

1. GitHub에 Coupling Project 리모트 Empty 레파지토리 생성

The screenshot shows the GitHub 'Create a new repository' page for user Mavben. The repository name is 'Coupling-Project', which is marked as available. The repository is set to be public. The 'Initialize this repository with' section has 'Add a README file' checked. The '.gitignore' template is set to 'None', and the license is also set to 'None'. On the right, the repository page for 'Coupling-Project' is visible, showing options to set up GitHub Copilot, invite collaborators, and quick setup instructions for desktop, command line, and existing repositories.

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 * **Mavben** / Repository name * **Coupling-Project**
✓ Coupling-Project is available.

Great repository names are short and memorable. Need inspiration? How about [reimagined-palm-tree](#) ?

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.](#)

Coupling-Project Public

[Set up GitHub Copilot](#)
Use GitHub's AI pair programmer to autocomplete suggestions as you code.

[Invite collaborators](#)
Find people using their GitHub username or email address.

Quick setup — if you've done this kind of thing before

[Set up in Desktop](#) or [HTTPS](#) [SSH](#) <https://github.com/Mavben/Coupling-Project.git>

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

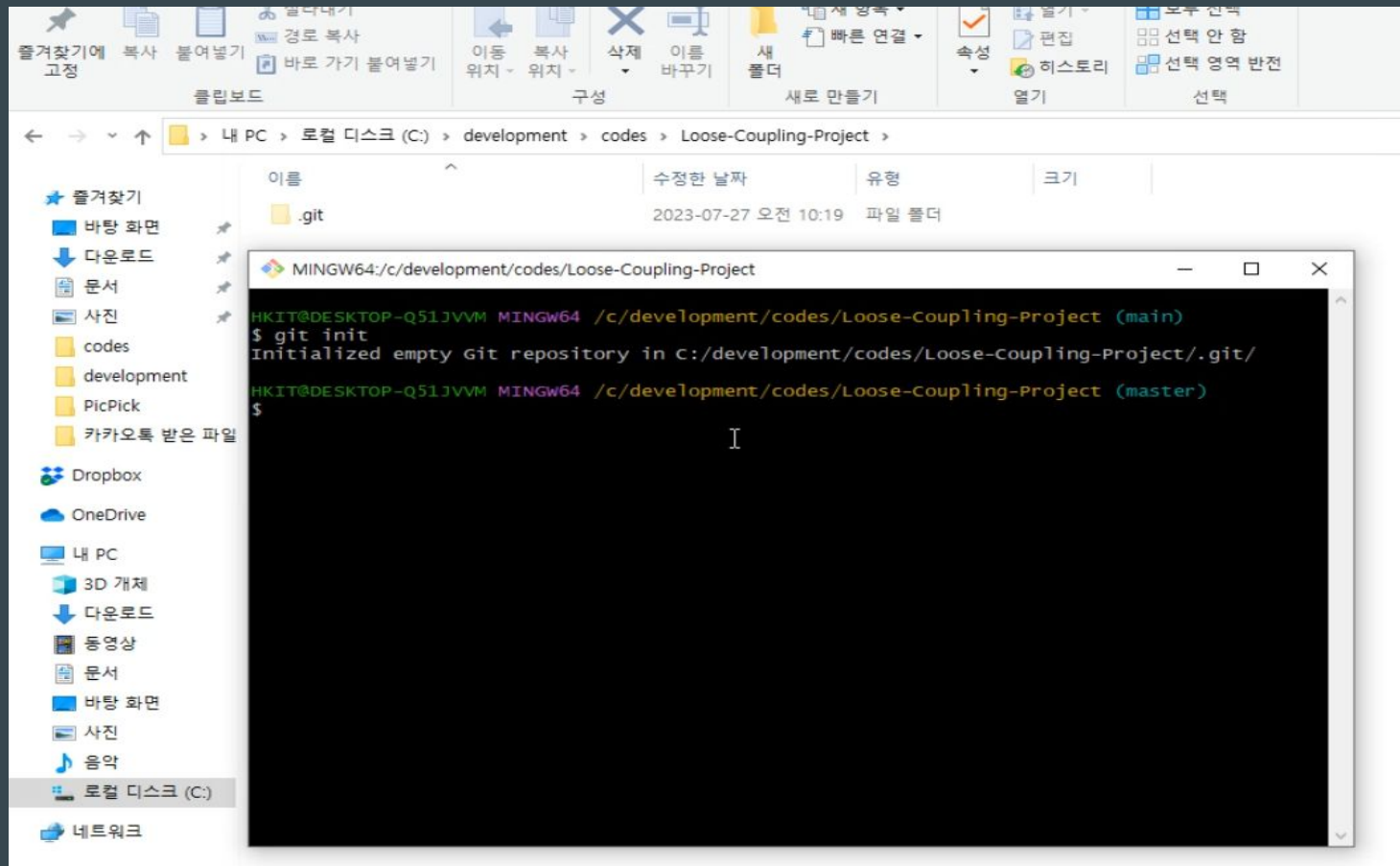
...or create a new repository on the command line

```
echo "# Coupling-Project" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/Mavben/Coupling-Project.git
git push -u origin main
```

