

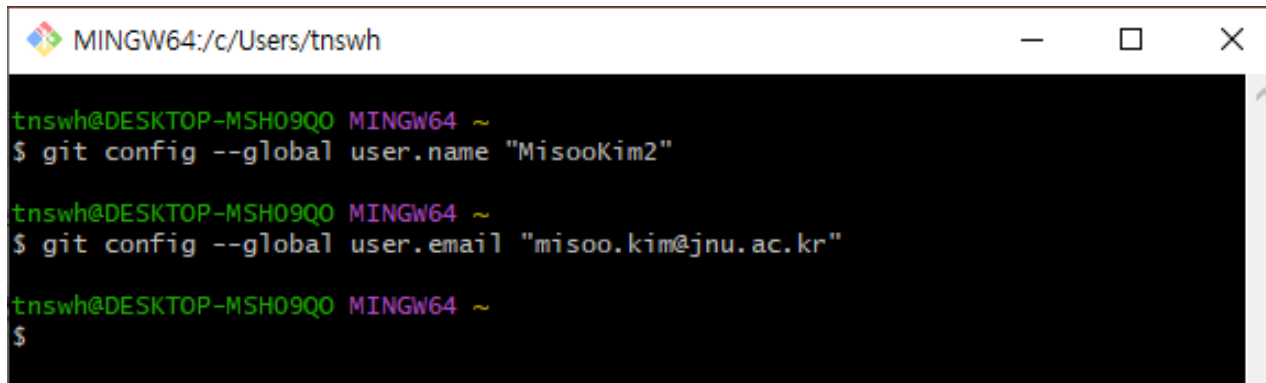
GitHub 문제 / 해결 모음

V1. 2023.09.06

내 PC에서 고정된 사용자 정보 사용하기

• 깃 환경 설정하기

- 버전을 저장할 때 마다 사용자 정보를 함께 저장하기 때문에, 사용자 정보 저장
- 앞으로 Git 사용시 명시할 사용자 정보를 GitBash에 입력
 - `git config --global user.name "GitHubID"`
 - `git config --global user.email "GitHubEmail"`



```
MINGW64:/c/Users/tnswh

tnswh@DESKTOP-MSH09QO MINGW64 ~
$ git config --global user.name "MisooKim2"

tnswh@DESKTOP-MSH09QO MINGW64 ~
$ git config --global user.email "misoo.kim@jnu.ac.kr"

tnswh@DESKTOP-MSH09QO MINGW64 ~
$
```

문제 상황 & 해결 방법 (1/11)

• 모르고 파일을 삭제하고 커밋해버렸어요!

>> 과거 커밋으로 저장소를 되돌리기 (커밋 두 개 이상 있어야함)

1. 가장 최근 커밋으로 돌리기: `git reset HEAD^`

• HEAD^: 가장 최근 커밋

2. 특정 커밋으로 돌리기: `git reset ---hard` 커밋해시

• 커밋 해시: 커밋에 부여된 아이디

The image contains three screenshots illustrating the Git commit history and the process of resetting to a previous commit.

- Left Screenshot:** A GitHub repository page for 'CPP202309lic'. It shows the 'main' branch with two commits: 'second commit' (by MisookKim2, 10 minutes ago) and 'first commit' (by MisookKim2, 17 minutes ago). The 'first commit' is highlighted in blue.
- Middle Screenshot:** A 'Commits' view for the 'main' branch. It shows the commit history on August 28, 2023. The 'second commit' (hash: 064ecaa) is highlighted in blue, and the 'first commit' (hash: c8f3d2b) is highlighted in red.
- Right Screenshot:** A 'Commit' view for the 'first commit' (hash: c8f3d2b). It shows the commit message 'first commit' and the file 'practice/week1/hello_world.c' with 7 additions and 0 deletions. The commit hash 'c8f3d2b' is highlighted in red.

