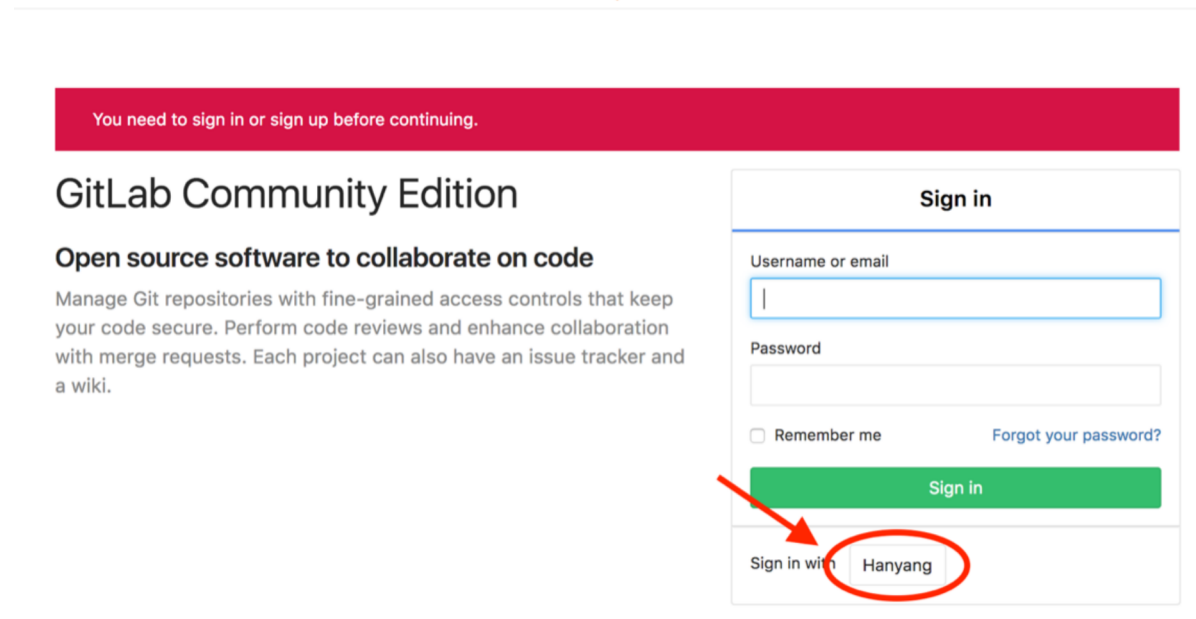


# GitLab for students

---

# GitLab login and initial setup

1. At [hconnect.hanyang.ac.kr](https://hconnect.hanyang.ac.kr), click "Sign in with Hanyang"



The screenshot shows the GitLab login interface. At the top, a red banner reads "You need to sign in or sign up before continuing." Below this, the text "GitLab Community Edition" is displayed, followed by the tagline "Open source software to collaborate on code" and a brief description of its features. On the right, the "Sign in" form is visible. It includes input fields for "Username or email" and "Password", a "Remember me" checkbox, and a "Forgot your password?" link. A green "Sign in" button is positioned below these fields. At the bottom of the form, there is a "Sign in with" section where the "Hanyang" provider is highlighted with a red circle. A red arrow points from the text in the first list item to this "Hanyang" button.

# GitLab login and initial setup

## 2. Login with Hanyang account



한양대학교 | 로그인

고객님의 정보에 접근하기 위하여 인증이 필요합니다.  
한양대학교 포털 한양인(HY-in)계정으로 로그인 하시기 바랍니다.

Portal Login

---

ID	<input type="text" value="2007002245"/>	<div>로그인</div>
Password	<input type="password" value="....."/>	

# GitLab login and initial setup

## 3. Agree to terms of information provision



 **한양대학교 | 개인정보의 제 3자 제공동의 요청**

한양대학교 OPEN API는 아래와 같은 개인정보를 온라인 소프트웨어 교육 지원 시스템 - Real 에 제공합니다.

**제공 받는자**

커넥트재단 (온라인 소프트웨어 교육 지원 시스템 - REAL)

**제공 목적**

웹상에서 학생 실습코드를 저장하고 빌드 하며 채점, 코드 리뷰를 수행하는 시스템 입니다.

실습 코드 저장은 GITLABCE를 활용할 예정이고  
코드 리뷰는 REVIEW BOARD, 빌드 및 채점은 JENKINS를 사용할 예정입니다.

한양대 도메인을 통해 서비스 하고 한양대 학생 인증을 적용할 예정입니다.

\*\* 실제 서비스를 사용자가 사용하는 환경 입니다.

**제공 항목**

모든 항목에 동의하시어만 이용 가능합니다.

