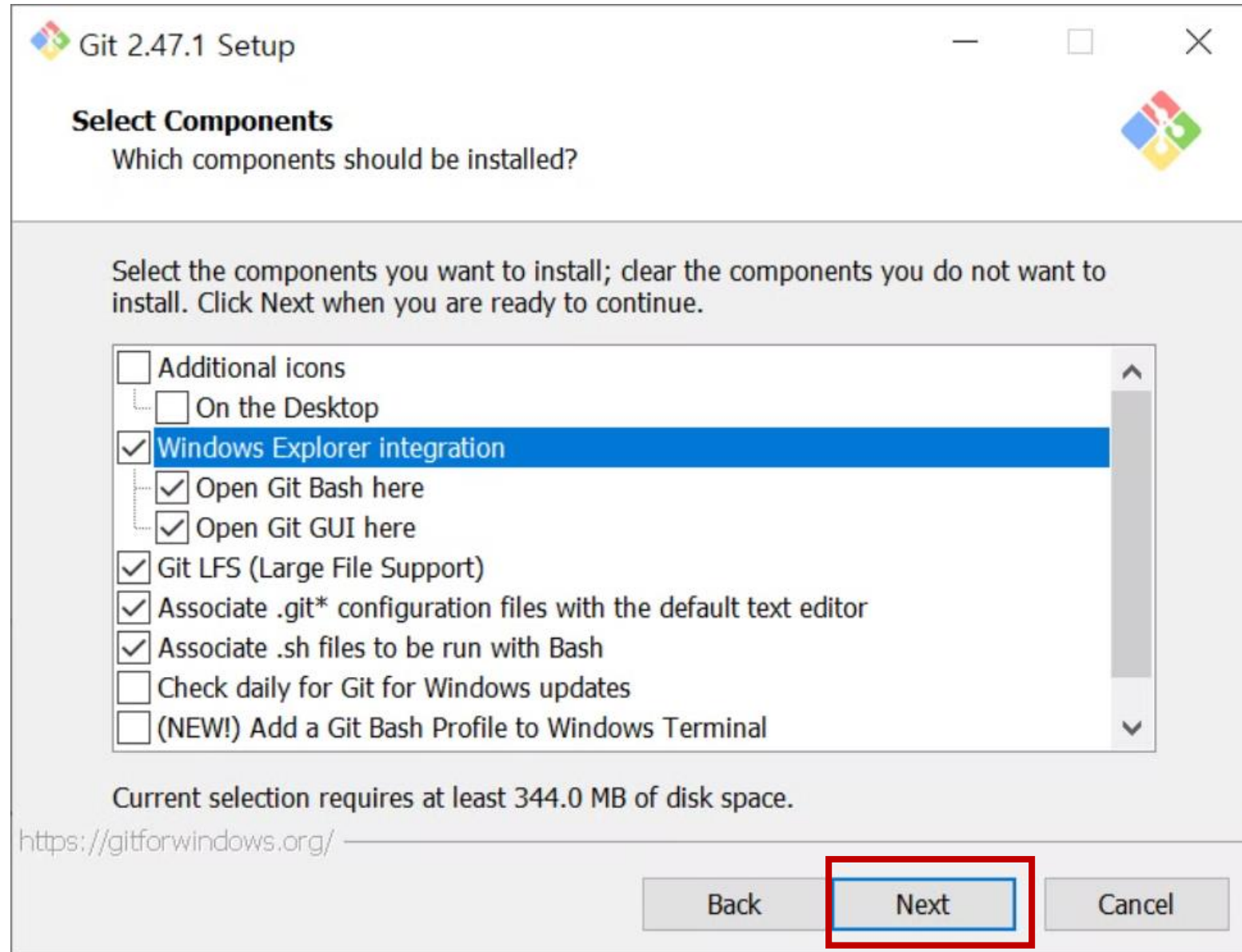
기본 설정으로 설치합니다





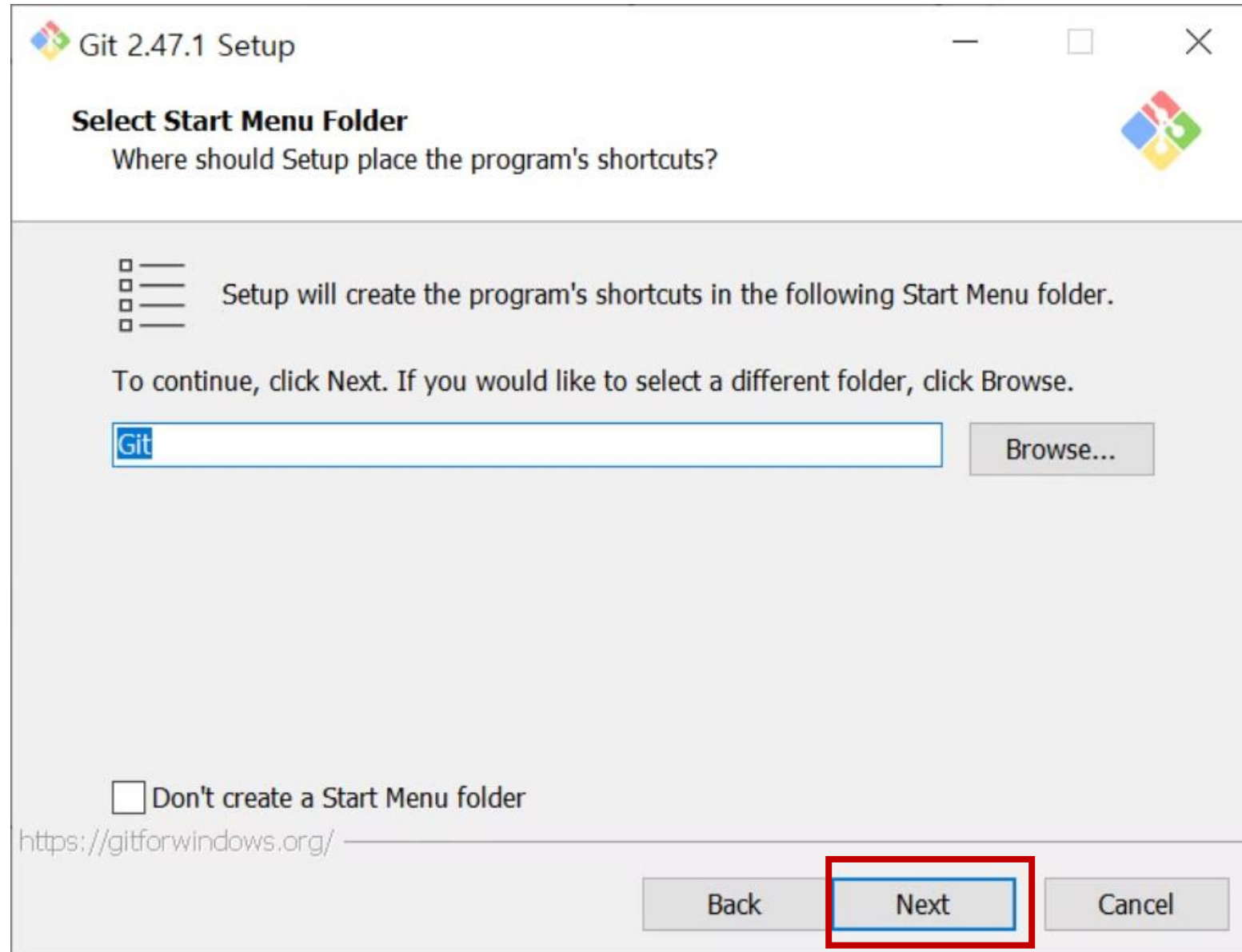
# Git 설치

기본 설정으로 설치합니다



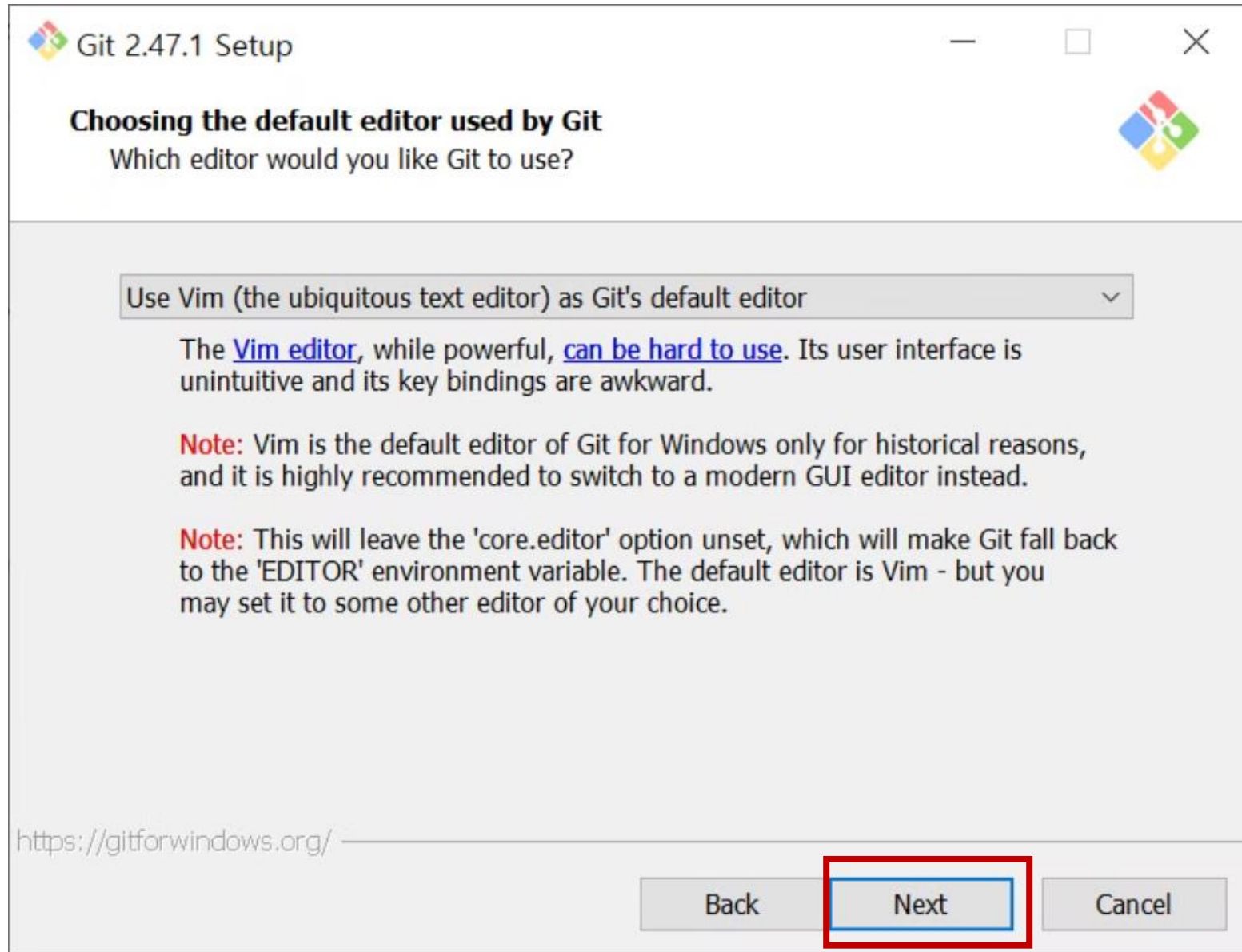
# Git 설치

기본 설정으로 설치합니다



# Git 설치

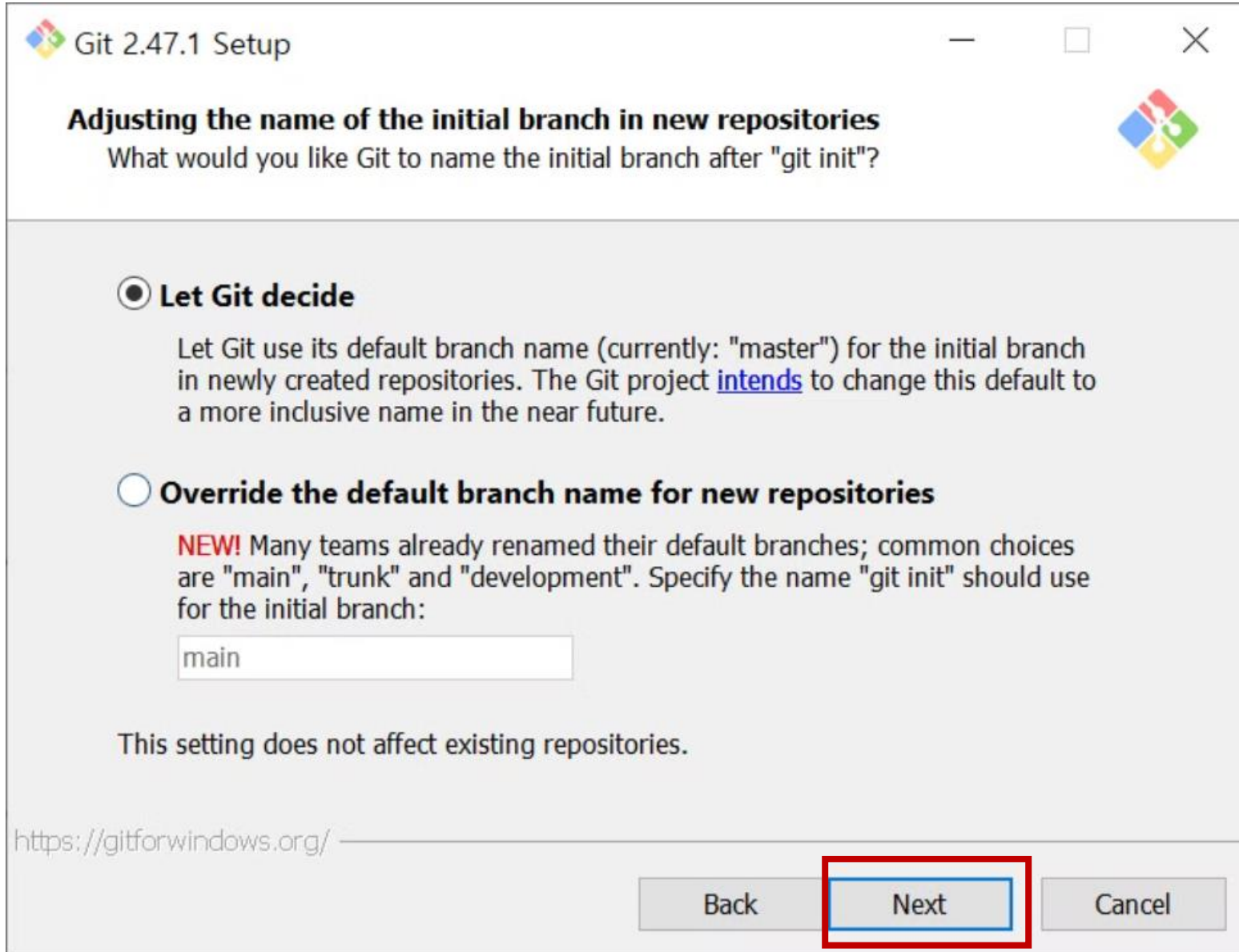
기본 설정으로 설치합니다





# Git 설치

기본 설정으로 설치합니다

A screenshot of the Git 2.47.1 Setup window. The window title is "Git 2.47.1 Setup". The main heading is "Adjusting the name of the initial branch in new repositories" with a subtitle "What would you like Git to name the initial branch after 'git init'?". There are two radio button options. The first option, "Let Git decide", is selected. The second option, "Override the default branch name for new repositories", is unselected. Below the second option, there is a text box containing the word "main". At the bottom, there are three buttons: "Back", "Next", and "Cancel". The "Next" button is highlighted with a red rectangle. The window also includes a URL "https://gitforwindows.org/" and a Git logo in the top right corner.