문제 상황 & 해결 방법 (2/11)

- 이전 커밋과 차이가 없는 최신 커밋 상태라 커밋할 게 없음

```
$ git commit -m "message"
On branch main
Your branch is up to date with 'origin/main'.
nothing to commit, working tree clean
```

- 해결: 커밋 메시지를 변경하고 싶은 경우

```
git commit -amend -m "새 메세지"
```

- 변경된 커밋이 없어서 푸시할 게 없음

```
$ git push -u origin main
Everything up-to-date
branch 'main' set up to track 'origin/main'.
```

- 해결: 변경 사항이 있을 때 add > commit > push를 다시 수행

문제 상황 & 해결 방법 (3/11)

- 수정된 파일을 스테이징 하지 않고 커밋을 하려는 경우

```
$ git commit -m "message"
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   test.txt

no changes added to commit (use "git add" and/or "git commit -a")
```

- 해결: 전체 파일을 다시 스테이징 > 다시 커밋

```
git add .
git commit -m "메시지"
```

```
user@ROSE-JNU MINGW64 ~/Desktop/test (main)
$ git add .

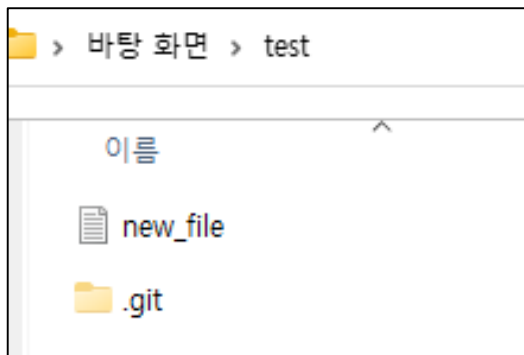
user@ROSE-JNU MINGW64 ~/Desktop/test (main)
$ git commit -m "message"
[main eb9781f] message
1 file changed, 1 insertion(+)
```

문제 상황 & 해결 방법 (4/11)

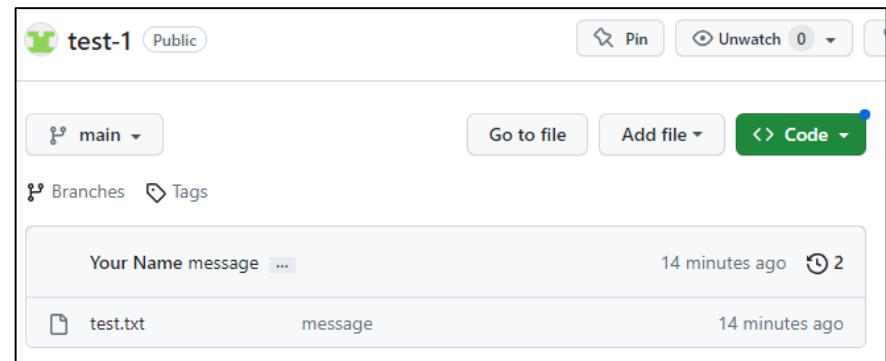
- 로컬 저장소에 원격 저장소에 있는 파일이 없어 PUSH 불가
(뒤에 이어서)

```
$ git push -u origin main
To https://github.com/MisooKim2/test-1.git
! [rejected]        main -> main (fetch first)
error: failed to push some refs to 'https://github.com/MisooKim2/test-1.git'
hint: Updates were rejected because the remote contains work that you do not
hint: have locally. This is usually caused by another repository pushing to
hint: the same ref. If you want to integrate the remote changes, use
hint: 'git pull' before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.
```

- 예) 아래와 같은 상태에서 push를 할 경우



지역 저장소



원격 저장소

문제 상황 & 해결 방법 (5/11)

- 로컬 저장소에 원격 저장소에 있는 파일이 없어 PUSH 불가
(뒤에 이어서)

```
$ git push -u origin main
To https://github.com/Misookim2/test-1.git
! [rejected]        main -> main (fetch first)
error: failed to push some refs to 'https://github.com/Misookim2/test-1.git'
hint: Updates were rejected because the remote contains work that you do not
hint: have locally. This is usually caused by another repository pushing to
hint: the same ref. If you want to integrate the remote changes, use
hint: 'git pull' before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.
```