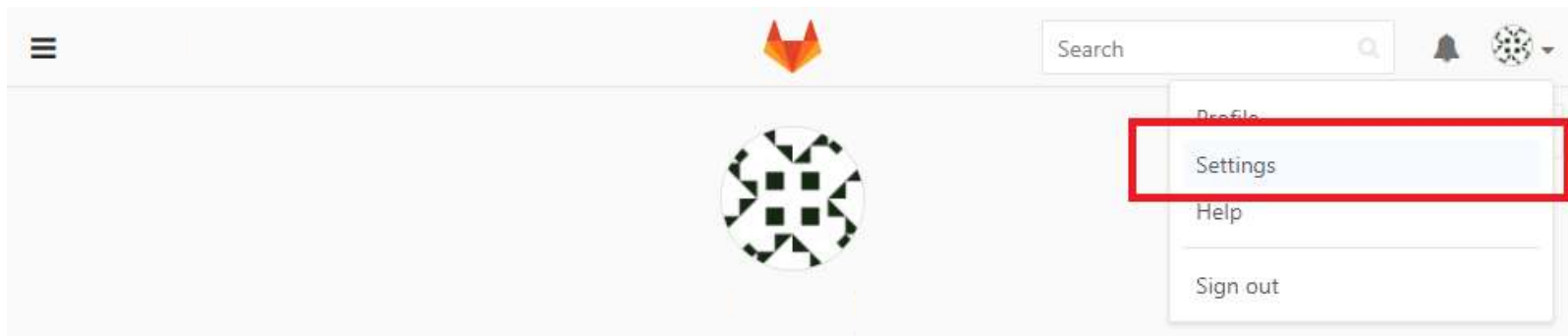
**로그인사용자 정보조회**

[포털에서 설정한 대표 신분 정보]  
로그인한 사용자의 성명, 사용자ID, 학번(개인번호), 재학(재직) 여부, 소속대학, 소속명, 소속코드, 소속ID, 사용자구분명의 정보를 제공합  
니다.

☒ 전체 동의합니다. ☐ 동의합니다.

