

# github 기초 실습



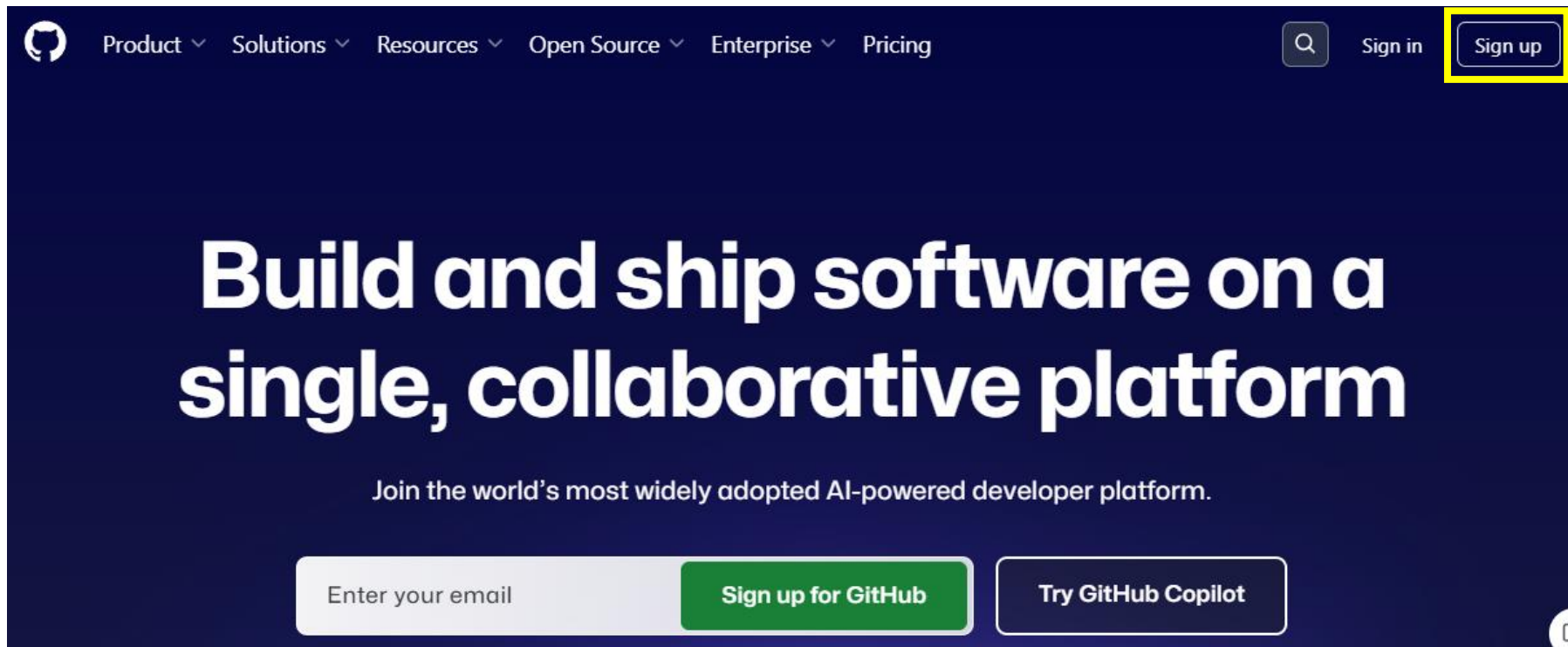
About..

컴퓨터소프트웨어공학과  
김 원 일



- git 서버

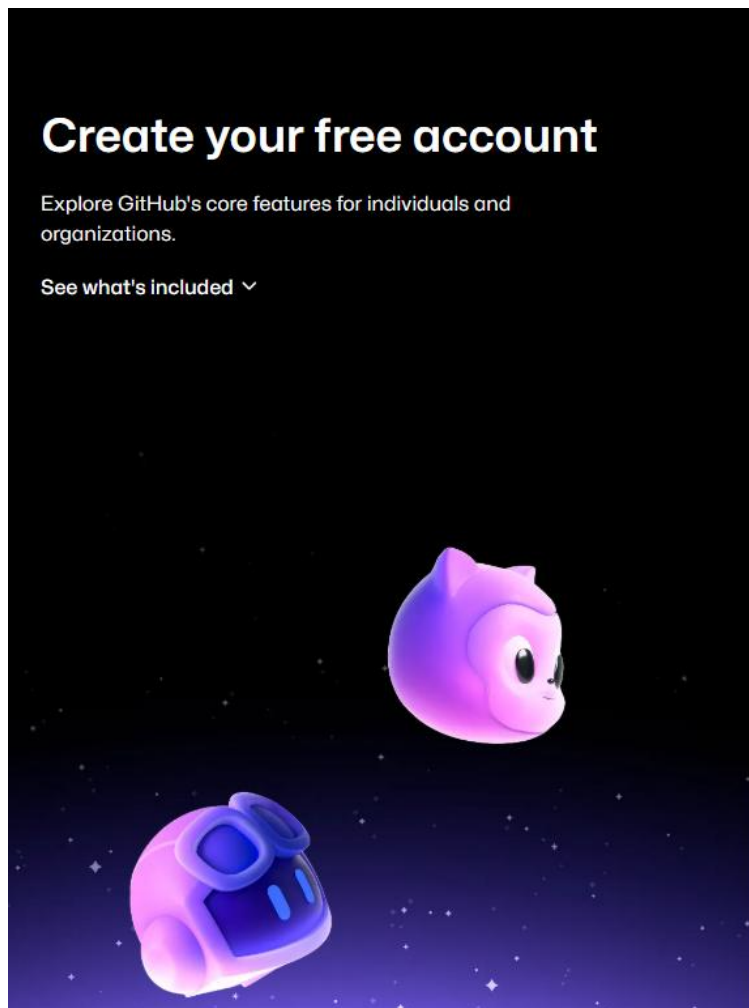
- 어디서나 접근 가능한 서비스를 제공하는 git 서버에 계정 생성
- 기본 서비스는 무료이며, 저장소 사용과 별도 서비스 사용 가능
- "Sign up"으로 새로운 계정 생성. 이미 있는 경우는 생성 필요 없음