Git 2.47.1 Setup

**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:

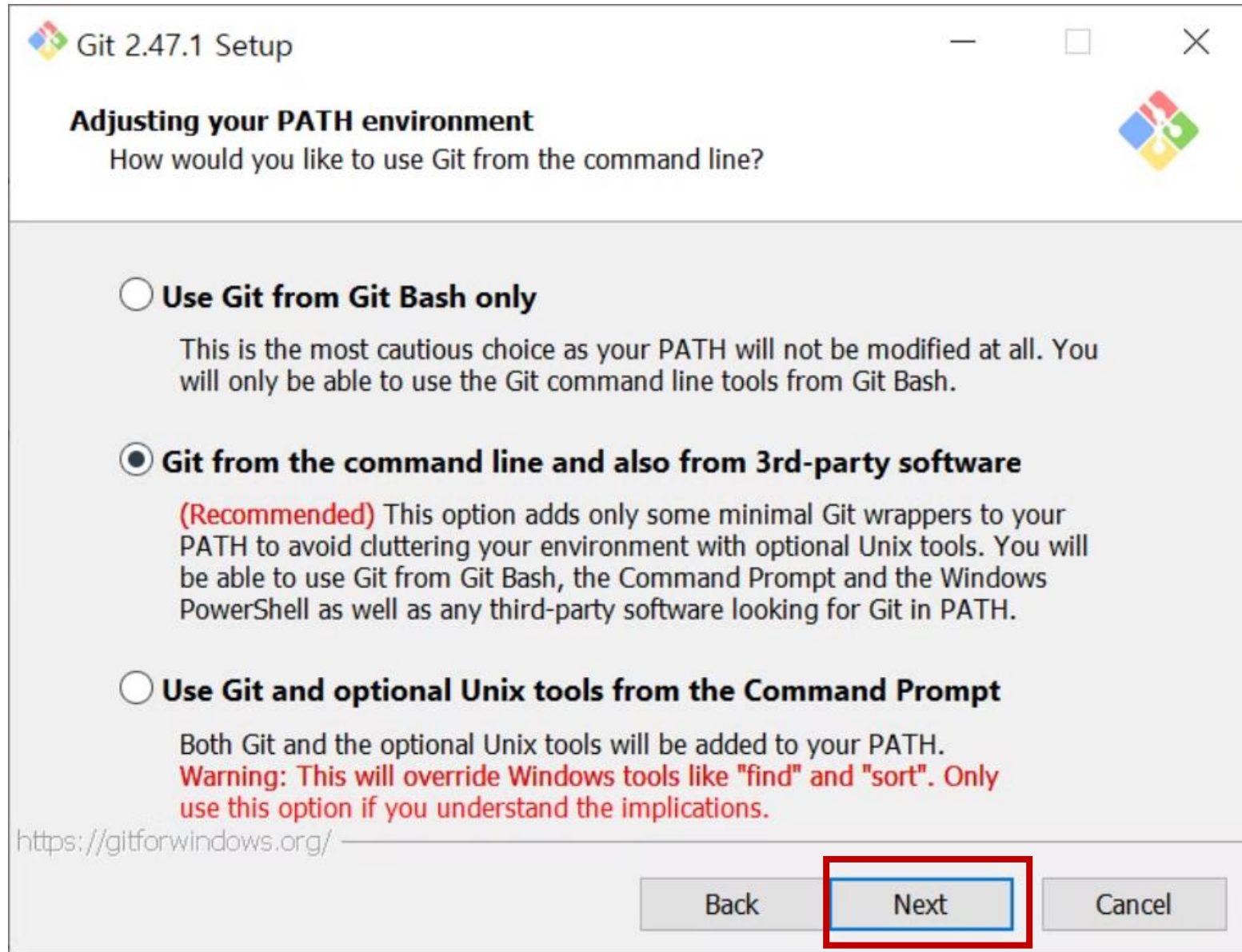
This setting does not affect existing repositories.