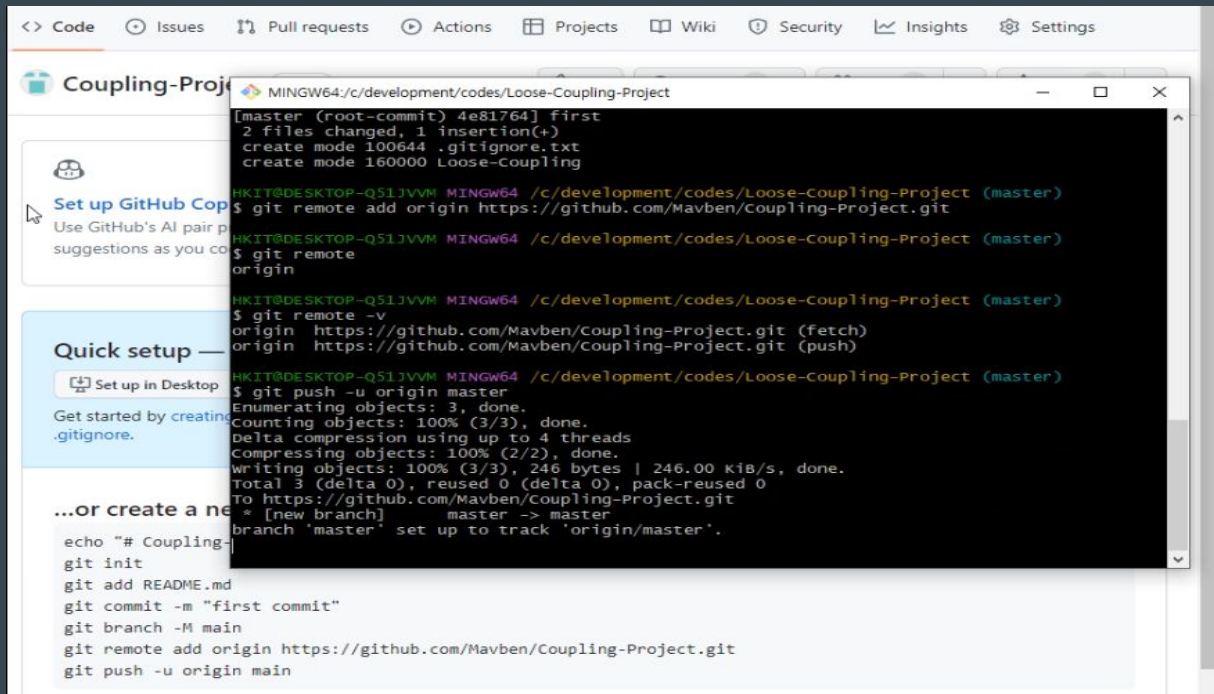
...or push an existing repository from the command line

```
git remote add origin https://github.com/Mavben/Coupling-Project.git
git branch -M main
```

2. Loose Coupling Project에 Git 로컬 레파지토리 생성(.gitignore 적용)



3. 리모트 레파지토리에 Push(source 파일만 push)



The screenshot shows the GitHub web interface for the 'Coupling-Project' repository. The terminal window displays the following commands and output:

```
MINGW64/c:/development/codes/Loose-Coupling-Project
[master (root-commit) 4e81764] first
2 files changed, 1 insertion(+)
create mode 100644 .gitignore.txt
create mode 160000 Loose-Coupling

HKIT@DESKTOP-Q51JVVW MINGW64 /c:/development/codes/Loose-Coupling-Project (master)
$ git remote add origin https://github.com/Mavben/Coupling-Project.git

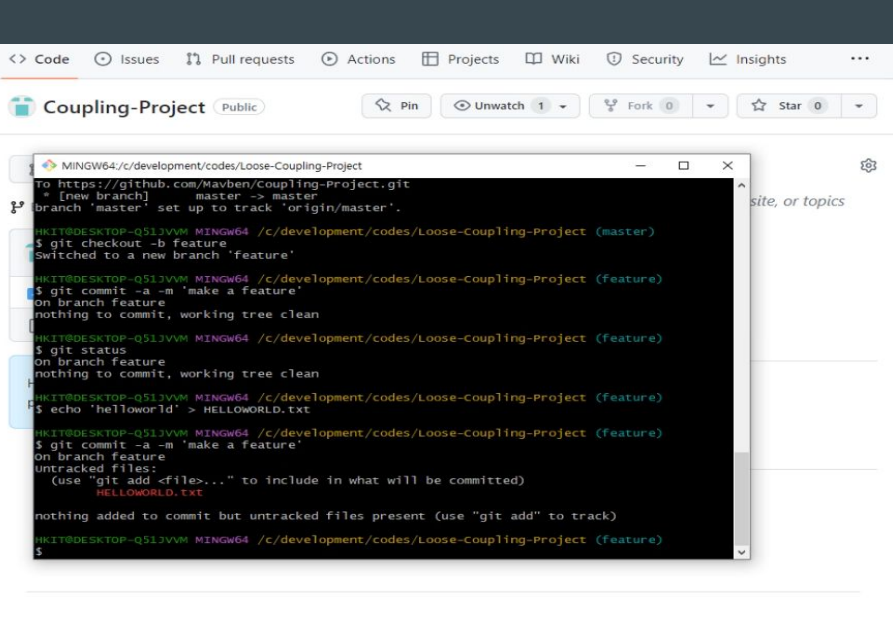
HKIT@DESKTOP-Q51JVVW MINGW64 /c:/development/codes/Loose-Coupling-Project (master)
$ git remote
origin

HKIT@DESKTOP-Q51JVVW MINGW64 /c:/development/codes/Loose-Coupling-Project (master)
$ git remote -v
origin https://github.com/Mavben/Coupling-Project.git (fetch)
origin https://github.com/Mavben/Coupling-Project.git (push)

HKIT@DESKTOP-Q51JVVW MINGW64 /c:/development/codes/Loose-Coupling-Project (master)
$ git push -u origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
writing objects: 100% (3/3), 246 bytes | 246.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Mavben/Coupling-Project.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
```