# github 계정 생성 - 2

- 간단한 정보 입력 후 메일로 계정 인증



Already have an account? [Sign in](#) →

## Sign up to GitHub

Email\*

Password\*

Password should be at least 15 characters OR at least 8 characters including a number and a lowercase letter.

Username\*

Username may only contain alphanumeric characters or single hyphens, and cannot begin or end with a hyphen.

Your Country/Region\*

Korea, South



For compliance reasons, we're required to collect country information to send you occasional updates and announcements.

Continue >

By creating an account, you agree to the [Terms of Service](#). For more information about GitHub's privacy practices, see the [GitHub Privacy Statement](#). We'll occasionally send you account-related emails.



## 중간고사



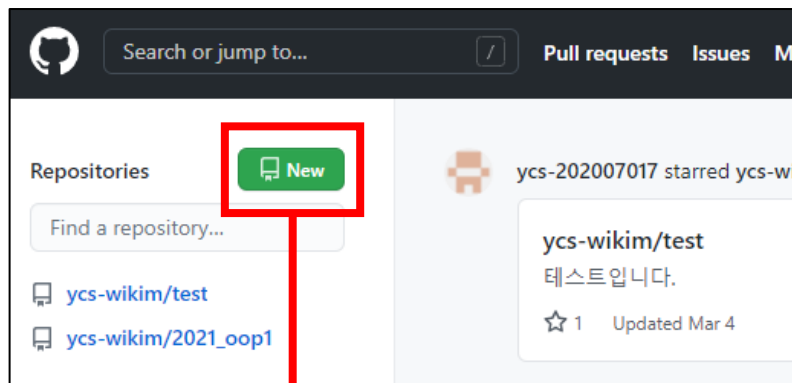
- **차주 목요일 11시에 중간고사 (OSS)**
- **범위 : 1장 ~ 대상 제외 → 총 13문제 ( + 5개 )**
  - vi 제외
  - 간단하게 명령어에 관련된 부분 : 단답식
  - 1번 문제 답 : VCS 또는 버전 관리 시스템
- **주의 사항**
  - 사람이 알아볼 수 있도록 글씨를 쓸 것.
  - 학번, 이름 정확하게 기재
  - 볼펜으로 할 것! - 시험지가 5년간 보관
  - 편지 작성 금지!
- **점수 : 30점 만점**



# 저장소 생성하기 - 1



- 메인 페이지  
- 저장소 생성



## Create a new repository

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

Required fields are marked with an asterisk (\*).

Owner \*      Repository name \*

ycs-wikim /

Great repository names are short and memorable. Need inspiration? How about [silver-adventure](#) ?

Description (optional)

- ☒ Public  
Anyone on the internet can see this repository. You choose who can commit.
- ☐ Private  
You choose who can see and commit to this repository.

### Initialize this repository with:

- ☐ Add a README file  
This is where you can write a long description for your project. [Learn more about READMEs.](#)

### Add .gitignore

.gitignore template: None ▾

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

### Choose a license

License: None ▾

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

You are creating a public repository in your personal account.

Create repository




## 저장소 생성하기 - 2



### • 저장소 이름

- 저장소는 항상 본인 계정 명 뒤에 입력한 이름으로 생성
- 저장소 이름은 로컬에서 디렉터리 이름으로 적용됨
- 이름 확인은 자신의 계정 내의 저장소와의 충돌만 확인
- 저장소에 대한 추가적인 설명도 입력이 가능
- 추가된 설명은 저장소 메인 페이지에 설명으로 나타남

Owner \*      Repository name \*

 ycs-wikim / 2021\_oop2 ✓

Great repository names are 2021\_oop2 is available. Need inspiration? How about [bookish-bassoon?](#)

Description (optional)





## 저장소 생성하기 - 2



### • 저장소 권한 설정

- 저장소 권한은 Public, Private 2가지만 설정 가능
- Public 권한 저장소
  - 저장소의 모든 코드와 내용을 누구나 접근하여 볼 수 있음
  - 누구나 볼 수 있지만 쓰기 권한은 존재하지 않고, 협업자(Collaborators)만 쓰기 가능
  - 무제한의 저장공간을 제공하며, 저장소당 500MB 정도의 공간을 제공
- Private 권한 저장소
  - 저장소의 모든 코드와 내용이 허용된 사용자들만 볼 수 있음
  - 허용된 사용자들에게 읽기/쓰기 권한을 별도로 설정할 수 있음
  - 유료였으나 무료로 변경되었으며, Public 권한 저장소와 동일한 공간을 제공
  - 테스트 또는 개인/팀만의 별도 코드 공유 용도로 사용

☒  **Public**  
Anyone on the internet can see this repository. You choose who can commit.