<https://gitforwindows.org/>

Back Next Cancel

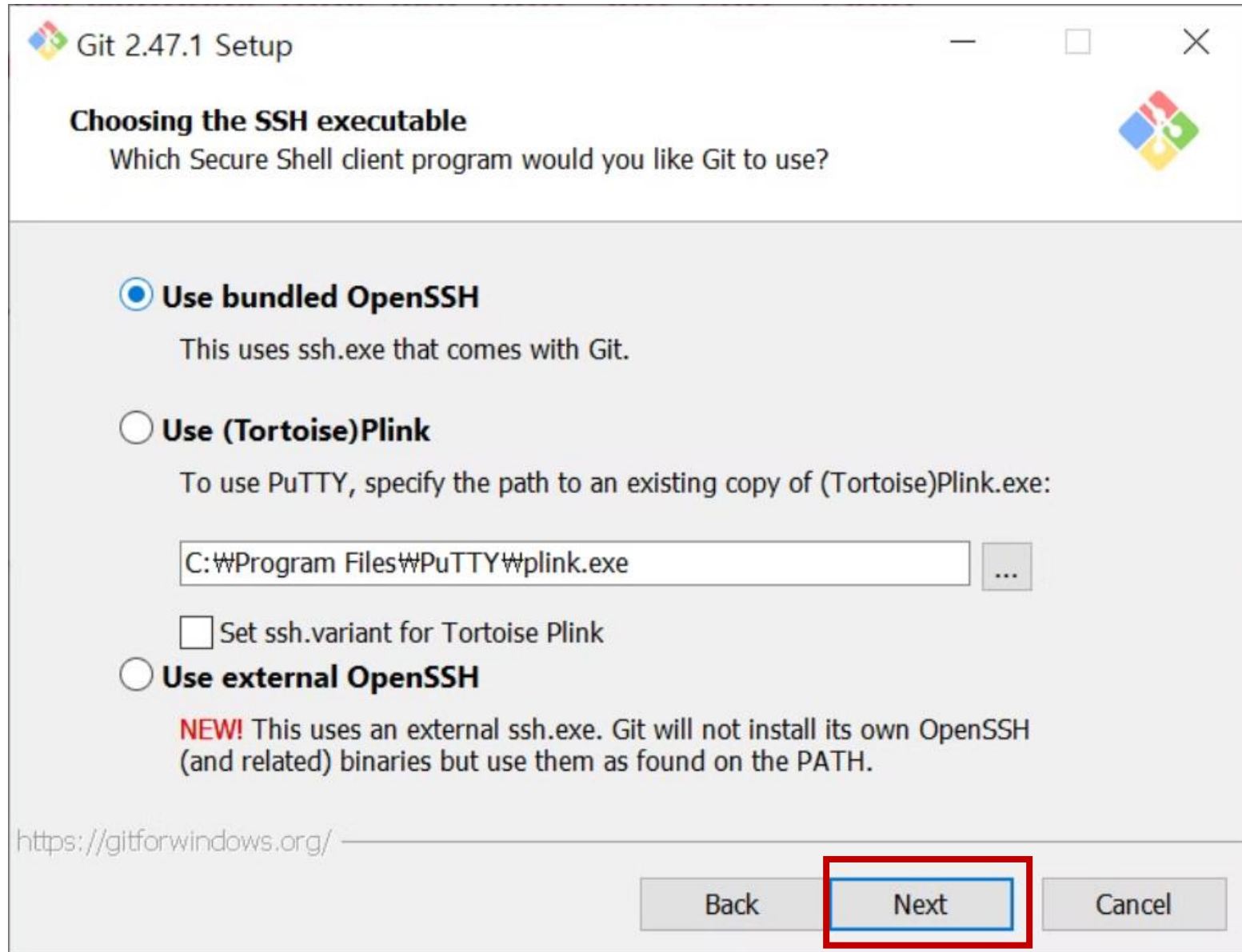
# Git 설치

기본 설정으로 설치합니다



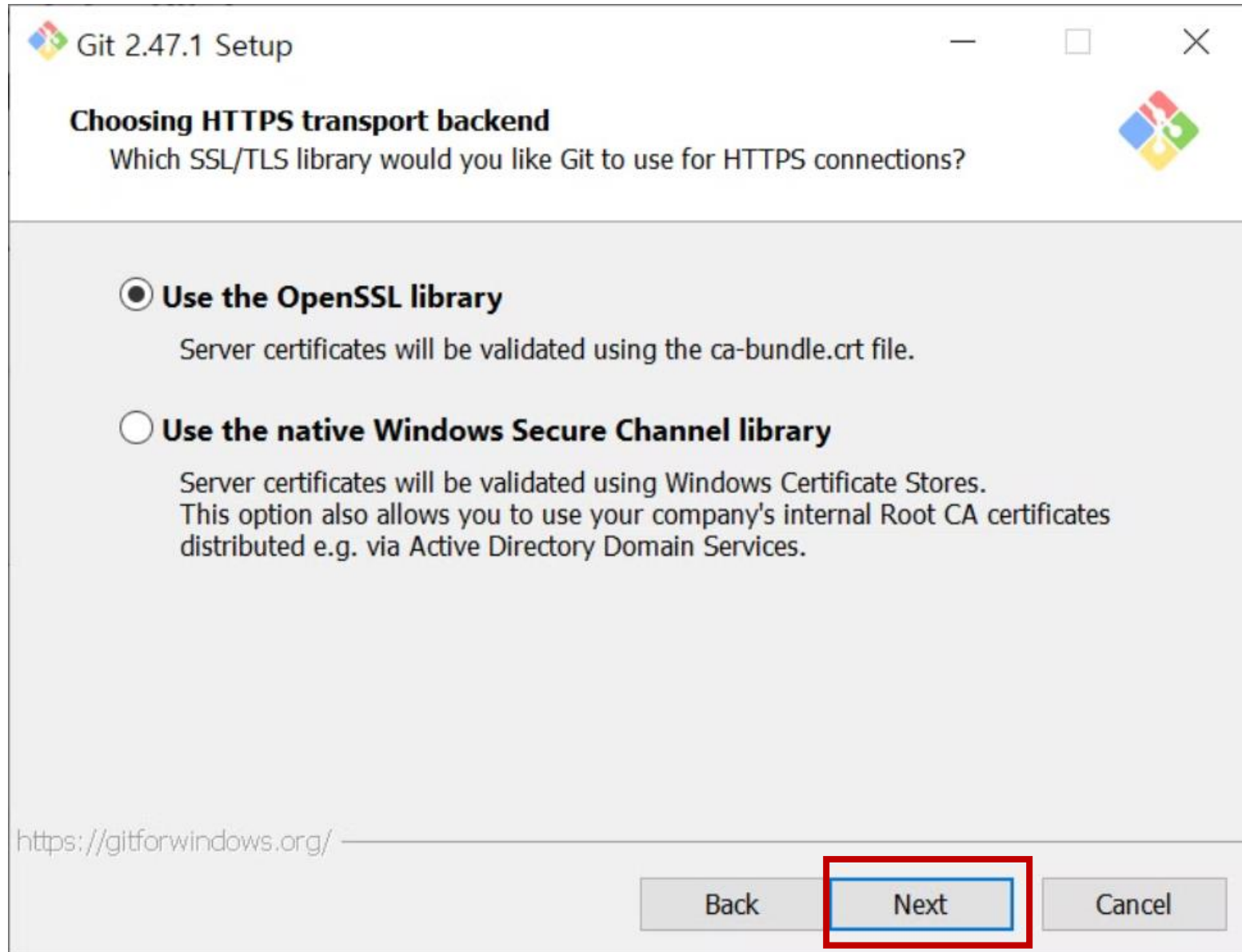
# Git 설치

기본 설정으로 설치합니다



# Git 설치

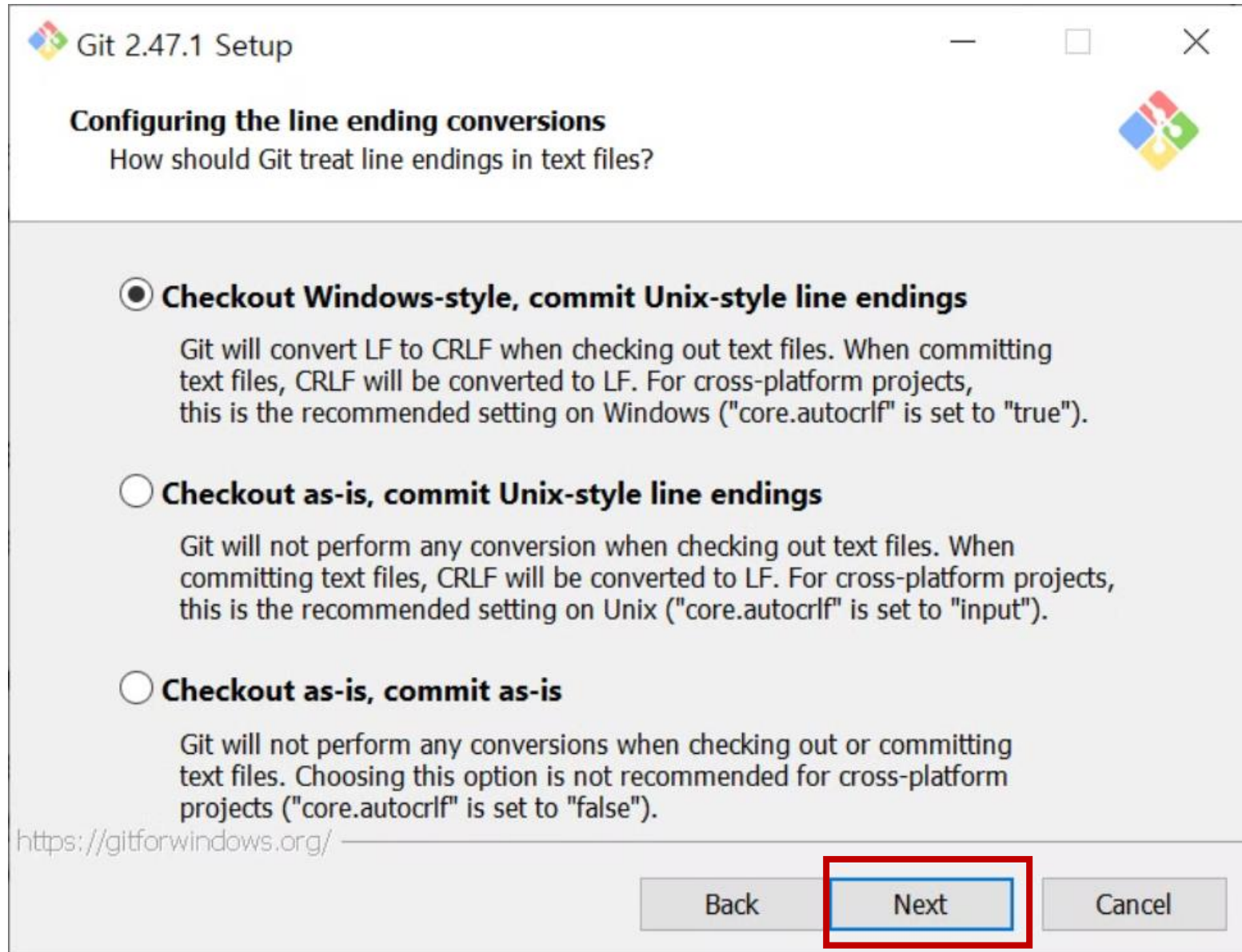
기본 설정으로 설치합니다





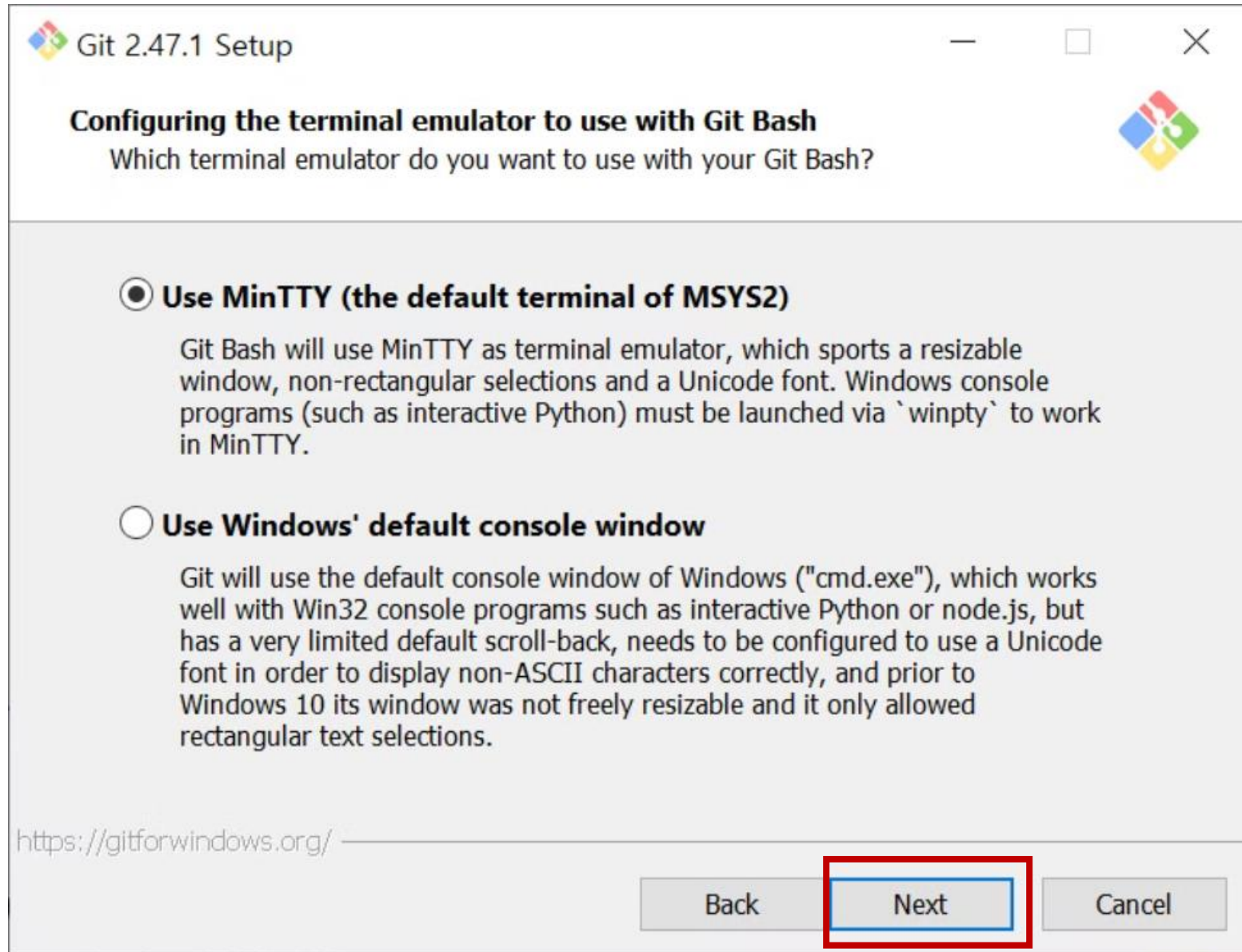
# Git 설치

기본 설정으로 설치합니다



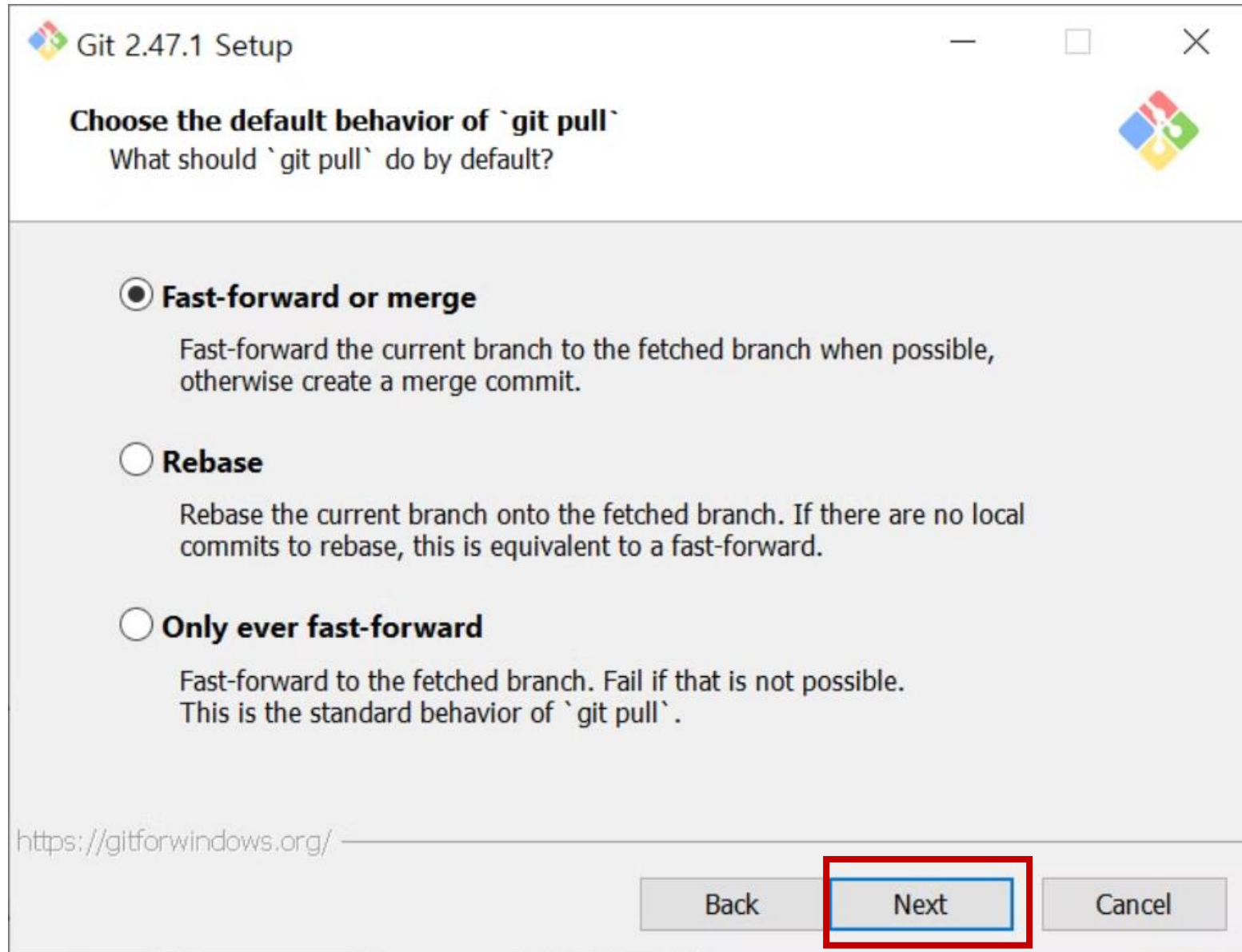