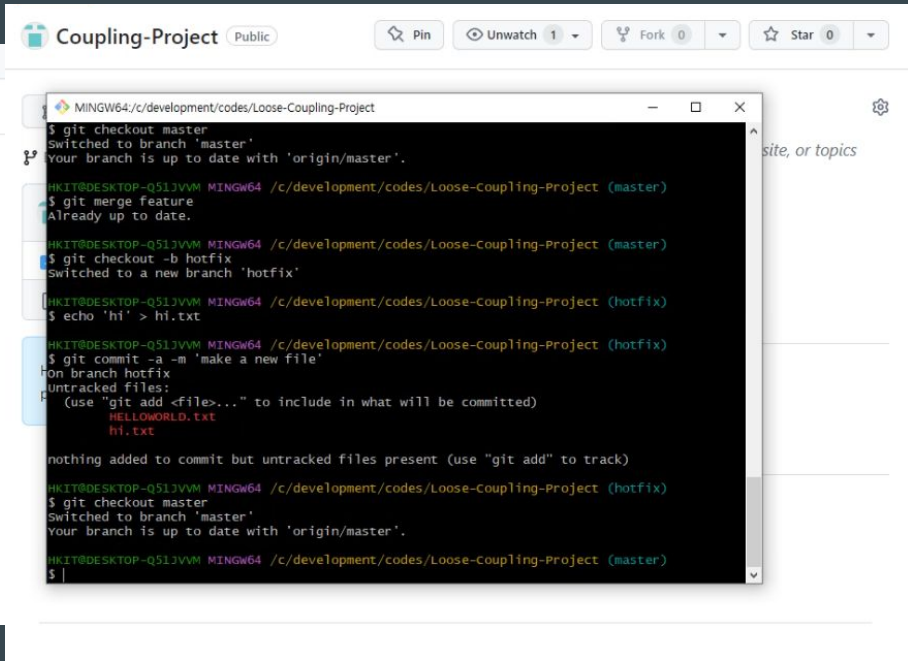
The GitHub page shows the repository name 'Coupling-Project' and a sidebar with 'Set up GitHub Copilot' and 'Quick setup' sections. The 'Quick setup' section includes a button 'Set up in Desktop' and a link 'Get started by creating .gitignore.' The '...or create a new repository' section shows a list of commands to initialize a repository and push to GitHub.

4. Loose Coupling Project에 토픽 브랜치 생성



The screenshot shows the GitHub repository page for 'Coupling-Project' (Public) with a terminal window open. The terminal displays the following commands and output:

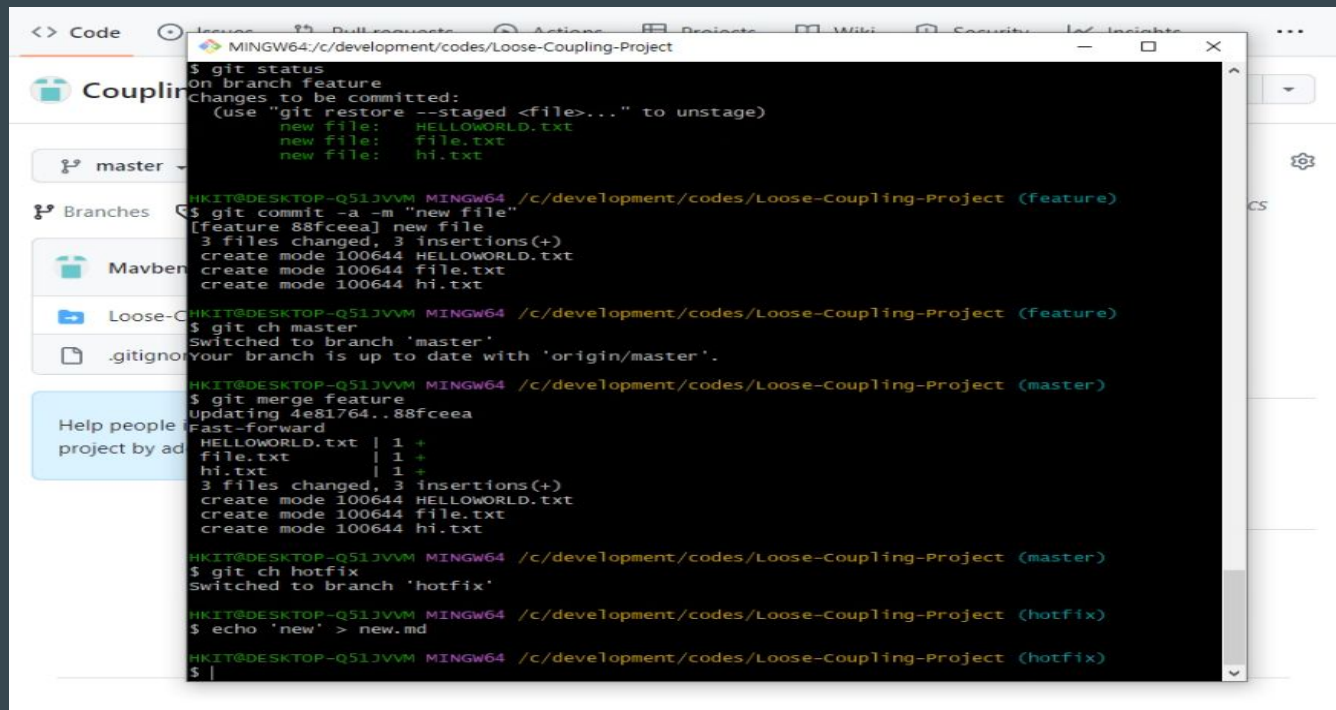
```
MINGW64/c:/development/codes/Loose-Coupling-Project
$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
$ git branch
* master
$ git checkout -b feature
Switched to a new branch 'feature'
$ git commit -a -m 'make a feature'
nothing to commit, working tree clean
$ git status
On branch feature
nothing to commit, working tree clean
$ echo 'helloworld' > HELLOWORLD.TXT
$ git commit -a -m 'make a feature'
On branch feature
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    HELLOWORLD.TXT
nothing added to commit but untracked files present (use "git add" to track)
$
```