☐  **Private**  
You choose who can see and commit to this repository.



## 저장소 생성하기 - 3



- “Add a README file”
  - 저장소 설명 파일 추가 여부 설정
  - 추가로 설정하면 저장소 생성시에 입력한 설명 내용이 입력
  - 저장소의 전체적인 설명이나 내용 등으로 내용 변경이 가능
  - 간단한 마크다운(Markdown) 언어로 수정
  - 저장소에는 “README.md”로 생성

Initialize this repository with:

☐ Add a README file  
This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore

.gitignore template: None ▾

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

License: None ▾

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

You are creating a public repository in your personal account.

Create repository

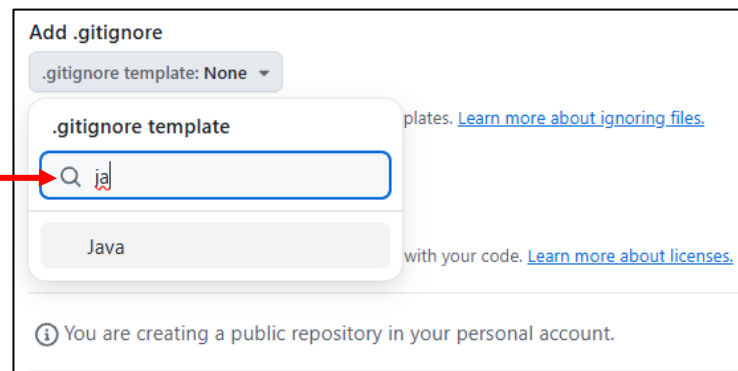
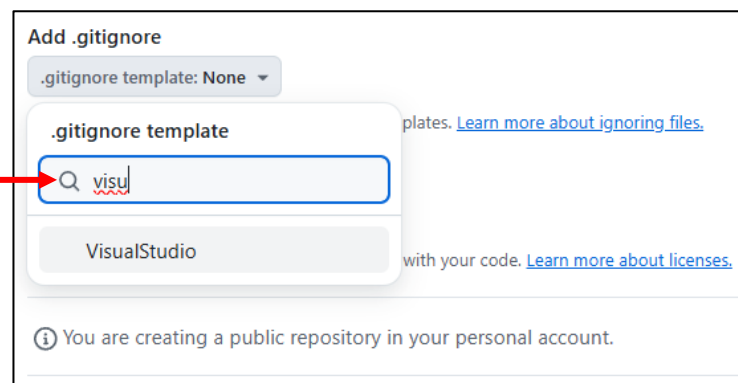
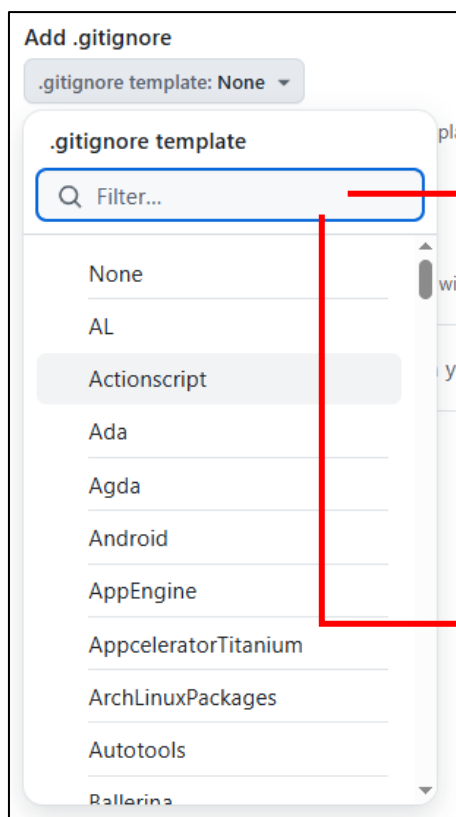




## 저장소 생성하기 - 4

- "Add .gitignore"

- 대상 제외 파일을 초기 저장소에 추가할 것인지 여부 선택
- 언어별로 미리 준비된 .gitignore 파일 템플릿이 다수 존재
- 필요한 대상 제외 언어나 도구를 선택하면 자동으로 적용





## 저장소 생성하기 - 5



- “Choose a license”
  - 저장소에 보관된 소스 코드와 문서들에 대한 라이선스 설정
  - 라이선스에 따라 보관된 코드에 대한 권한이 결정
  - 보통 “GNU General Public License v.3.0” 정도를 선택하고 생성

Choose a license

License: None ▾

License

None

Apache License 2.0

GNU General Public License v3.0

MIT License

BSD 2-Clause "Simplified" License

BSD 3-Clause "New" or "Revised" License

Boost Software License 1.0

Creative Commons Zero v1.0 Universal

Initialize this repository with:

☒ Add a README file  
This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore  

.gitignore template: VisualStudio ▾

  
Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license  

License: GNU General Public License v3.0 ▾

  
A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

This will set main as the default branch. Change the default name in your [settings](#).

You are creating a public repository in your personal account.