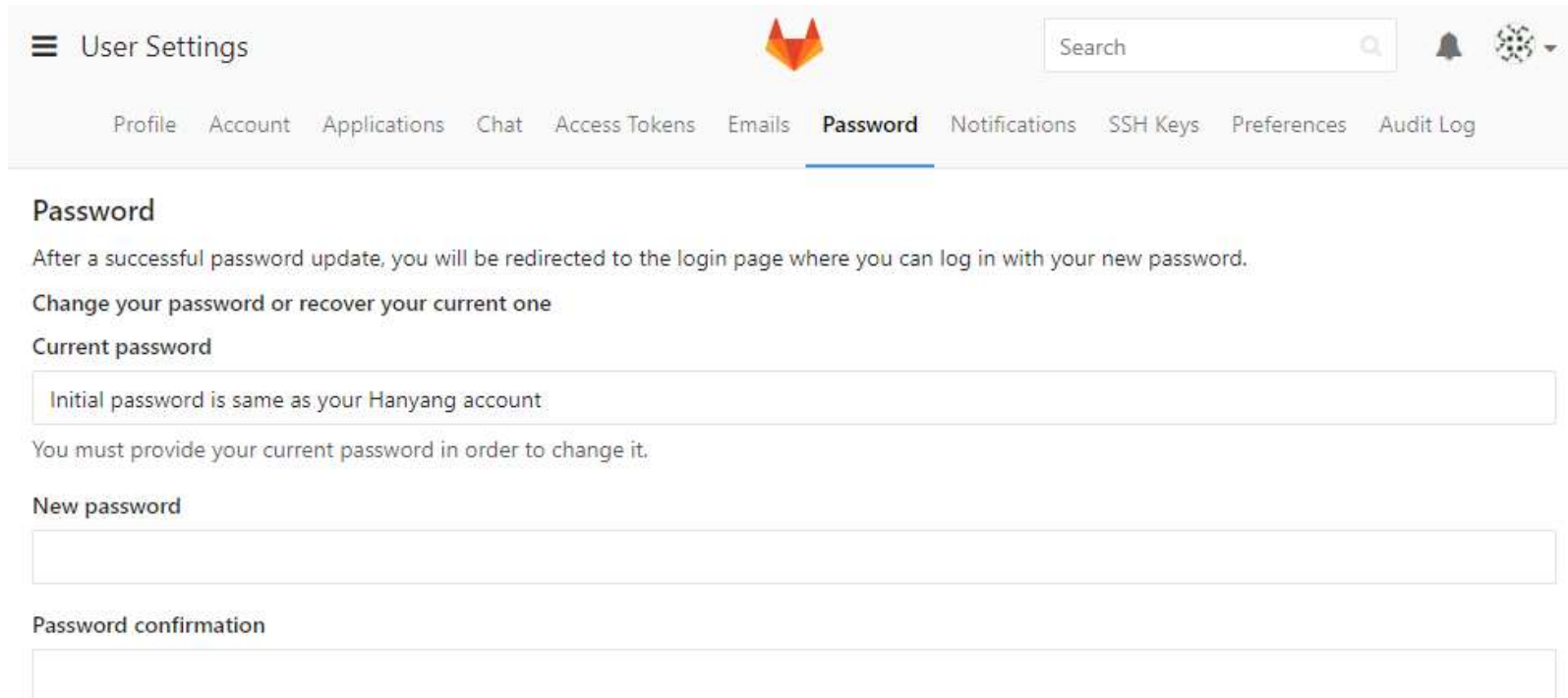
# GitLab login and initial setup

## 4. Set password



# GitLab login and initial setup

## 4. Set password (DO NOT FORGET !!– different password is allowed)



The screenshot shows the GitLab User Settings interface. At the top, there's a navigation bar with a hamburger menu, the text 'User Settings', the GitLab logo, a search bar, and notification icons. Below this is a horizontal menu with links: Profile, Account, Applications, Chat, Access Tokens, Emails, Password (which is underlined), Notifications, SSH Keys, Preferences, and Audit Log. The main content area is titled 'Password'. It contains a paragraph: 'After a successful password update, you will be redirected to the login page where you can log in with your new password.' Below this is a link: 'Change your password or recover your current one'. Then, there's a section for 'Current password' with a text input field containing the text 'Initial password is same as your Hanyang account'. Below this is a note: 'You must provide your current password in order to change it.' Then, there's a section for 'New password' with an empty text input field. Finally, there's a section for 'Password confirmation' with an empty text input field.

User Settings

Search

Profile Account Applications Chat Access Tokens Emails **Password** Notifications SSH Keys Preferences Audit Log

### Password

After a successful password update, you will be redirected to the login page where you can log in with your new password.

[Change your password or recover your current one](#)

Current password

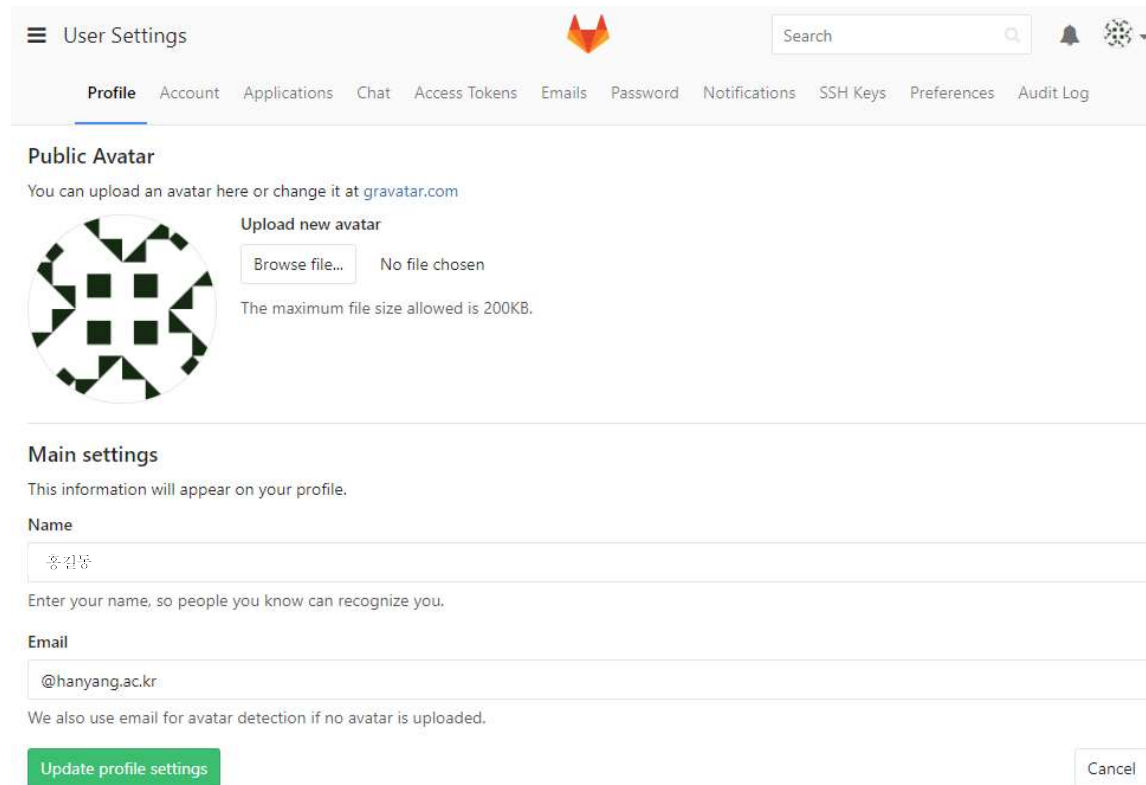
You must provide your current password in order to change it.