The screenshot shows the GitHub repository page for 'Coupling-Project' (Public) with a terminal window open. The terminal displays the following commands and output:

```
MINGW64/c:/development/codes/Loose-Coupling-Project
$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
$ git merge feature
Already up to date.
$ git checkout -b hotfix
Switched to a new branch 'hotfix'
$ echo 'hi' > hi.txt
$ git commit -a -m 'make a new file'
On branch hotfix
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    HELLOWORLD.TXT
    hi.txt
nothing added to commit but untracked files present (use "git add" to track)
$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
$
```

5. fast-forward merge 수행



The screenshot shows a VS Code interface with a terminal window open. The terminal is running a series of Git commands in a MINGW64 environment. The commands and their outputs are as follows:

```
$ git status
On branch feature
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   HELLOWORLD.txt
        new file:   file.txt
        new file:   hi.txt

HKIT@DESKTOP-Q51J3VM MINGW64 /c/development/codes/Loose-Coupling-Project (feature)
$ git commit -a -m "new file"
[feature 88fcee] new file
 3 files changed, 3 insertions(+)
 create mode 100644 HELLOWORLD.txt
 create mode 100644 file.txt
 create mode 100644 hi.txt

HKIT@DESKTOP-Q51J3VM MINGW64 /c/development/codes/Loose-Coupling-Project (feature)
$ git ch master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.

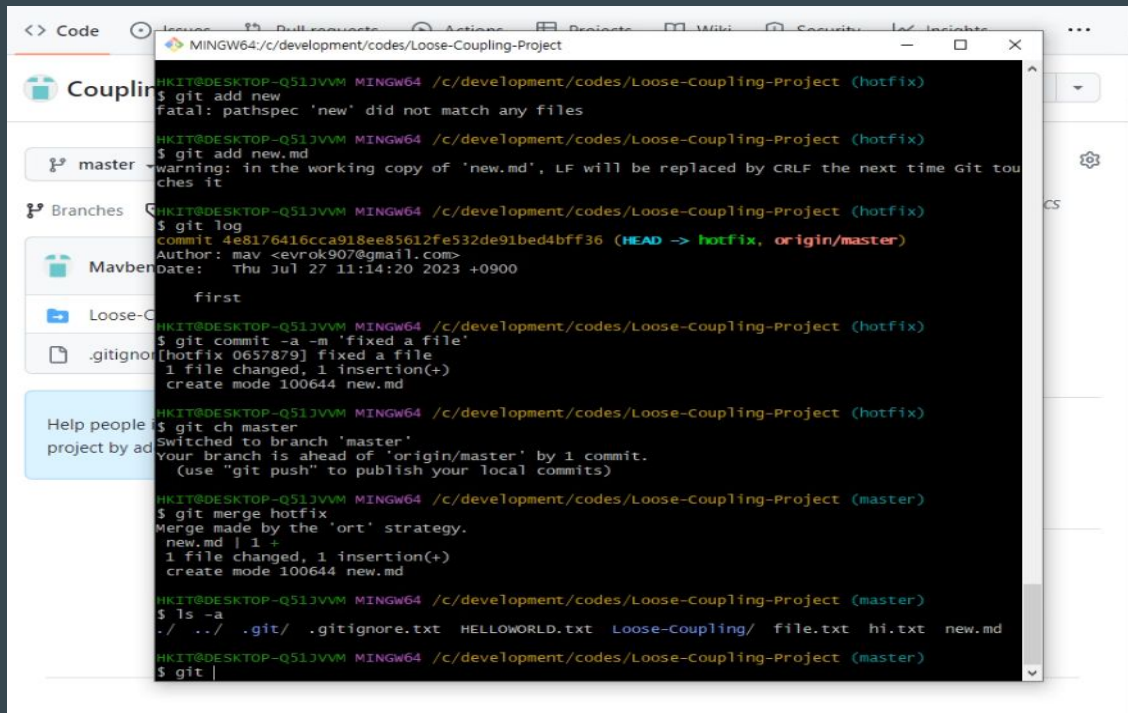
HKIT@DESKTOP-Q51J3VM MINGW64 /c/development/codes/Loose-Coupling-Project (master)
$ git merge feature
Updating 4e81764..88fcee
Fast-forward
  HELLOWORLD.txt | 1 +
   file.txt       | 1 +
   hi.txt         | 1 +
  3 files changed, 3 insertions(+)
 create mode 100644 HELLOWORLD.txt
 create mode 100644 file.txt
 create mode 100644 hi.txt

HKIT@DESKTOP-Q51J3VM MINGW64 /c/development/codes/Loose-Coupling-Project (master)
$ git ch hotfix
Switched to branch 'hotfix'

HKIT@DESKTOP-Q51J3VM MINGW64 /c/development/codes/Loose-Coupling-Project (hotfix)
$ echo 'new' > new.md

HKIT@DESKTOP-Q51J3VM MINGW64 /c/development/codes/Loose-Coupling-Project (hotfix)
$ |
```