Create repository



# 저장소 생성하기 - 6

- 생성된 저장소의 초기 화면
  - 설정 내용이 적용되어 있는지 확인

The screenshot shows the GitHub interface for a repository named 'testres' under the user 'ygs-wikim'. The repository is public and has 1 branch and 0 tags. The file list shows three files: '.gitignore', 'LICENSE', and 'README.md', all from the 'Initial commit' at 'now'. The README file is selected and shows the title 'testres' and the description '테스트를 위한 저장소.' (Repository for testing). The right sidebar contains a 'Monica' chatbot, a '저장소 요약' (Repository Summary) section with a button '이 저장소 요약' (Summary of this repository), and an 'About' section with details like '테스트를 위한 저장소.', 'Readme', 'GPL-3.0 license', 'Activity', '0 stars', '1 watching', and '0 forks'.

Repository: ygs-wikim / testres

Files:

File	Commit	Time
.gitignore	Initial commit	now
LICENSE	Initial commit	now
README.md	Initial commit	now

Selected File: README

Repository Name: testres

Description: 테스트를 위한 저장소.

Repository Summary:

가장 진보된 모델을 지원하여 저장소 내용을 빠르게 이해할 수 있도록 도와줍니다

이 저장소 요약

About:

테스트를 위한 저장소.

Readme

GPL-3.0 license

Activity

0 stars

1 watching

0 forks



# 저장소 삭제



## • 저장소 삭제하기

- 저장소 상단의 "Settings" 메뉴 클릭
- 설정 화면 가장 아래 "Danger Zone"에서 삭제 가능

[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#)

### Danger Zone

Change repository visibility  
This repository is currently public.

Change visibility

Disable branch protection rules  
Disable branch protection rules enforcement and APIs

Disable branch protection rules

Transfer ownership  
Transfer this repository to another user or to an organization where you have the ability to create repositories.

Transfer

Archive this repository  
Mark this repository as archived and read-only.

Archive this repository

Delete this repository  
Once you delete a repository, there is no going back. Please be certain.

Delete this repository



# 저장소 로컬로 가져오기 - 1



## • 서버 내용 복제

- 복제로 내려 받는 위치에 저장소 이름의 디렉터리가 생성
- 서버 내용을 복제하기 위한 디렉터리를 생성. git init으로 초기화하지 않음!
- "git clone" 명령은 원격지 서버의 내용을 복제하라는 명령
- 형식) git clone https://github.com/계정명/저장소명(.git)
- 계정명이 **ycs-wikim**이고, 저장소명이 **testres** 이라면
- git clone https://github.com/ycs-wikim/testres(.git) 으로 복제

```
MINGW64:/f/202507001

unange1@DESKTOP-I80QG85 MINGW64 /f/202507001
$ git clone https://github.com/ycs-wikim/testres
Cloning into 'testres'...
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 5 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (5/5), 15.73 KiB | 700.00 KiB/s, done.

unange1@DESKTOP-I80QG85 MINGW64 /f/202507001
$ |
```



## 저장소 로컬로 가져오기 - 2



### • 저장소 확인

- 생성된 디렉터리로 이동하여 파일 내용 확인
- 복제를 통해 내려 받은 디렉터리는 자동으로 저장소로 설정
- 저장소 생성 시점에 만든 파일들이 github 사이트와 동일하게 생성됨을 확인
- `".git/"` 디렉터리가 자동으로 생성되어 있음을 확인할 수 있음

```
MINGW64:/f/202507001/testres

unangel@DESKTOP-I8OQG85 MINGW64 /f/202507001
$ cd testres/

unangel@DESKTOP-I8OQG85 MINGW64 /f/202507001/testres (main)
$ ls -al
total 49
drwxr-xr-x 1 unangel 197121      0 Apr 16 20:57 ./
drwxr-xr-x 1 unangel 197121      0 Apr 16 20:57 ../
drwxr-xr-x 1 unangel 197121      0 Apr 16 20:57 .git/
-rw-r--r-- 1 unangel 197121  7360 Apr 16 20:57 .gitignore
-rw-r--r-- 1 unangel 197121 35823 Apr 16 20:57 LICENSE
-rw-r--r-- 1 unangel 197121    43 Apr 16 20:57 README.md

unangel@DESKTOP-I8OQG85 MINGW64 /f/202507001/testres (main)
$
```



## 저장소 로컬로 가져오기 - 3



### • 저장소의 상태 확인

- 원격 서버의 존재 유/무에 따라 출력되는 정보가 다름
- 서버가 없는 경우 브랜치 정보와 커밋에 대한 정보만 출력
- 서버가 있으면 "origin/main"의 최신 정보로 업데이트 되었음을 출력
- 로컬/원격 저장소가 정보 교환한 정보를 저장소 상태에서 확인

```
MINGW64:/f/suho
unangel@DESKTOP-I8OQG85 MINGW64 /f/suho (master)
$ git status
On branch master
nothing to commit, working tree clean

unangel@DESKTOP-I8OQG85 MINGW64 /f/suho (master)
$
```

원격 서버가 없는 저장소

```
MINGW64:/f/202507001/testres
unangel@DESKTOP-I8OQG85 MINGW64 /f/202507001/testres (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean

unangel@DESKTOP-I8OQG85 MINGW64 /f/202507001/testres (main)
$
```