New password

Password confirmation

# GitLab login and initial setup

## 5. Set email



The screenshot shows the GitLab 'User Settings' page. The top navigation bar includes a hamburger menu, the text 'User Settings', the GitLab logo, a search bar, and notification icons. Below this is a sub-navigation bar with tabs: Profile (selected), Account, Applications, Chat, Access Tokens, Emails, Password, Notifications, SSH Keys, Preferences, and Audit Log. The main content area is divided into two sections. The first section, 'Public Avatar', includes a circular placeholder for a profile picture and an 'Upload new avatar' button. Below the button is a 'Browse file...' button and the text 'No file chosen'. A note states 'The maximum file size allowed is 200KB.' The second section, 'Main settings', contains a note 'This information will appear on your profile.' followed by two text input fields. The first field is labeled 'Name' and contains the Korean text '홍길동'. The second field is labeled 'Email' and contains '@hanyang.ac.kr'. Below the email field is a note 'We also use email for avatar detection if no avatar is uploaded.' At the bottom of the form are two buttons: a green 'Update profile settings' button and a grey 'Cancel' button.

User Settings

Profile Account Applications Chat Access Tokens Emails Password Notifications SSH Keys Preferences Audit Log

**Public Avatar**

You can upload an avatar here or change it at [gravatar.com](https://gravatar.com)

Upload new avatar

Browse file... No file chosen

The maximum file size allowed is 200KB.

**Main settings**

This information will appear on your profile.

**Name**

홍길동

Enter your name, so people you know can recognize you.

**Email**

@hanyang.ac.kr

We also use email for avatar detection if no avatar is uploaded.

Update profile settings Cancel

# GitLab login and initial setup

---

5. Set email – confirm email address in your inbox



# GitLab login and initial setup

6. After setting password, you can login without clicking "Sign in with Hanyang"



## GitLab Community Edition

### Open source software to collaborate on code

Manage Git repositories with fine-grained access controls that keep your code secure. Perform code reviews and enhance collaboration with merge requests. Each project can also have an issue tracker and a wiki.

### Sign in

Username or email

2007002245

Password

.....|

☐ Remember me

[Forgot your password?](#)