6. 3-way merge 수행



```
MINGW64/c:/development/codes/Loose-Coupling-Project (hotfix)
$ git add new
fatal: pathspec 'new' did not match any files

MINGW64/c:/development/codes/Loose-Coupling-Project (hotfix)
$ git add new.md
warning: in the working copy of 'new.md', LF will be replaced by CRLF the next time Git touches it

MINGW64/c:/development/codes/Loose-Coupling-Project (hotfix)
$ git log
commit 4e8176416cca918ee85612fe532de91bed4bffa36 (HEAD -> hotfix, origin/master)
Author: mav <evrok907@gmail.com>
Date: Thu Jul 27 11:14:20 2023 +0900

    first

MINGW64/c:/development/codes/Loose-Coupling-Project (hotfix)
$ git commit -a -m 'fixed a file'
[hotfix 0657879] fixed a file
1 file changed, 1 insertion(+)
 create mode 100644 new.md

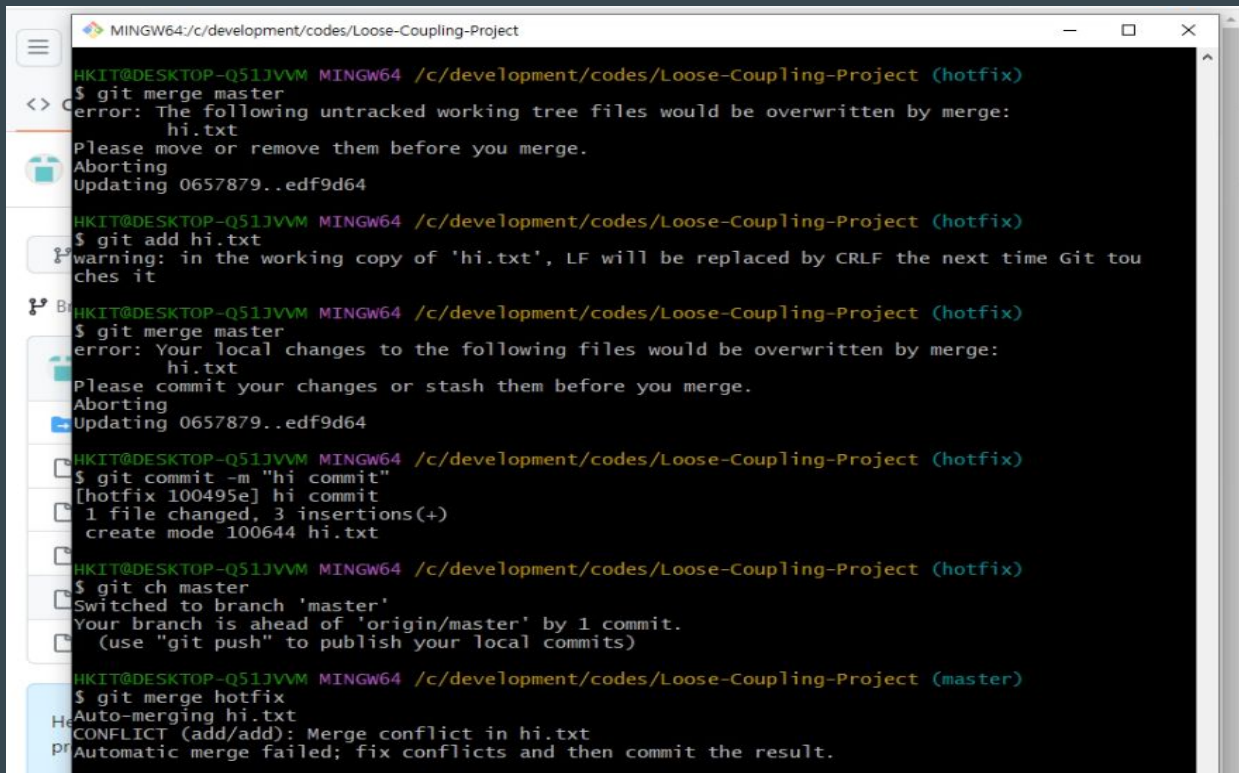
MINGW64/c:/development/codes/Loose-Coupling-Project (hotfix)
$ git ch master
Switched to branch 'master'
Your branch is ahead of 'origin/master' by 1 commit.
(use "git push" to publish your local commits)

MINGW64/c:/development/codes/Loose-Coupling-Project (master)
$ git merge hotfix
Merge made by the 'ort' strategy.
 new.md | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 new.md

MINGW64/c:/development/codes/Loose-Coupling-Project (master)
$ ls -a
./ ../ .git/ .gitignore.txt HELLOWORLD.txt Loose-Coupling/ file.txt hi.txt new.md

MINGW64/c:/development/codes/Loose-Coupling-Project (master)
$ git |
```