- 해결: 원격 저장소에 있는 파일 가져오기

```
git pull origin main
```

>> 추가 문제 발생 할 수 있음 (뒤에 이어서)

```
$ git pull origin main
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 6 (delta 0), reused 6 (delta 0), pack-reused 0
Unpacking objects: 100% (6/6), 394 bytes | 14.00 KiB/s, done.
From https://github.com/Misookim2/test-1
 * branch                main          -> FETCH_HEAD
 * [new branch]          main          -> origin/main
fatal: refusing to merge unrelated histories
```

문제 상황 & 해결 방법 (6/11)

- 로컬 저장소에 원격 저장소에 있는 파일이 없어 PUSH 불가
- >> 로컬에 원격저장소와 관련된 기록이 없어서 PULL 불가

```
$ git pull origin main
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 6 (delta 0), reused 6 (delta 0), pack-reused 0
Unpacking objects: 100% (6/6), 394 bytes | 14.00 KiB/s, done.
From https://github.com/MisookKim2/test-1
 * branch          main      -> FETCH_HEAD
 * [new branch]    main      -> origin/main
fatal: refusing to merge unrelated histories
```

- 해결: 아래 명령어로 관련되지 않은 기록도 함께 PULL

```
git pull origin main --allow-unrelated-histories
```


문제 상황 & 해결 방법 (7/11)

- 원격 저장소에 업로드할 권한이 없음

```
remote: Permission to https://github.com/MisookKim2/test-1.git denied to RosePasta.  
fatal: unable to access  
'https://github.com/MisookKim2/test-1.git/': The requested URL returned error: 403  
:
```

- 해결 1: 아래 명령어를 입력해 repository 주소에 계정 정보 주기

```
git remote set-url origin https://아이디@github.com/아이디/저장소이름.git
```

위 작업 후 다시 push 해보기

- 해결 2: 크롬 / 엣지에 로그인 된 GitHub 로그아웃 > 다시 push
- 해결 3: 윈도우 찾기 > 자격 증명 관리자 > window 자격 증명
> 일반 자격 증명 > GitHub 관련 계정 제거 >> 다시 push

문제 상황 & 해결 방법 (8/11)

- 원격 저장소 주소가 이미 등록되어 있음

```
$ git remote add origin "https://github.com/MisookKim2/test-1"  
error: remote origin already exists.
```

- 해결: 원격 저장소 주소를 변경하고 싶은 경우 아래 명령어를 입력

```
git remote set-url origin "새 저장소 주소"
```

- 인증 방식이 password에서 토큰 방식으로 바뀜 (뒤에 이어서)

```
remote: Support for password authentication was removed on August 13, 2021. Please use a personal access token instead.  
remote: Please see https://github.blog/2020-12-15-token-authentication-requirements-for-git-operations/ for more information.  
fatal: unable to access 'https://github.com/MisookKim2/test-01.git/': The requested URL returned error: 403
```