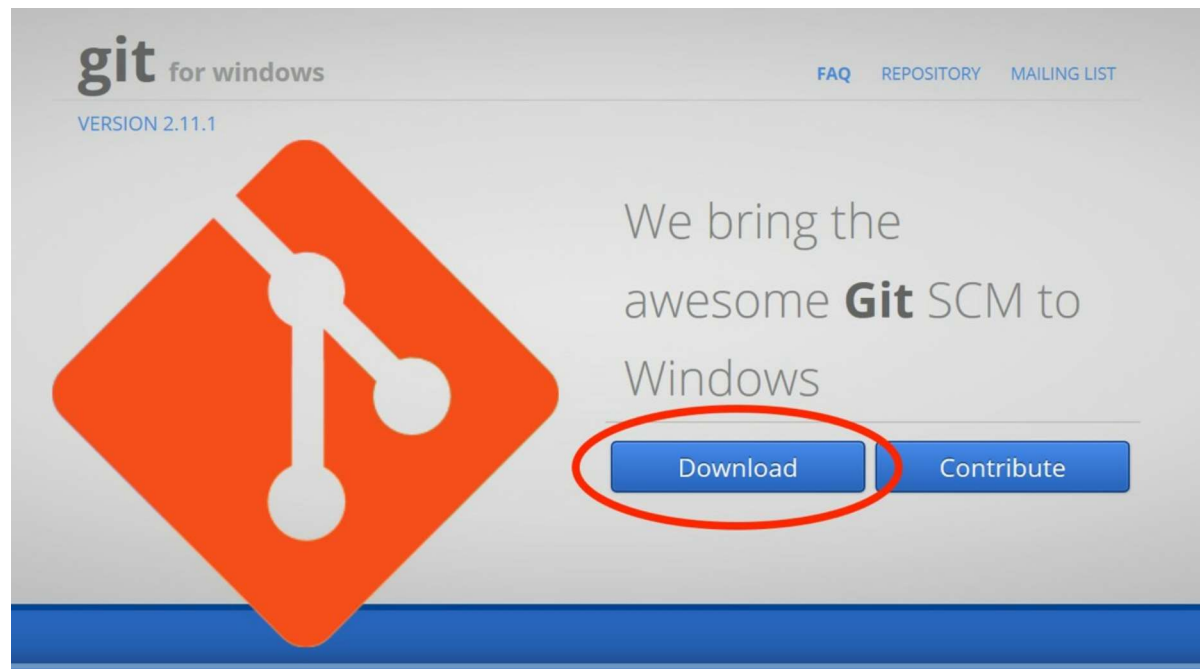
Sign in

Sign in with

Hanyang

# Git Installation (Windows)

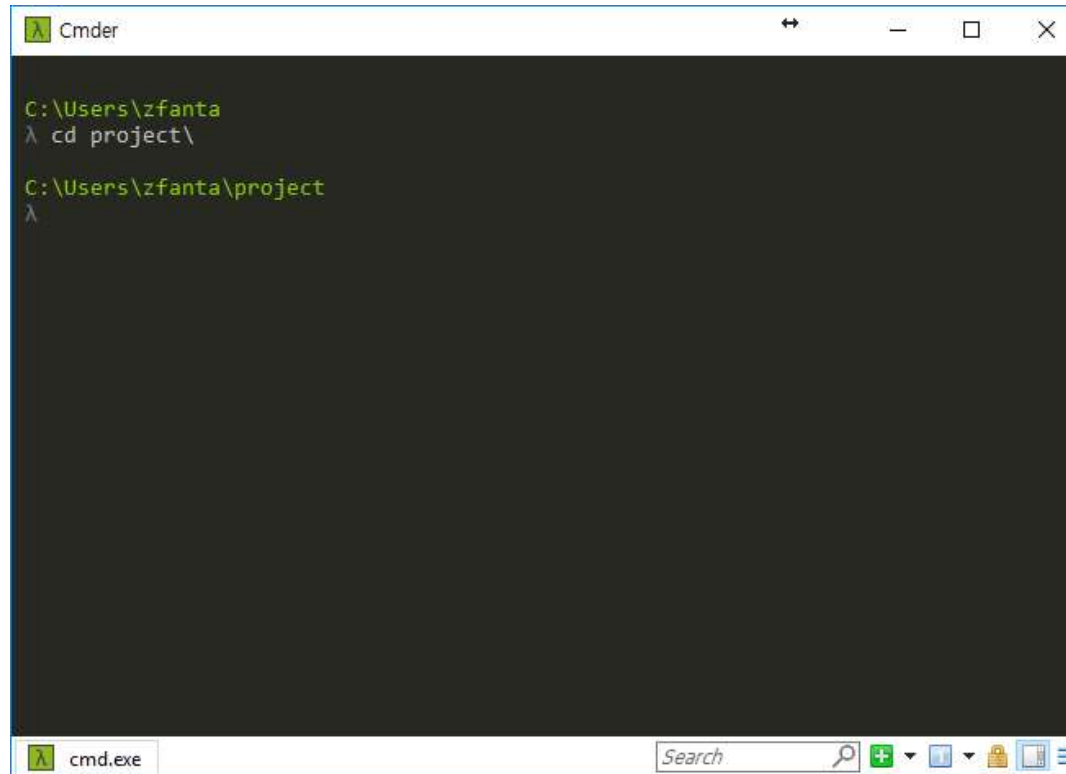
1. <https://git-for-windows.github.io/>



# Git Installation (Windows)

---

2. Launch Git Bash
3. Move to working directory (ex: \$ cd project)



```
C:\Users\zfanta
λ cd project\


C:\Users\zfanta\project
λ
```

The screenshot shows a Windows Command Prompt window titled 'Cmder'. The prompt is at 'C:\Users\zfanta' and the user has entered 'cd project\' to change the directory. The new prompt is now 'C:\Users\zfanta\project'. The taskbar at the bottom shows 'cmd.exe' and a search bar.

# Git Installation (Linux)

Ubuntu `$ sudo apt-get install git`

Fedora `$ sudo yum install git`



A terminal window titled "mrbin2002 — mrbin2002@ubuntu: ~ — ssh mrbin2002@10.211.55.7 — 74x21". The terminal shows the command `sudo apt-get install git` being executed. The output indicates that Git is already the newest version and no upgrades are needed.

```
mrbin2002@ubuntu:~$ sudo apt-get install git
[sudo] password for mrbin2002:
Reading package lists... Done
Building dependency tree
Reading state information... Done
git is already the newest version.
0 upgraded, 0 newly installed, 0 to remove and 51 not upgraded.
mrbin2002@ubuntu:~$
```