7.Conflict 발생 후, Merge



```
MINGW64/c:/development/codes/Loose-Coupling-Project
HKIT@DESKTOP-Q51JVVM MINGW64 /c:/development/codes/Loose-Coupling-Project (hotfix)
$ git merge master
error: The following untracked working tree files would be overwritten by merge:
    hi.txt
Please move or remove them before you merge.
Aborting
Updating 0657879..edf9d64

HKIT@DESKTOP-Q51JVVM MINGW64 /c:/development/codes/Loose-Coupling-Project (hotfix)
$ git add hi.txt
warning: in the working copy of 'hi.txt', LF will be replaced by CRLF the next time Git touches it

HKIT@DESKTOP-Q51JVVM MINGW64 /c:/development/codes/Loose-Coupling-Project (hotfix)
$ git merge master
error: Your local changes to the following files would be overwritten by merge:
    hi.txt
Please commit your changes or stash them before you merge.
Aborting
Updating 0657879..edf9d64

HKIT@DESKTOP-Q51JVVM MINGW64 /c:/development/codes/Loose-Coupling-Project (hotfix)
$ git commit -m "hi commit"
[hotfix 100495e] hi commit
1 file changed, 3 insertions(+)
create mode 100644 hi.txt

HKIT@DESKTOP-Q51JVVM MINGW64 /c:/development/codes/Loose-Coupling-Project (hotfix)
$ git ch master
Switched to branch 'master'
Your branch is ahead of 'origin/master' by 1 commit.
(use "git push" to publish your local commits)

HKIT@DESKTOP-Q51JVVM MINGW64 /c:/development/codes/Loose-Coupling-Project (master)
$ git merge hotfix
Auto-merging hi.txt
CONFLICT (add/add): Merge conflict in hi.txt
Automatic merge failed; fix conflicts and then commit the result.
```


8. 리모트 레파지토리에서 로컬 레파지토리와 동일한 코드 라인에서 다른 코드 적용(Conflict 발생) 후 Merge

```
HKIT@DESKTOP-Q51JVVM MINGW64 /c/development/codes/Loose-Coupling-Project/Loose-Coupling (main)
$ git commit -am "2nd"
[main 289a04f] 2nd
1 file changed, 1 insertion(+), 1 deletion(-)

HKIT@DESKTOP-Q51JVVM MINGW64 /c/development/codes/Loose-Coupling-Project/Loose-Coupling (main)
$ git pull origin main
remote: Enumerating objects: 13, done.
remote: Counting objects: 100% (13/13), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 7 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (7/7), 869 bytes | 37.00 KiB/s, done.
From https://github.com/Mavben/Loose-Coupling
* branch      main      -> FETCH_HEAD
   a554c89..51a7eae main  -> origin/main
Auto-merging src/com/test/lc/file
CONFLICT (content): Merge conflict in src/com/test/lc/file
Automatic merge failed; fix conflicts and then commit the result.

HKIT@DESKTOP-Q51JVVM MINGW64 /c/development/codes/Loose-Coupling-Project/Loose-Coupling (main|MERGIN
$ git status
On branch main
I
Your branch and 'origin/main' have diverged,
and have 1 and 1 different commits each, respectively.
(use "git pull" to merge the remote branch into yours)

You have unmerged paths.
(fix conflicts and run "git commit")
(use "git merge --abort" to abort the merge)

Unmerged paths:
  (use "git add <file>..." to mark resolution)
    both modified:   src/com/test/lc/file

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

HKIT@DESKTOP-Q51JVVM MINGW64 /c/development/codes/Loose-Coupling-Project/Loose-Coupling (main|MERGIN
$ git commit -am "merge"
[main 54ddd02] merge
```


