

# 협업과 git

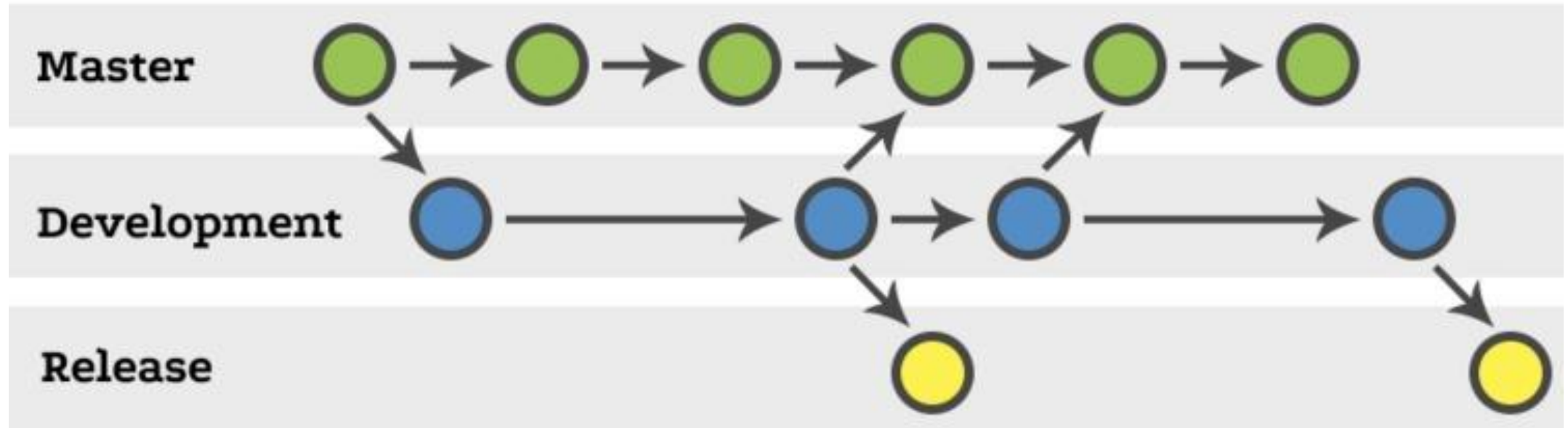
git branch와 git merge



**git**

# Git branch

# Branch!



# Branch!

- 기존 **master**의 내용을 유지하면서 새로운 작업 영역(차원)을 여는 방법입니다!
- **Branch** 를 파서 작업을 하면 해당 작업 사항은 **master** 에 전혀 영향을 끼치지 않습니다!
- **Branch**에서 원하는 기능을 개발 하다가, 완성이 되면 추후에 **Merge** 기능을 통해 **master** 로 합치면 됩니다!

# 확인 / 추가 / 이동 / 이름 변경 / 삭제

- 확인 : `git branch`
- 추가 : `git branch` 브랜치 이름
- 이동 : `git switch` 브랜치 이름 / `git checkout` 브랜치 이름
- 이름 변경 : `git branch -m` 기존브랜치이름 새로운브랜치이름
- 삭제 : `git branch -d` 브랜치 이름

새로운 git 생성

# 새로운 깃 저장소 만들기!

- 원하는 폴더를 만들어 주세요!
- 해당 폴더를 Vscode 로 열어 주세요!
- 터미널을 켜고!
- `Git init`
- `Index.html` 파일 만들고 기초 코드 넣기!
- `Git add . → git commit -m "fc"`



# 새로운 깃 브랜치 만들기!

- Git branch css
- Git branch
  - 만들어진 브랜치 확인
- Git switch css / git checkout css
- Git branch
  - 현재 브랜치 위치 확인

# 새로운 깃 브랜치에 파일 추가하기!

- Style.css 파일 추가!
- Index.html 파일에 링크도 걸기!
- 해당 내용 커밋하기!
- `Git switch master(main)` / `Git checkout master(main)`
- Master 브랜치에는 아까 변경한 사항이 전혀 반영 되어 있지 않죠?
- 이런 식으로, 각자가 담당한 부분을 작업을 하고 작업이 완료 되면 그 때에 합치면 됩니다!