# Git Installation (MacOS)

---

1. <https://git-scm.com/download/mac>

2. Mount dmg and launch git-x.x.x-xxx.pkg



# Git Basic Usage

---

1. After installation, set user

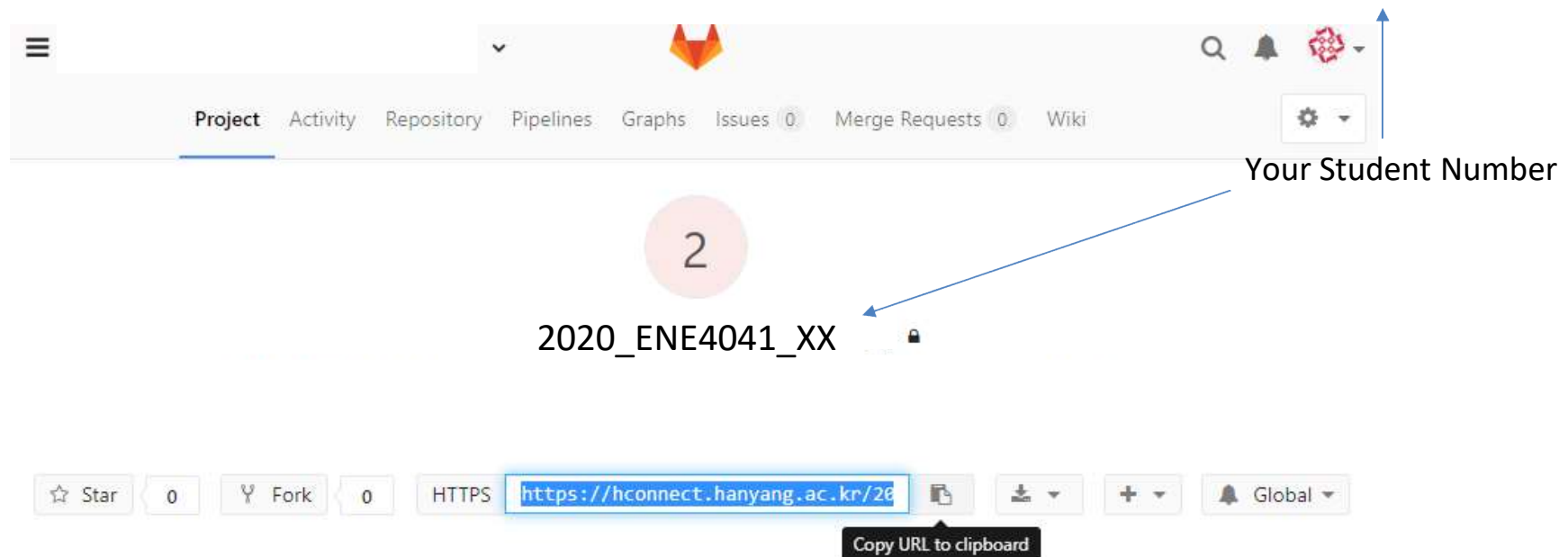
```
$ git config --global user.name "20xxxxxxx"  
$ git config --global user.email "20xxxxxxx@hanyang.ac.kr"
```

( user.name is student id,  
user.email is email registered Gitlab(<https://hconnect.hanyang.ac.kr>)

# Git Basic Usage

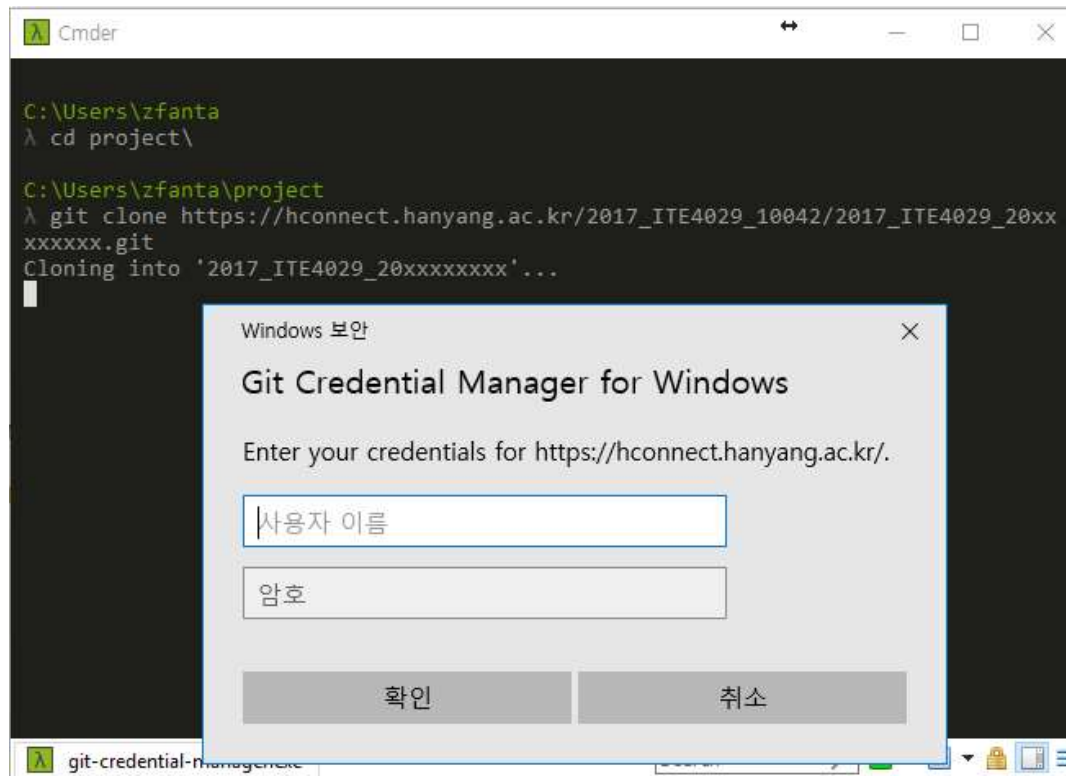
## 2. Clone your Git repository

```
$ git clone https://hconnect.hanyang.ac.kr/2020_ENE4041_11791/2020_ENE4041_XXXXXXXXXX.git
```