9. 리모트 레파지토리와 로컬 레파지토리에 생성된 토픽 브랜치 확인 후 토픽 브랜치 제거

```
MINGW64/c/development/codes/Loose-Coupling-Project
From https://github.com/Mavben/Coupling-Project
* branch      master      -> FETCH_HEAD
Already up to date.

HKIT@DESKTOP-Q51J3VVM MINGW64 /c/development/codes/Loose-Coupling-Project (master)
$ git add MetadataCollector.java
fatal: pathspec 'MetadataCollector.java' did not match any files

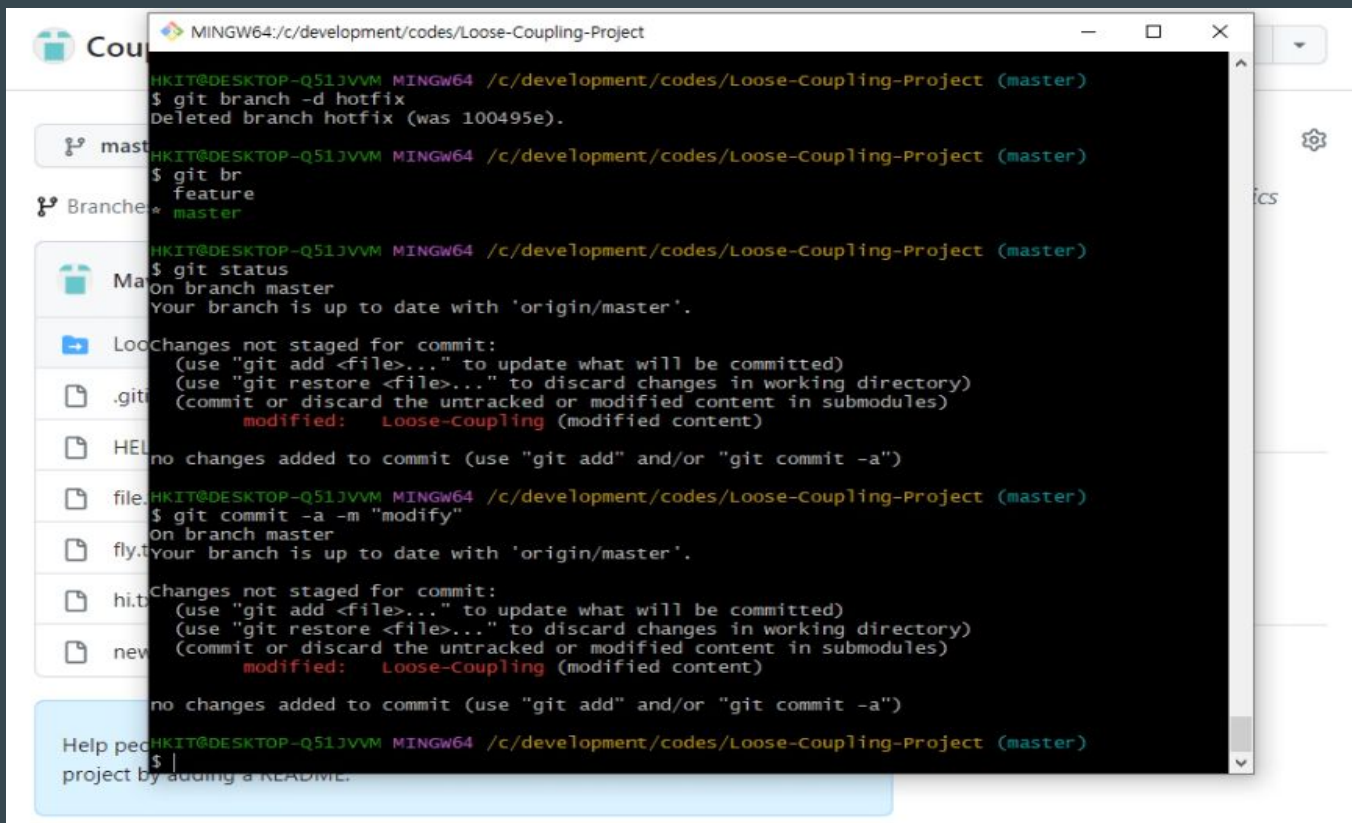
HKIT@DESKTOP-Q51J3VVM MINGW64 /c/development/codes/Loose-Coupling-Project (master)
$ git add .

HKIT@DESKTOP-Q51J3VVM MINGW64 /c/development/codes/Loose-Coupling-Project (master)
$ cd Coupling-Project
bash: cd: Coupling-Project: No such file or directory

HKIT@DESKTOP-Q51J3VVM MINGW64 /c/development/codes/Loose-Coupling-Project (master)
$ git push origin master
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 4 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (6/6), 644 bytes | 644.00 KiB/s, done.
Total 6 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/Mavben/Coupling-Project.git
   edf9d64..150801a  master -> master

HKIT@DESKTOP-Q51J3VVM MINGW64 /c/development/codes/Loose-Coupling-Project (master)
$ git branch -d hotfix
Deleted branch hotfix (was 100495e).
```

10. 로컬 레파지토리에서 프로젝트 업데이트 후 commit



```
MINGW64:/c/development/codes/Loose-Coupling-Project
HKIT@DESKTOP-Q51J3VVM MINGW64 /c/development/codes/Loose-Coupling-Project (master)
$ git branch -d hotfix
Deleted branch hotfix (was 100495e).

HKIT@DESKTOP-Q51J3VVM MINGW64 /c/development/codes/Loose-Coupling-Project (master)
$ git br
feature
* master

HKIT@DESKTOP-Q51J3VVM MINGW64 /c/development/codes/Loose-Coupling-Project (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

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)
  (commit or discard the untracked or modified content in submodules)
        modified:   Loose-Coupling (modified content)

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

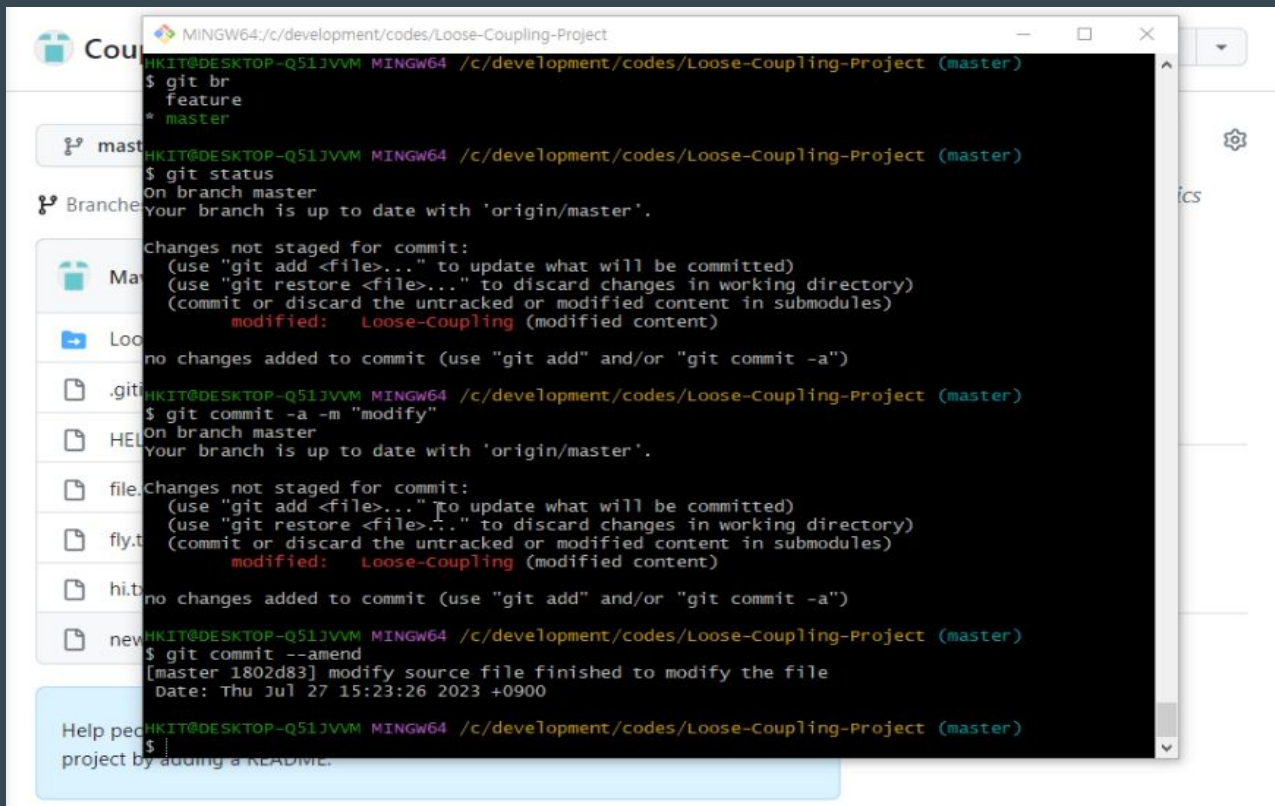
HKIT@DESKTOP-Q51J3VVM MINGW64 /c/development/codes/Loose-Coupling-Project (master)
$ git commit -a -m "modify"
On branch master
Your branch is up to date with 'origin/master'.

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)
  (commit or discard the untracked or modified content in submodules)
        modified:   Loose-Coupling (modified content)

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

HKIT@DESKTOP-Q51J3VVM MINGW64 /c/development/codes/Loose-Coupling-Project (master)
$
```