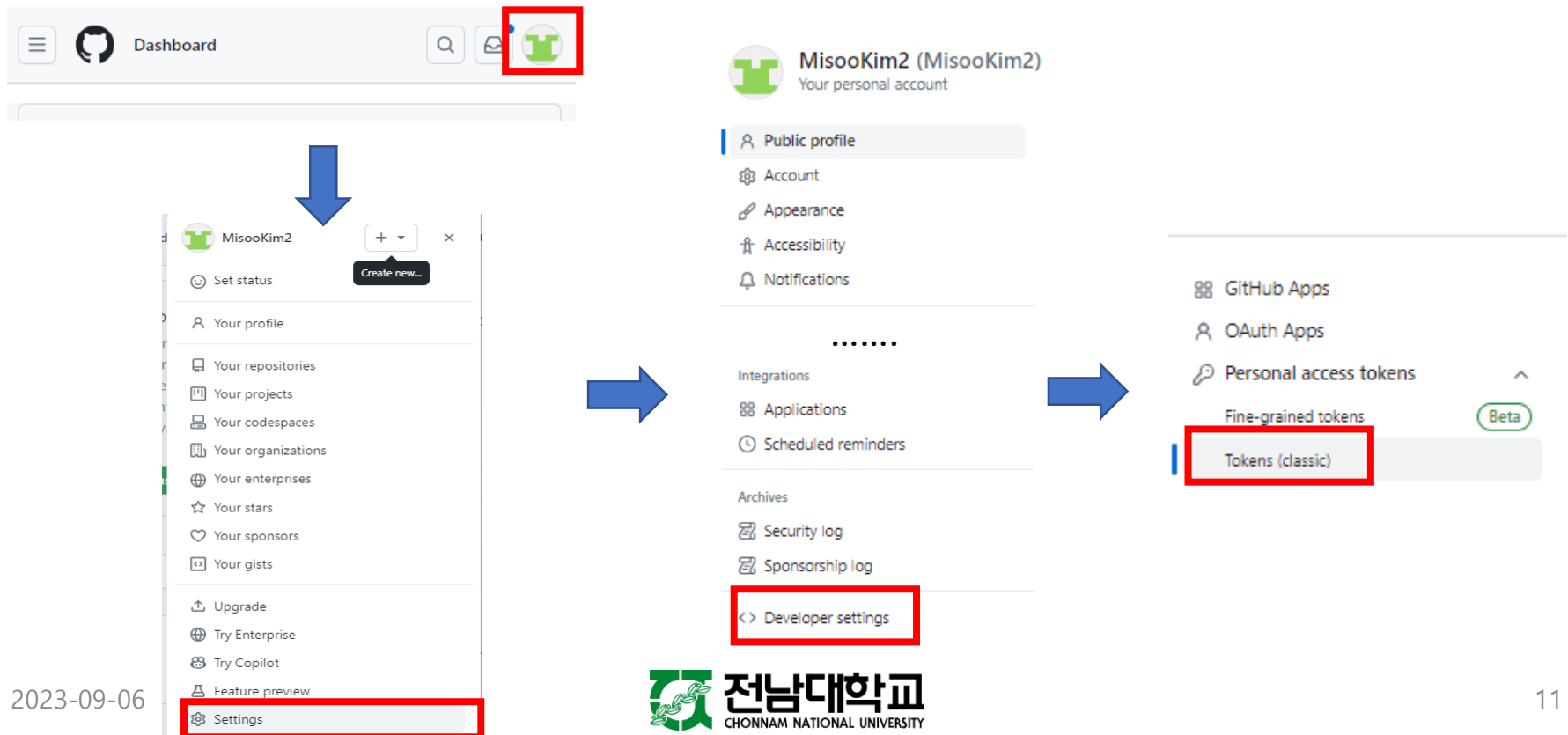
- 해결 1: Git을 최신 버전으로 다시 설치
- 해결 2: GitHub에서 토큰 받은 후 비밀번호 대신 입력

문제 상황 & 해결 방법 (9/11)

- 인증 방식이 password에서 토큰 방식으로 바뀜

- 토큰 받는 방법

1) 대시보드 > 사용자 정보 > settings > Developer setting



문제 상황 & 해결 방법 (10/11)

- 인증 방식이 password에서 토큰 방식으로 바뀜
 - 토큰 받는 방법
 - 1) 대시보드 > 사용자 정보 > settings > Developer setting
 - 2) Generate new token > generate new token (classic)
 - > Note: PL_class_test (변경 가능), Expiration : 2023-12-22

Personal access tokens (classic)

Tokens you have generated that can be used to access the Git

PL_class — admin:enterprise, admin:pgp_key, admin:org, admin:org_hooks, admin:repo_hook, admin:ssh signing key, audit log, codespace, copilot,

Generate new token

Revoke all

Generate new token (Beta)

Fine-grained, repo-scoped

Generate new token (classic)

For general use

Delete



New personal access token (classic)

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

Note

PL_class_test

What's this token for?

Expiration *

Custom...

2023-12-22

문제 상황 & 해결 방법 (11/11)

- 인증 방식이 password에서 토큰 방식으로 바뀜

- 토큰 받는 방법

- 1) 대시보드 > 사용자 정보 > settings > Developer setting
- 2) Generate new token > generate new token (classic)
> Note: PL_class_test (변경 가능), Expiration : 2023-12-22
- 3) 체크박스 모두 체크 > Generate Token > Token 복사 후 비번으로 사용 (토큰은 한 번만 보여줌. 메모장에 저장 후 계속 사용. 까먹을 경우 다시 발급받기)