# Git 설치

기본 설정으로 설치합니다



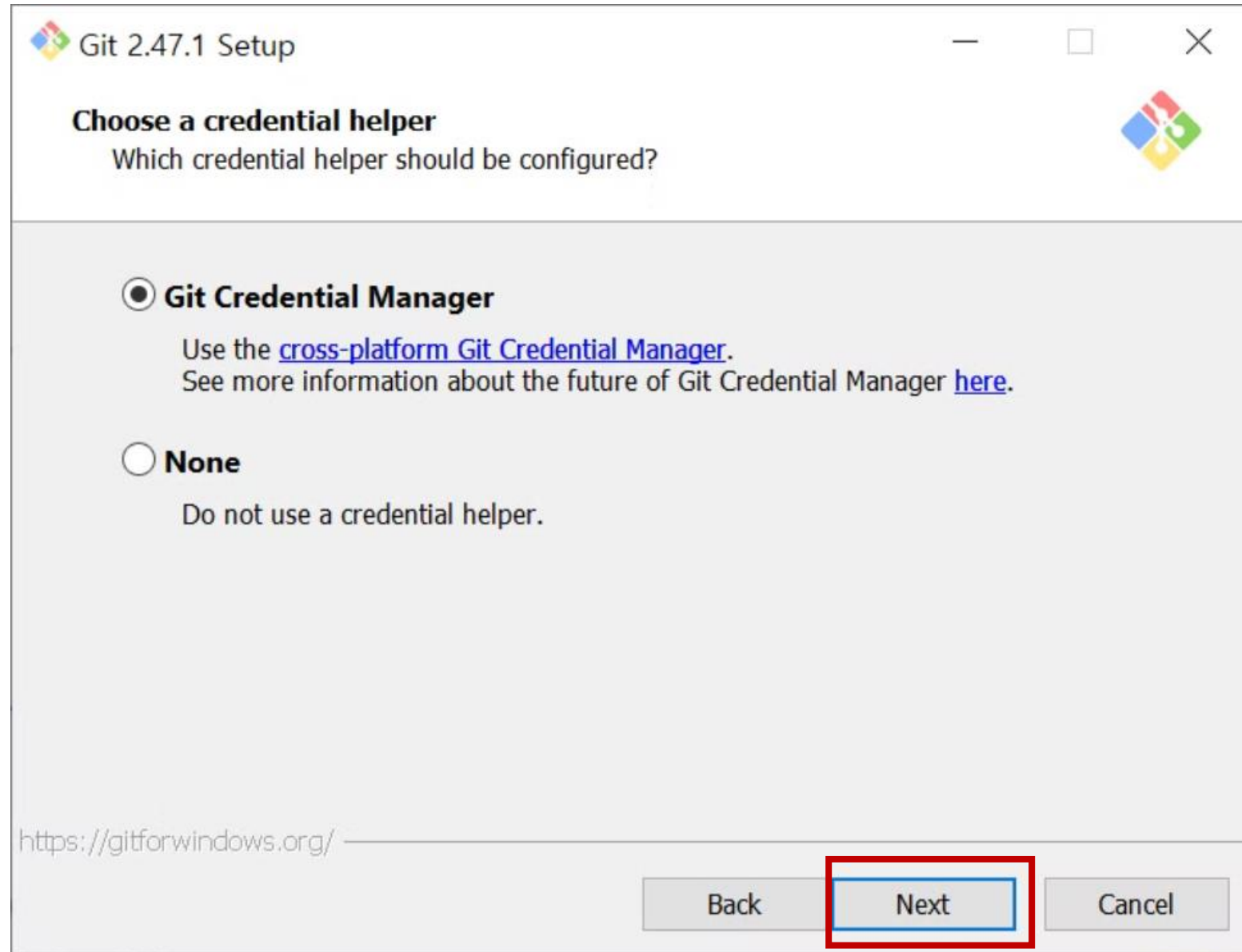
# Git 설치

기본 설정으로 설치합니다



# Git 설치

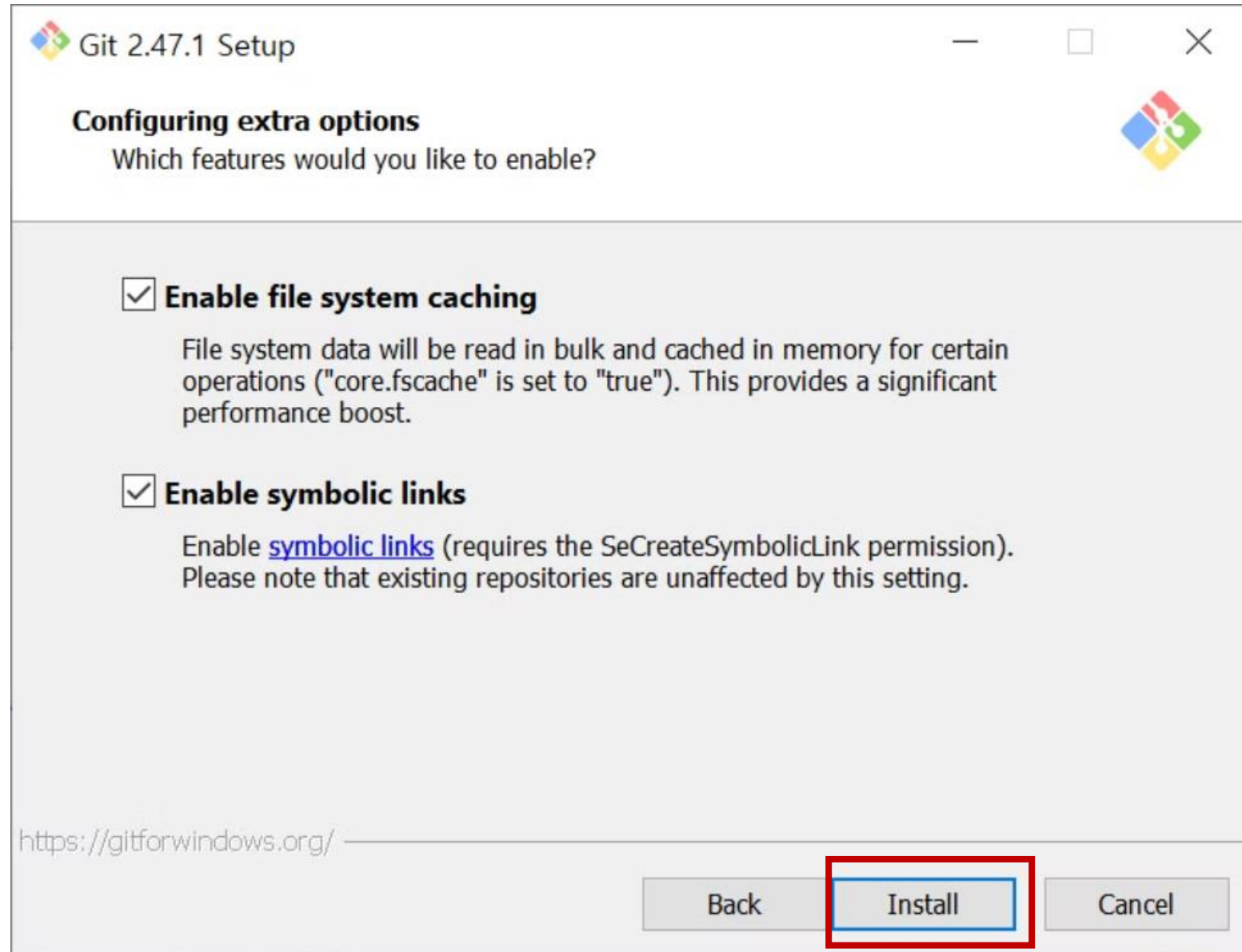
기본 설정으로 설치합니다





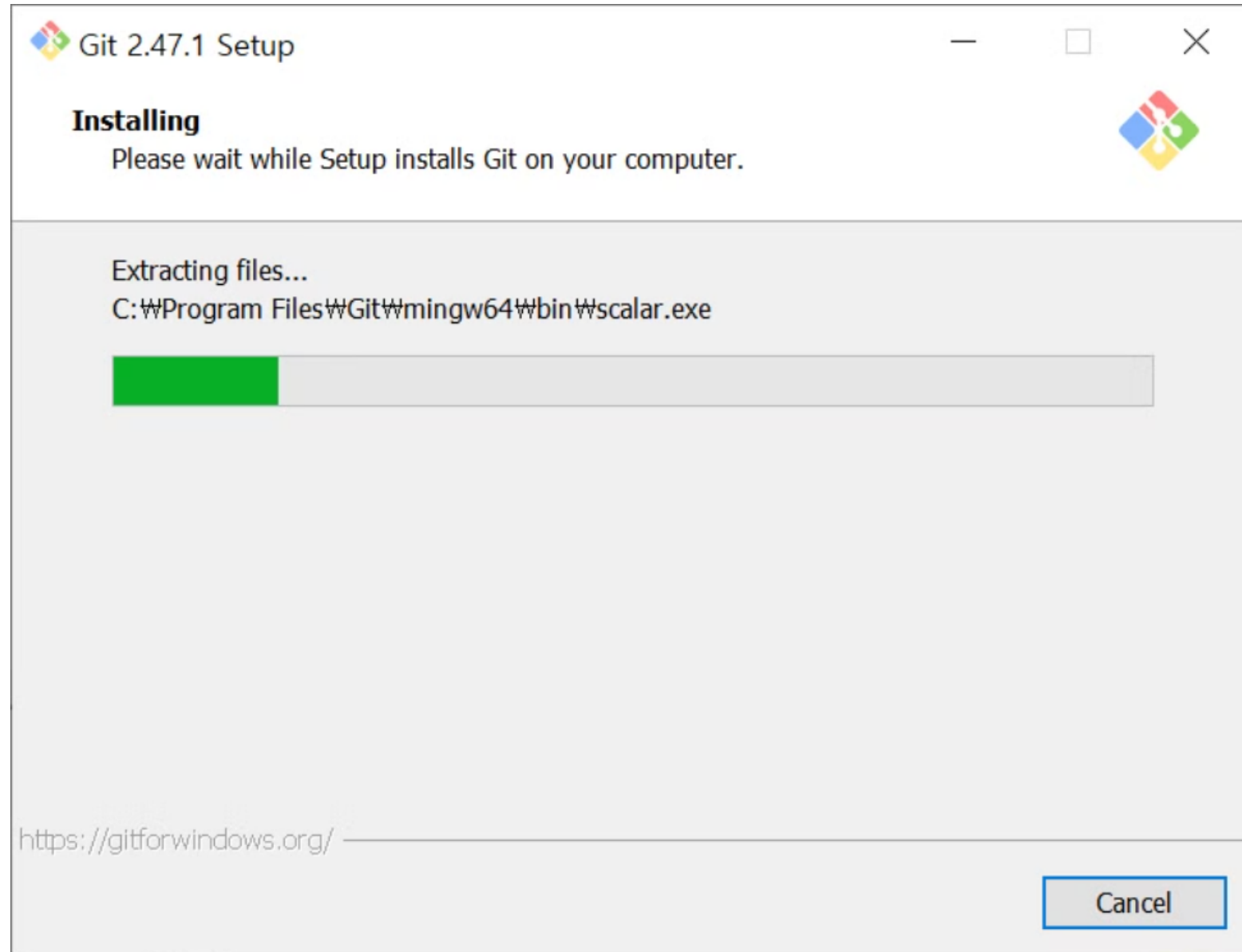
# Git 설치

기본 설정으로 설치합니다



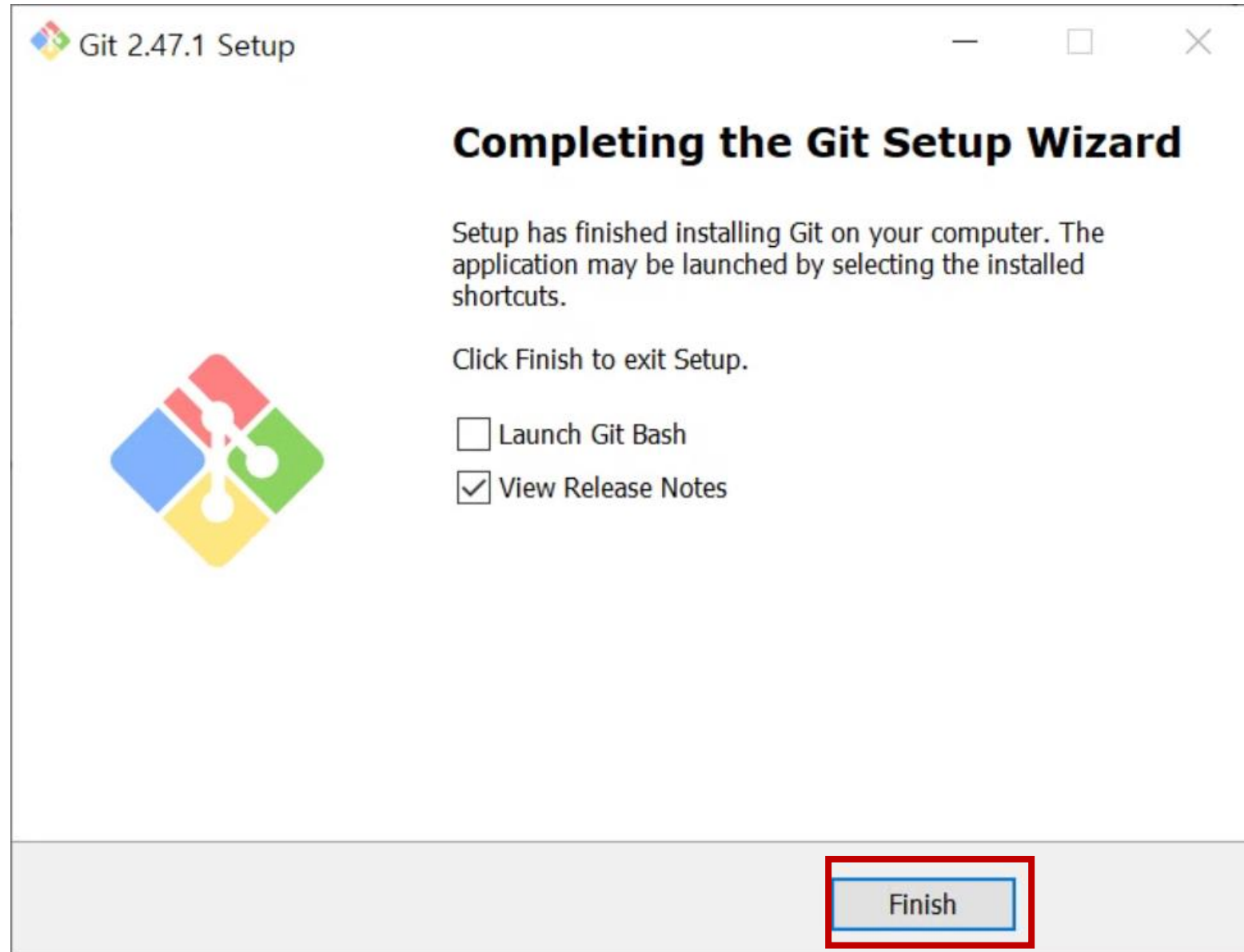
# Git 설치

기본 설정으로 설치합니다



# Git 설치

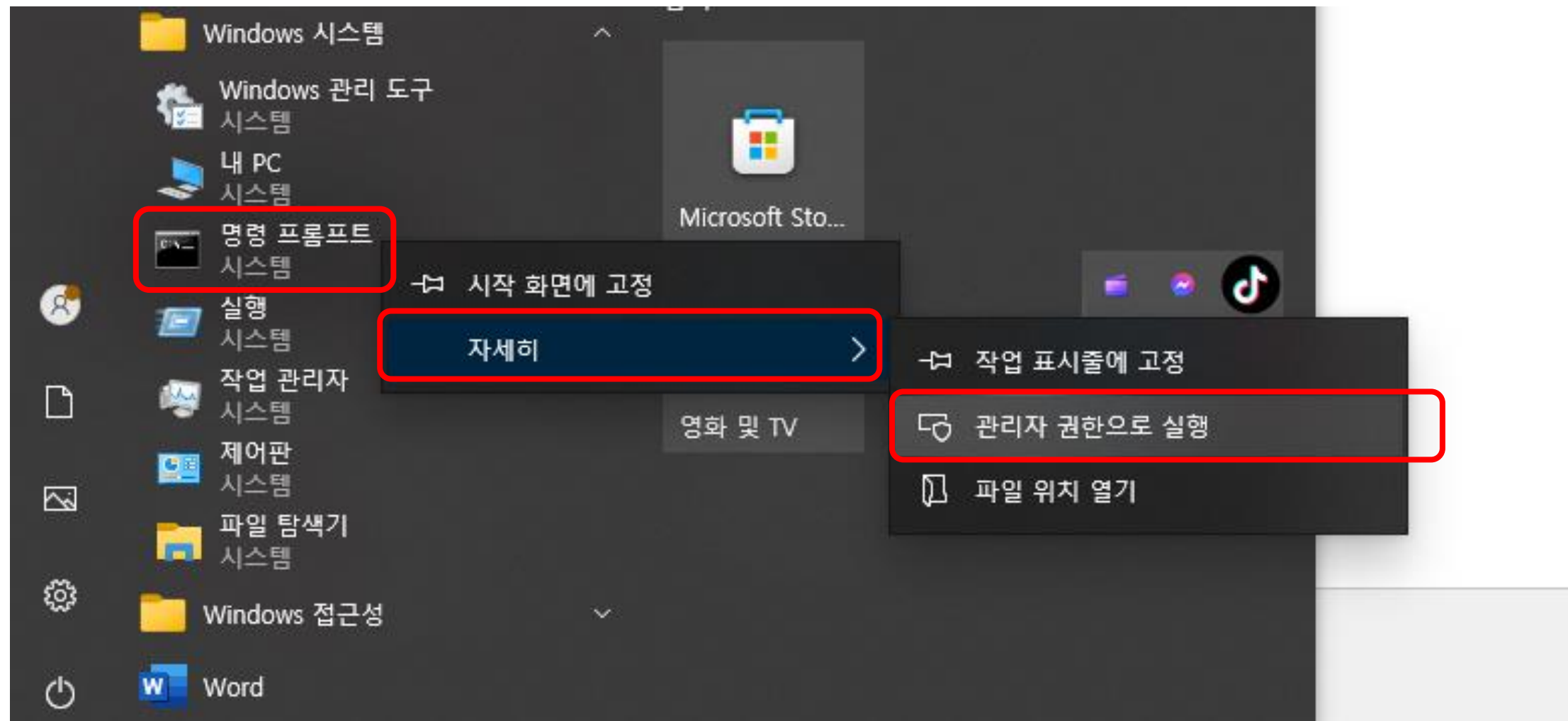
기본 설정으로 설치합니다