원격 서버가 있는 저장소



# 로컬 저장소 실습 - 1



## • 파일 추가 및 관리

- 새로운 파일을 추가하고, Commit하여 로컬 데이터베이스 정보 업데이트
- 간단하게 ls 명령어의 실행 결과를 저장하는 파일을 생성
- 해당 파일을 Staging 상태로 변경하고, commit 수행

```
MINGW64:/f/202507001/testres

unange1@DESKTOP-I8OQG85 MINGW64 /f/202507001/testres (main)
$ ls -al > list.txt

unange1@DESKTOP-I8OQG85 MINGW64 /f/202507001/testres (main)
$ git add list.txt
warning: in the working copy of 'list.txt', LF will be replaced by CRLF the next time Git touches it

unange1@DESKTOP-I8OQG85 MINGW64 /f/202507001/testres (main)
$ git commit
```

```
MINGW64:/f/202507001/testres

[main 1a223e8] 1. list.txt [ + ] list current directory info.
1 file changed, 8 insertions(+)
create mode 100644 list.txt

unange1@DESKTOP-I8OQG85 MINGW64 /f/202507001/testres (main)
$ |
```





### • 저장소 상태 확인

- 현재 브랜치에 대한 정보를 추가적으로 제공
- 원격지 서버 저장소에는 여전히 3개의 파일만 존재하고 있음
- 로컬 저장소에는 새롭게 파일이 추가되어 있음을 메시지로 출력
- 서버 저장소보다 1개의 commit이 더 많음을 알 수 있음
- 서버로 업로드하는 명령어인 "git push"에 대한 정보도 같이 출력

```
MINGW64:/f/202507001/testres
unange1@DESKTOP-I8OQG85 MINGW64 /f/202507001/testres (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean

unange1@DESKTOP-I8OQG85 MINGW64 /f/202507001/testres (main)
$
```



## 로컬 저장소 실습 - 3



### • 서버로 업로드

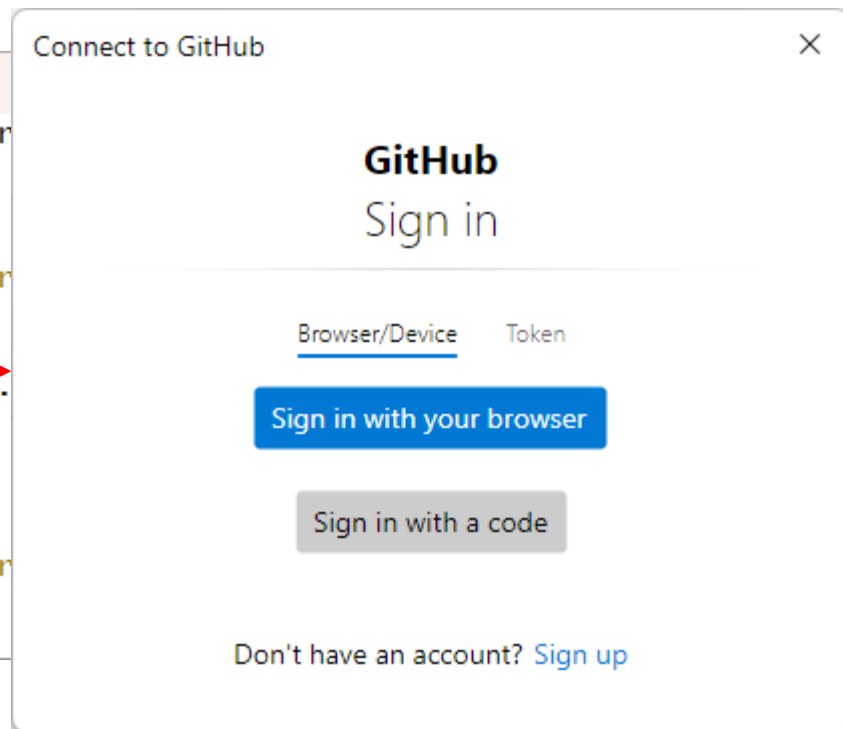
- "git push"는 별도의 옵션 없이 원격지로 로컬 정보를 전송하는 명령어
- 현재 commit되어 있는 정보를 서버로 전달하여 동기화를 수행
- 서버 접속/연결을 위한 별도의 창이 나타나는 것을 확인

```
MINGW64:/f/202507001/testres
[main 1a223e8] 1. list.txt [ + ] list current dir
1 file changed, 8 insertions(+)
create mode 100644 list.txt

unange1@DESKTOP-I8OQG85 MINGW64 /f/202507001/testres
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
(use "git push" to publish your local commits)

nothing to commit, working tree clean

unange1@DESKTOP-I8OQG85 MINGW64 /f/202507001/testres
$ git push
```





- github 인증

- 인증을 위한 2가지 방법 제공. ID/PWD 기반 인증은 수행했다가 폐기(보안)
- 웹 브라우저로 인증은 웹 브라우저에서 로그인으로 인증을 수행
- 개인 접근 토큰(징표, 형식물)은 서버와 통신할 때 입력 정보로 자동 인증 수행

Connect to GitHub

GitHub

Sign in

Browser/Device

Token

Sign in with your browser

Sign in with a code

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

Connect to GitHub

GitHub

Sign in

Browser/Device

Token

Personal access token

Sign in

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



# github 인증 토큰 생성 - 1

## • 화면 상단의 가장 오른쪽 아이콘 클릭

- "Settings" → "Developer settings" → "Personal access tokens" → Tokens(classic) → "Generate new token" 순으로 선택

The screenshot illustrates the steps to generate a classic personal access token on GitHub:

- Top Bar:** The GitHub icon (green octocat) is highlighted with a red box and a red arrow pointing to the 'Developer settings' link in the 'Settings' dropdown.
- Settings Dropdown:** The 'Developer settings' link is highlighted with a red box and a red arrow pointing to the 'Personal access tokens' link in the left sidebar.
- Left Sidebar:** The 'Personal access tokens' link is highlighted with a red box and a red arrow pointing to the 'Tokens (classic)' link in the main content area.
- Main Content Area:** The 'Tokens (classic)' link is highlighted with a red box and a red arrow pointing to the 'Generate new token (classic)' button in the 'Generate new token' section.