# Merge

# 이제 내용을 합쳐야 겠죠?

- Merge 는 마스터에서 작업을 합니다!
- `git merge css`
- Css 브랜치의 내용이 master 로 합쳐진 것을 볼 수 있습니다!
- `Git branch`

# Conflict

# Conflict 란?

- Conflict는 merge를 하려고 할 때, master의 코드와 branch의 코드가 동일한 곳에서 다를 때 발생합니다!
- 다른 곳에서 다른 경우, 또는 삭제된 경우는 사용자가 Edit 한 것으로 판단하여 Conflict가 발생하지 않습니다.
- 동일한 곳에서 코드가 다르면 컴퓨터 입장에서는 어떤 코드가 맞는지 모르므로 해당 부분은 사용자가 처리해 줘야 합니다!

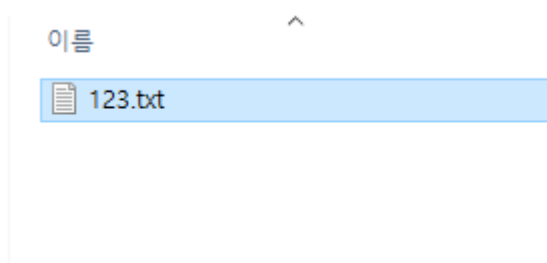
# Conflict 상황 만들기!

- 먼저 `master`의 `index.html` 페이지 제목을 `Master`로 변경해 봅시다.
- 깃 커밋하기!
- `Git switch css / git checkout css`
- `Css` 브랜치의 `index.html` 파일의 제목을 `CSS`로 변경해 봅시다
- 깃 커밋하기!
- `Git switch master / git checkout master`

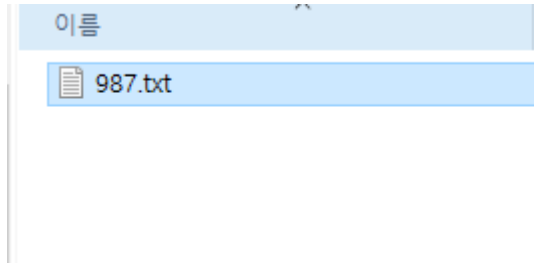
# Merge

Case1. a 브랜치와 b 브랜치에서 서로 다른 파일을 수정했을 때

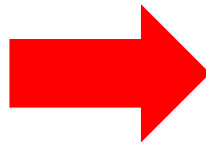
a 브랜치



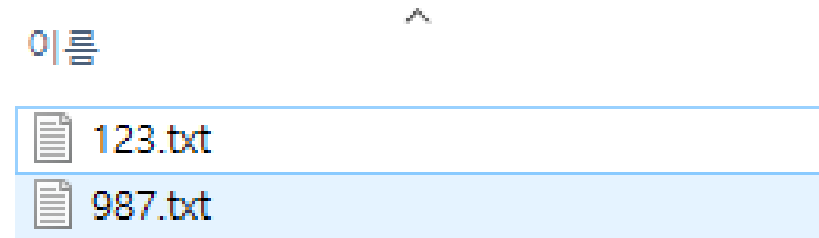
b 브랜치



```
git checkout a  
git merge b
```



a 브랜치





# Merge

Case2. 서로 같은 파일에서 다른 부분을 수정했을 때

원본

```
<html>
  <head>
    <title>원본</title>
  </head>
  <body>
    <h1>반가워요</h1>
  </body>
</html>
```

a 브랜치

```
<html>
  <head>
    <title>a에서 수정</title>
  </head>
  <body>
    <h1>반가워요</h1>
  </body>
</html>
```

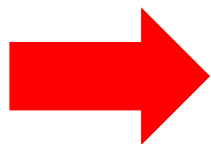
b 브랜치

```
<html>
  <head>
    <title>원본</title>
  </head>
  <body>
    <h1>b에서 수정</h1>
  </body>
</html>
```

# Merge

Case2. 서로 같은 파일이나 다른 부분을 수정했을 때

```
git checkout a  
git merge b
```



a 브랜치

```
<html>  
  <head>  
    <title>a에서 수정</title>  
  </head>  
  <body>  
    <h1>b에서 수정</h1>  
  </body>  
</html>
```

# Merge

Case3. 서로 같은 파일이고 같은 부분을 수정했을 때

원본

```
<html>
  <head>
    <title>원본</title>
  </head>
  <body>
    <h1>반가워요</h1>
  </body>
</html>
```

a 브랜치

```
<html>
  <head>
    <title>a에서 수정</title>
  </head>
  <body>
    <h1>반가워요</h1>
  </body>
</html>
```

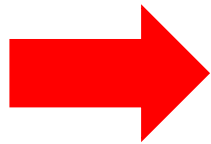
b 브랜치

```
<html>
  <head>
    <title>b에서 수정</title>
  </head>
  <body>
    <h1>반가워요</h1>
  </body>
</html>
```