# Git Basic Usage

3. Enter username(student id) and password(set in GitLab)





# Git Basic Usage

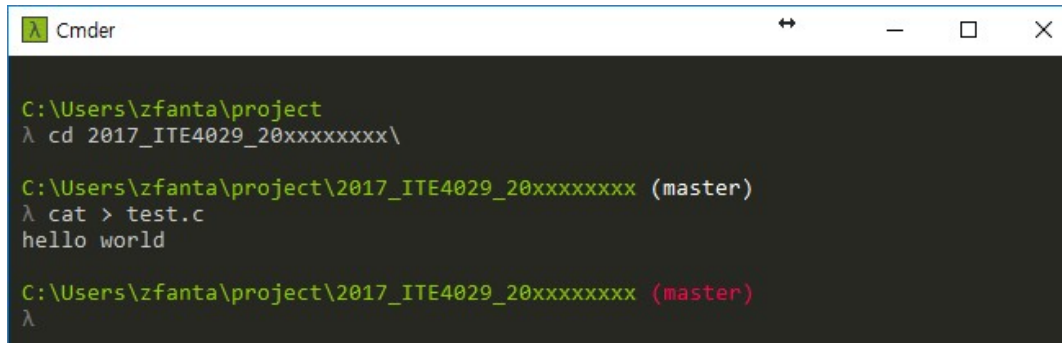
---

## 4. Move to cloned directory

```
$ cd 2020_ENE4041_XXXXXXXXX
```

## 5. Create file

```
$ vi test.c
```



```
C:\Users\zfanta\project
λ cd 2017_ITE4029_20xxxxxxxxx\

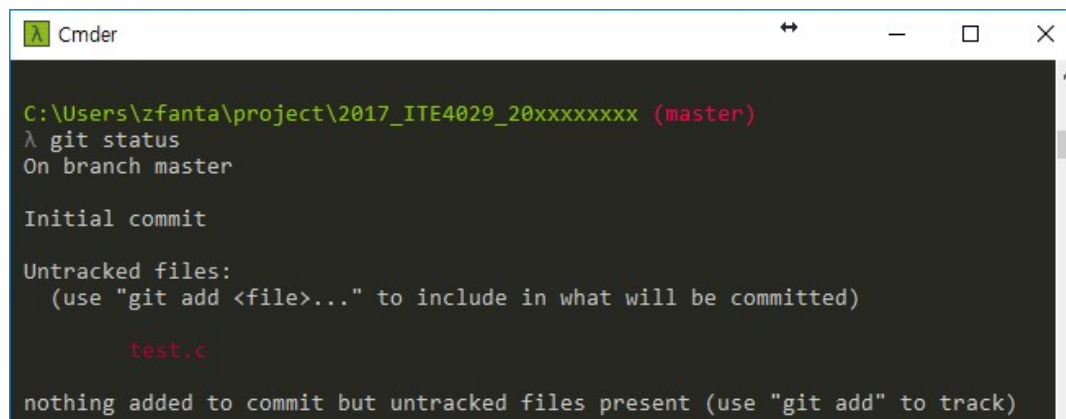
C:\Users\zfanta\project\2017_ITE4029_20xxxxxxxxx (master)
λ cat > test.c
hello world

C:\Users\zfanta\project\2017_ITE4029_20xxxxxxxxx (master)
λ
```

# Git Basic Usage

6. If you check current status, created file is categorized as untracked.

```
$ git status
```



```
C:\Users\zfanta\project\2017_ITE4029_20xxxxxxx (master)
λ git status
On branch master

Initial commit

Untracked files:
  (use "git add <file>..." to include in what will be committed)

       test.c

nothing added to commit but untracked files present (use "git add" to track)
```