11. 10번 상황에서 commit 한 메시지 수정



```
MINGW64:/c/development/codes/Loose-Coupling-Project
HKIT@DESKTOP-Q51JVVW MINGW64 /c/development/codes/Loose-Coupling-Project (master)
$ git br
feature
* master

HKIT@DESKTOP-Q51JVVW MINGW64 /c/development/codes/Loose-Coupling-Project (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

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:   Loose-Coupling (modified content)

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

HKIT@DESKTOP-Q51JVVW MINGW64 /c/development/codes/Loose-Coupling-Project (master)
$ git commit -a -m "modify"
On branch master
Your branch is up to date with 'origin/master'.

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:   Loose-Coupling (modified content)

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

HKIT@DESKTOP-Q51JVVW MINGW64 /c/development/codes/Loose-Coupling-Project (master)
$ git commit --amend
[master 1802d83] modify source file finished to modify the file
Date: Thu Jul 27 15:23:26 2023 +0900

HKIT@DESKTOP-Q51JVVW MINGW64 /c/development/codes/Loose-Coupling-Project (master)
$
```

12. 팀 프로젝트 수행(Collaborate 또는 fork)

The screenshot shows a GitHub repository page for 'jhcode33 / Git-Practice'. The repository is public and has 1 watch, 0 forks, and 0 stars. The main branch is 'main'. The repository contains several files and folders, including 'resource', 'src/com/jeonghui/exam', '.gitignore', 'README.md', 'areum.txt', and 'yjh.txt'. The 'README.md' file is selected, showing its content. The content of 'README.md' is in Korean and describes the purpose of the repository, which is to provide a practical example of Git usage for team projects. It mentions the use of 'Collaborate' for team work and 'fork' for individual work. The repository also has a 'Releases' section with no published releases and a 'Packages' section with no published packages.

jhcode33 / Git-Practice

Type to search

Code Issues Pull requests Actions Projects Wiki Security Insights

Git-Practice Public

Watch 1 Fork 0 Star 0

main 4 branches 0 tags

Go to file Add file Code

About

Git을 학습하고, 실전에 적용할 수 있도록 실습했습니다.

git merge branch collaborate

Readme Activity 0 stars 1 watching 0 forks Report repository

Releases

No releases published Create a new release

Packages

No packages published Publish your first package

jhcode33 Merge branch 'main' of https://github.com/jhcode33/L... 4570062 4 days ago 18 commits

File	Commit Message	Time
resource	add resource & modified README.md	4 days ago
src/com/jeonghui/exam	add Losse-Coupling code	4 days ago
.gitignore	add Losse-Coupling code	4 days ago
README.md	add resource & modified README.md	4 days ago
areum.txt	add areum.txt	4 days ago
yjh.txt	add jihye.txt	4 days ago

README.md

Git 실습

Git 학습했던 내용들을 실제로 적용해 보았습니다. branch 생성, fast-forward merge, conflict merge, 3 way merge, remote branch와 local branch 충돌 해결 등의 문제를 직접해보았습니다. 마지막으로 team을 구성해 Collaborate를 진행해 각자 branch를 만들고 변경사항을 commit, push 후 팀장인 제가 remote에서 merge, local에서 merge를 진행했습니다.