# Merge

Case3. 서로 같은 파일이고 같은 부분을 수정했을 때

```
git checkout a  
git merge b
```



**Conflict**  
(충돌)

# Conflict 상황 만들기!

- Git merge css

```
tetz@DESKTOP-P7Q4OLL MINGW64 ~/Desktop/git (master)
$ git merge css
Auto-merging index.html
CONFLICT (content): Merge conflict in index.html
Automatic merge failed; fix conflicts and then commit the result.
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
Accept Current Change | Accept Incoming Change | Accept Both Changes | Compare Changes
<<<<<< HEAD (Current Change)
<title>Master</title>
=====
<title>CSS</title>
>>>>>> css (Incoming Change)
<link rel="stylesheet" href="./style.css">
</head>
```

# Conflict 상황 해결하기!

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
Accept Current Change | Accept Incoming Change | Accept Both Changes | Compare Changes
<<<<<< HEAD (Current Change)
  <title>Master</title>
=====
  <title>CSS</title>
>>>>>> css (Incoming Change)
  <link rel="stylesheet" href="./style.css">
</head>
```

- **Current Change**는 master 브랜치의 코드 상황을 보여 줍니다
- **Incoming Change**는 css 브랜치의 코드 상황을 보여 줍니다!
- 즉, 동일한 제목 부분에서 서로 코드가 달라서 conflict 가 발생!
- 이제 사용자는 어떤 코드를 사용할지 정해주면 됩니다!

# Conflict 상황 해결하기!

- 여기서는 무조건 Vscode 를 사용하는 편이 정신 건강에 좋습니다
- Git 으로 해결하려 들면, vim으로 코드 수정 해야 하거든요 ㅎㅎㅎ

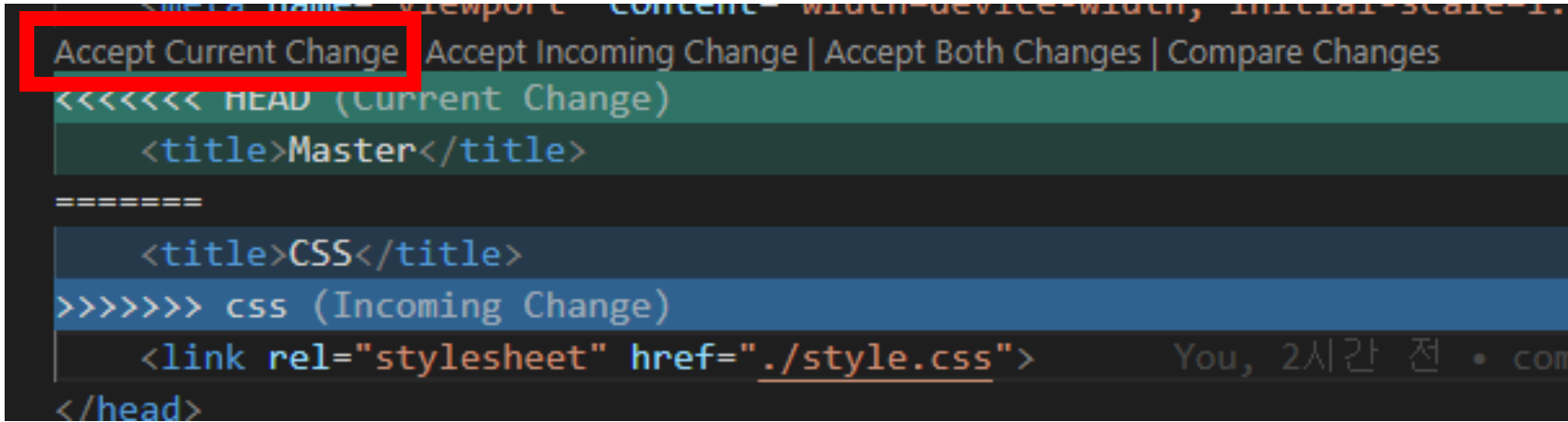
```