# github 인증 토큰 생성 - 2




## • 추가 인증 요청 및 생성 설정

- 저장소 접근을 위해서 "repo" 범위를 반드시 선택해야 함
- 페이지 가장 아래의 "Generate Token"을 선택하여 토큰을 생성



Confirm access

 Signed in as @ycs-wikim

Password

[Forgot password?](#)

Confirm

Tip: You are entering [sudo mode](#). After you've performed a sudo-protected action, you'll only be asked to re-authenticate again after a few hours of inactivity.

New personal access token (classic)

Personal access tokens (classic) function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

Note

What's this token for?

Expiration

The token will expire on the selected date

Select scopes

Scopes define the access for personal tokens. [Read more about OAuth scopes](#).

☒ repo 저장소 접근 권한 선택 : 반드시 체크

☐ repo.status Access commit status

☐ repo\_deployment Access deployment status

☐ public\_repo Access public repositories

☐ repo:invite Access repository invitations

☐ security\_events Read and write security events

☐ workflow Update GitHub Action workflows



# github 인증 토큰 생성 - 3

## • 토큰 정보 복사

- 블록으로 선택할 경우 특수 문자가 입력될 가능성이 있음
- 실제 토큰 정보 옆에 복사 버튼이 있으므로 버튼으로 복사
- 복사한 상태로 인증을 요청한 창에 붙여 넣고, 인증을 수행

The screenshot shows the 'Personal access tokens (classic)' page on GitHub. A modal window titled 'Connect to GitHub' is open, showing the 'Token' tab for signing in. Red annotations highlight the workflow: a red box around the copy icon in the token list, a red arrow pointing to a 'Copied!' notification, and another red box around the 'Sign in' button in the modal.

Personal access tokens (classic)

Tokens you have generated that can be used to access the [GitHub API](#).

Make sure to copy your personal access token now. You won't be able to see it again!

Token	Expiration	Last used
ghp_YM5eU4yWfQ0M	7202호 키 — repo	Last used within the last week
7202호실 — repo	Expires on Mon, Jun 9 2025.	
2024년 5406호 2반 — repo	This token has no expiration date.	Never used
2024년 5406호 — repo	This token has no expiration date.	Last used within the last 10 months

Connect to GitHub

GitHub Sign in

Browser/Device Token

Sign in

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



## • 서버 전송 완료

- 정상적으로 모든 절차를 수행하면 다음과 같이 전송이 수행
- 로컬에서 전송될 정보의 크기와 전송 속도 등의 정보가 출력
- 원격 저장소 경로 정보가 표시되며, 해시와 브랜치 정보가 출력
- 원격으로 연결된 저장소는 서버와 정보 차이가 발생할 수 있으므로 주의
- 추가한 commit은 반드시 서버로 전송하여 관리하는 것이 좋음

```
MINGW64:/f/202507001/testres

unangel@DESKTOP-I8OQG85 MINGW64 /f/202507001/testres (main)
$ git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 16 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 433 bytes | 433.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/yys-wikim/testres
   be18584..1a223e8  main -> main

unangel@DESKTOP-I8OQG85 MINGW64 /f/202507001/testres (main)
$ |
```



## 로컬 저장소 실습 - 6



### • github 페이지 확인

- 추가한 파일과 commit 시에 입력한 메시지를 확인할 수 있음
- list.txt 클릭 시, 탐색기처럼 파일과 디렉터리 내용을 확인하는 페이지로 이동
- 메시지 클릭 시, 해당 commit에서 발생한 내용을 확인하는 페이지로 이동

testres Public

main 1 Branch 0 Tags

Go to file t Add file <> Code

ygs-wikim 1. list.txt 1a223e8 · 42 minutes ago 2 Commits

.gitignore	Initial commit	8 hours ago
LICENSE	Initial commit	8 hours ago
README.md	Initial commit	8 hours ago
list.txt	1. list.txt [ + ] list current directory info.	42 minutes ago

README GPL-3.0 license





# 로컬 저장소 실습 - 7

## • 현재 저장소 정보

- "<>Code" 화면에서 파일 이름을 클릭한 경우에 출력되는 페이지
- 현재 저장소에 존재하는 파일과 디렉토리를 탐색기 형식으로 보기 가능
- 선택한 파일의 이력(history) 정보를 확인할 수 있음
- 파일을 다운로드 받거나 복사할 수 있는 메뉴 사용 가능
- 직접 온라인 웹페이지에서 수정할 수 있는 기능 제공
- 코드와 수정 내역을 확인할 수 있는 기능 제공