### 3. 영화 정보 수집 프로젝트 다운로드

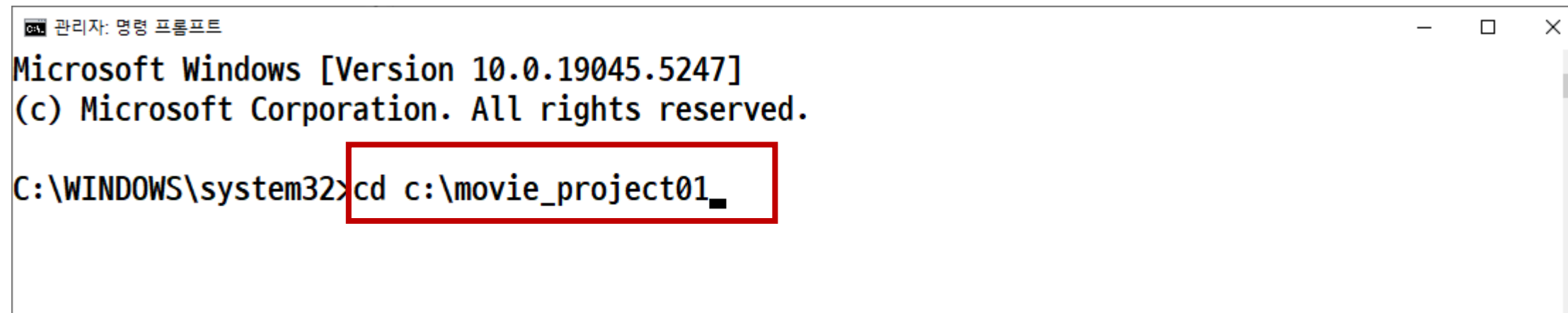
영화 정보 수집 프로젝트를 다운로드 합니다

# 영화 정보 수집 프로젝트 다운로드



## 영화 정보 수집 프로젝트 다운로드

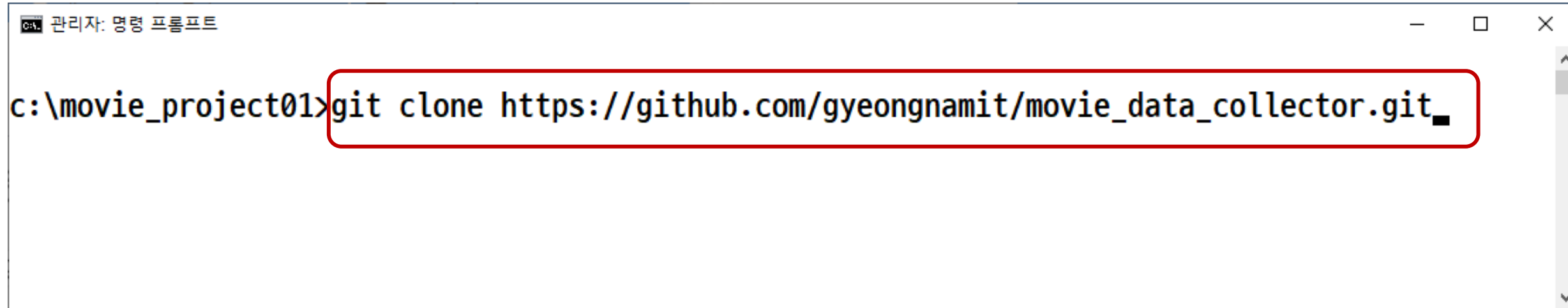
- 아래를 입력 합니다



A screenshot of a Windows Command Prompt window. The title bar reads "관리자: 명령 프롬프트" (Administrator: Command Prompt). The window content shows the following text: "Microsoft Windows [Version 10.0.19045.5247]" and "(c) Microsoft Corporation. All rights reserved." Below this, the command prompt shows "C:\WINDOWS\system32>". The command "cd c:\movie\_project01\_" is entered and highlighted with a red rectangular box. The cursor is at the end of the command.

# 영화 정보 수집 프로젝트 다운로드

## ■ 아래를 입력 합니다



```
c:\movie_project01>git clone https://github.com/gyeongnamit/movie_data_collector.git
```

## ■ 명령어의 의미

- git clone은 "이 프로젝트를 내 컴퓨터로 복사하겠다"는 명령입니다.뒤에 있는 주소 [https://github.com/gyeongnamit/movie\\_data\\_collector.git](https://github.com/gyeongnamit/movie_data_collector.git)는 복사하려는 프로젝트의 위치를 나타냅니다.

## ■ 결과

- 이 명령어를 실행하면 컴퓨터에 "movie\_data\_collector"라는 폴더가 생기고, 그 안에 프로젝트 파일들이 다운로드됩니다.