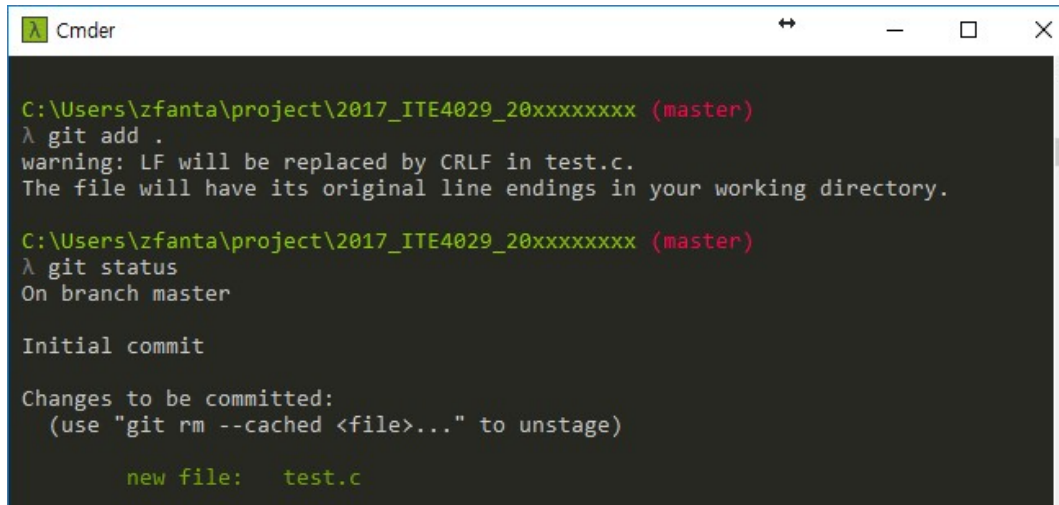
# Git Basic Usage

7. Move all created or modified files in directory to staged area

```
$ git add .
```

8. Check status again

```
$ git status
```



```
C:\Users\zfanta\project\2017_ITE4029_20xxxxxxx (master)
λ git add .
warning: LF will be replaced by CRLF in test.c.
The file will have its original line endings in your working directory.

C:\Users\zfanta\project\2017_ITE4029_20xxxxxxx (master)
λ git status
On branch master

Initial commit

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

        new file:   test.c
```

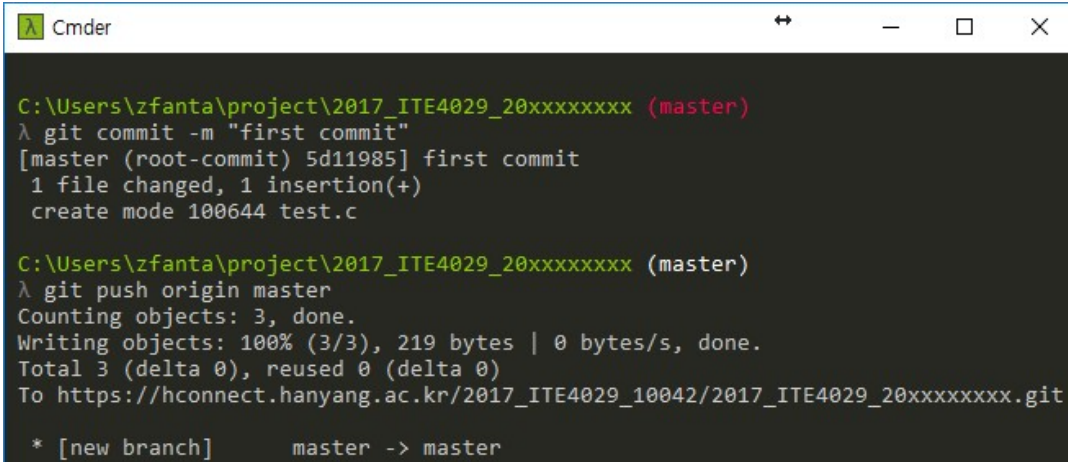
# Git Basic Usage

9. Commit added or modified files(affect only local repository)

```
$ git commit -m "first commit"
```

10. Push commits to remote repository

```
$ git push origin master
```



```
C:\Users\zfanta\project\2017_ITE4029_20xxxxxxx (master)
λ git commit -m "first commit"
[master (root-commit) 5d11985] first commit
1 file changed, 1 insertion(+)
create mode 100644 test.c

C:\Users\zfanta\project\2017_ITE4029_20xxxxxxx (master)
λ git push origin master
Counting objects: 3, done.
Writing objects: 100% (3/3), 219 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://hconnect.hanyang.ac.kr/2017_ITE4029_10042/2017_ITE4029_20xxxxxxx.git
* [new branch]      master -> master
```

# Git Basic Usage

## 11. Commits sent to the remote repository via git push are visible in the GitLab webpage

The screenshot displays the GitLab web interface for a project named '2020\_ENE4041\_2020119717'. The interface includes a sidebar with navigation options like Project overview, Details, Activity, Releases, Repository, Issues, Merge Requests, CI / CD, Operations, Analytics, Wiki, Snippets, and Settings. The main content area shows the project details, including the project ID (284), 2 commits, 1 branch, 0 tags, and 164 KB of files. A commit by 'Test, Kyujin Kim's' is highlighted. Below the commit, there are buttons for 'Delete week1\_2020119717', 'Add README', 'Add LICENSE', 'Add CHANGELOG', 'Add CONTRIBUTING', and 'Add Kubernetes cluster'. The bottom of the page shows a table with columns for Name, Last commit, and Last update.