The screenshot displays a web-based file explorer interface. On the left, a sidebar shows a file tree with 'main' as the selected branch. The main area shows the contents of 'list.txt' in a 'Code' view. The file is 389 bytes and contains 8 lines of text. The interface includes a search bar, a file list, and a code editor. Several elements are highlighted with red boxes: the 'Code' tab, the 'Blame' tab, the 'Raw' button, the copy button, the download button, the edit button, and the 'History' button. The file content is as follows:

```
1 total 49
2 drwxr-xr-x 1 unangel 197121 0 Apr 16 21:16 ./
3 drwxr-xr-x 1 unangel 197121 0 Apr 16 20:57 ../
4 drwxr-xr-x 1 unangel 197121 0 Apr 16 21:15 .git/
5 -rw-r--r-- 1 unangel 197121 7360 Apr 16 20:57 .gitignore
6 -rw-r--r-- 1 unangel 197121 35823 Apr 16 20:57 LICENSE
7 -rw-r--r-- 1 unangel 197121 43 Apr 16 20:57 README.md
8 -rw-r--r-- 1 unangel 197121 0 Apr 16 21:16 list.txt
```



## • 온라인 파일 수정

- 웹에서 직접 파일을 수정할 수 있으며, 수정 내용을 즉시 확인 가능
- 파일이 수정되므로 commit과 수정 내역 취소도 수행할 수 있도록 지원

The screenshot illustrates the process of editing a file online. It is divided into two panels. The top panel shows the initial state where the 'Edit' button is selected. The bottom panel shows the state after clicking 'Commit changes...', where the 'Preview' button is selected and a confirmation message 'file changed....' is displayed. Red boxes and arrows highlight the key UI elements and the flow of the process.

**Top Panel (Edit Mode):**

- Buttons: **Edit** (highlighted), Preview, Cancel changes, **Commit changes...** (highlighted).
- File Content (list.txt):

```
1 total 49
2 drwxr-xr-x 1 unangel 197121  0 Apr 16 21:16 ./
3 drwxr-xr-x 1 unangel 197121  0 Apr 16 20:57 ../
4 drwxr-xr-x 1 unangel 197121  0 Apr 16 21:15 .git/
5 -rw-r--r-- 1 unangel 197121 7360 Apr 16 20:57 .gitignore
6 -rw-r--r-- 1 unangel 197121 35823 Apr 16 20:57 LICENSE
7 -rw-r--r-- 1 unangel 197121  43 Apr 16 20:57 README.md
8 -rw-r--r-- 1 unangel 197121  0 Apr 16 21:16 list.txt
9
```

**Bottom Panel (Preview Mode):**

- Buttons: Edit, **Preview** (highlighted), Cancel changes, **Commit changes...** (highlighted).
- File Content (list.txt):

```
... @@ -3,6 +3,8 @@ drwxr-xr-x 1 unangel 197121  0 Apr 16 21:16 ./
3 drwxr-xr-x 1 unangel 197121  0 Apr 16 20:57 ../
4 drwxr-xr-x 1 unangel 197121  0 Apr 16 21:15 .git/
5 -rw-r--r-- 1 unangel 197121 7360 Apr 16 20:57 .gitignore
6 -rw-r--r-- 1 unangel 197121 35823 Apr 16 20:57 LICENSE
8 -rw-r--r-- 1 unangel 197121 35823 Apr 16 20:57 LICENSE
```
- Confirmation Message (highlighted):

```
6
7 file changed....
8
```



## 로컬 저장소 실습 - 9



- commit 정보 확인

- "<>Code" 화면에서 commit 메시지를 클릭하면 출력되는 페이지
- 해당 commit에서 수정된 내역에 대한 상세 정보를 제공
- commit에서 포함되지 않은 수정이 발생하지 않은 파일은 출력되지 않음
- 수정된 파일에서 추가되거나 삭제된 내용을 줄 단위로 확인 가능

Commit 1a223e8

ycs-wikim committed 44 minutes ago

1. list.txt  
[ + ] list current directory info.

main 1 parent be18584 commit 1a223e8

Filter files...

1 file changed +8 -0 lines changed

Search within code

list.txt

list.txt

@@ -0,0 +1,8 @@  
1 + total 49  
2 + drwxr-xr-x 1 unangel 197121 0 Apr 16 21:16 ./  
3 + drwxr-xr-x 1 unangel 197121 0 Apr 16 20:57 ../  
4 + drwxr-xr-x 1 unangel 197121 0 Apr 16 21:15 .git/  
5 + -rw-r--r-- 1 unangel 197121 7360 Apr 16 20:57 .gitignore  
6 + -rw-r--r-- 1 unangel 197121 35823 Apr 16 20:57 LICENSE  
7 + -rw-r--r-- 1 unangel 197121 43 Apr 16 20:57 README.md  
8 + -rw-r--r-- 1 unangel 197121 0 Apr 16 21:16 list.txt

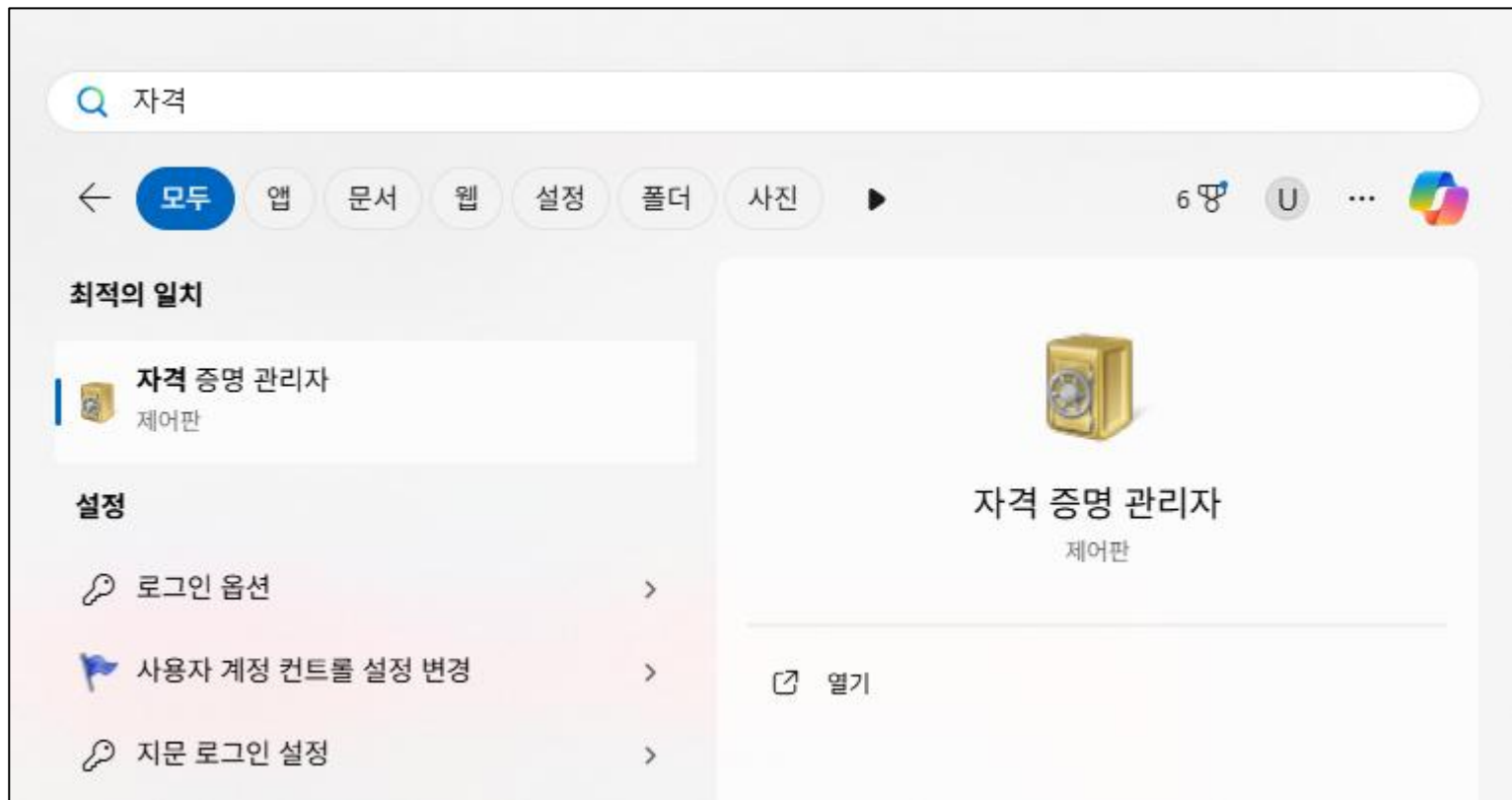


# github 인증 토큰 삭제 - 1



## • 인증 정보 삭제

- 다중 사용자 환경의 PC에 인증 정보(토큰)을 남기면 보안 문제 발생 가능
- 윈도우의 찾기 또는 제어판에서 "자격 증명 관리자"를 찾고 실행
- 실행되면 "Windows 자격 증명"을 선택





# github 인증 토큰 삭제 - 2

## • 자격 증명 삭제

- 삭제하지 않으면 본인의 인증 정보로 누구나 저장소 접근이 가능해짐
- 공용 PC라면 반드시 삭제하여 인증 정보에 의한 보안 사고를 방지해야 함

자격 증명 관리

웹 사이트, 연결된 응용 프로그램 및 네트워크에 대해 저장된 로그인 정보를 보고 삭제합니다.

웹 자격 증명      Windows 자격 증명

자격 증명 백업(B)    자격 증명 복원(R)

Windows 자격 증명  
Windows 자격 증명이 없습니다.

인증서 기반 자격 증명  
인증서가 없습니다.

일반 자격 증명

수정한 날짜	수정한 날짜	수정한 날짜	수정한 날짜
git:https://github.com	수정한 날짜: 2	수정한 날짜: 2	수정한 날짜: 2
	수정한 날짜: 2	수정한 날짜: 2	수정한 날짜: 2
	수정한 날짜: 2	수정한 날짜: 2	수정한 날짜: 2

인터넷 또는 네트워크 주소: git:https://github.com  
사용자 이름: ycs-wikim  
암호: .....  
지속성: 로컬 컴퓨터  
편집    **제거**

일반 자격 증명 삭제

이 일반 자격 증명을 완전히 삭제하시겠습니까?

인터넷 또는 네트워크 주소: git:https://github.com  
사용자 이름: ycs-wikim

**예(Y)**    아니요(N)