<meta name="viewport" content="width=device-width, initial-scale=1.0">
Accept Current Change | Accept Incoming Change | Accept Both Changes | Compare Changes
<<<<<< HEAD (Current Change)
  <title>Master</title>
=====
  <title>CSS</title>
>>>>>> css (Incoming Change)
  <link rel="stylesheet" href="./style.css">
</head>

```

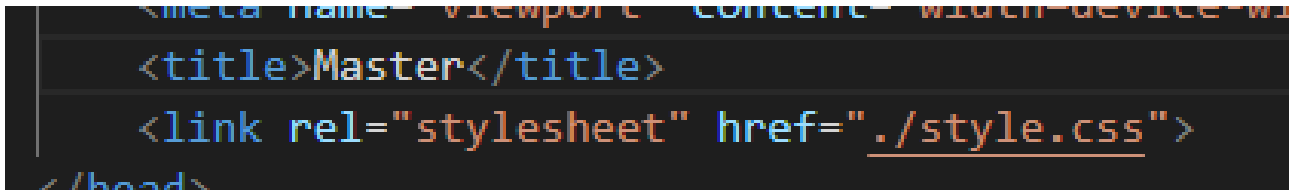
```
tecmint@TecMint ~  
~  
~  
~  
~  
~  
VIM - Vi IMproved  
  
version 8.0.3  
by Bram Moolenaar et al.  
Modified by pkg-vim-maintainers@lists.alioth.debian.org  
Vim is open source and freely distributable  
  
Help poor children in Uganda!  
type :help iccf<Enter>      for information  
  
type :q<Enter>              to exit  
type :help<Enter> or <F1>   for on-line help  
type :help version8<Enter> for version info  
  
0,0-1 All
```

# Conflict 상황 해결하기!



```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
Accept Current Change | Accept Incoming Change | Accept Both Changes | Compare Changes
<<<<<< HEAD (Current Change)
  <title>Master</title>
=====
  <title>CSS</title>
>>>>>> css (Incoming Change)
  <link rel="stylesheet" href="./style.css">
</head>
```

- Accept Current Change 를 누르면 master의 코드를 보존 합니다



```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Master</title>
<link rel="stylesheet" href="./style.css">
</head>
```



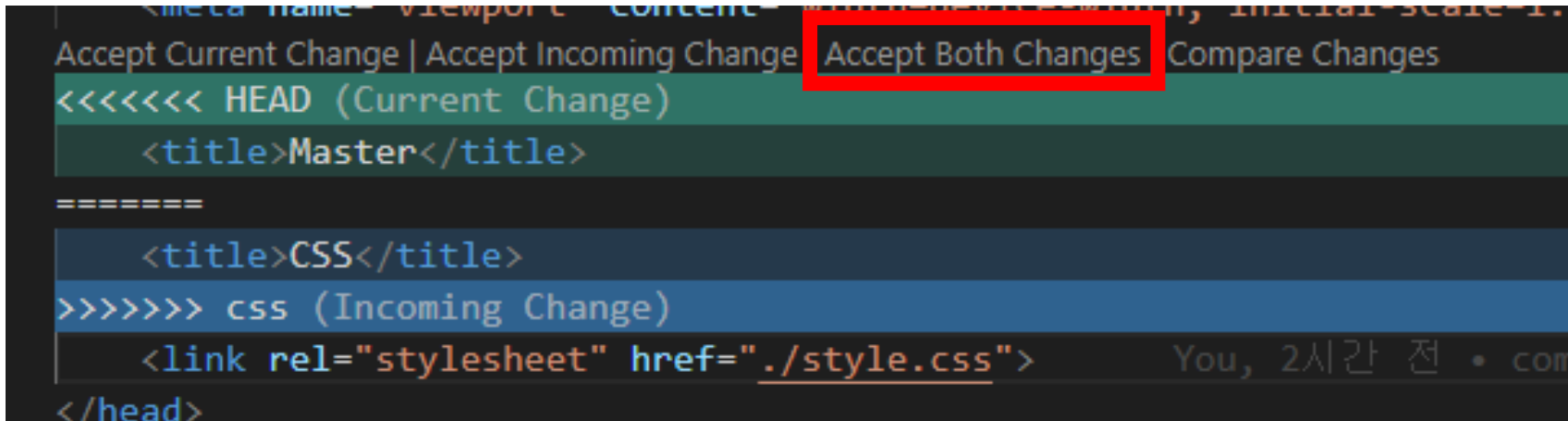
# Conflict 상황 해결하기!

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
Accept Current Change Accept Incoming Change Accept Both Changes | Compare Changes
<<<<<< HEAD (Current Change)
<title>Master</title>
=====
<title>CSS</title>
>>>>>> css (Incoming Change)
<link rel="stylesheet" href="./style.css">
</head>
```

- Accept Incoming Change 를 누르면 css 브랜치의 코드를 보존

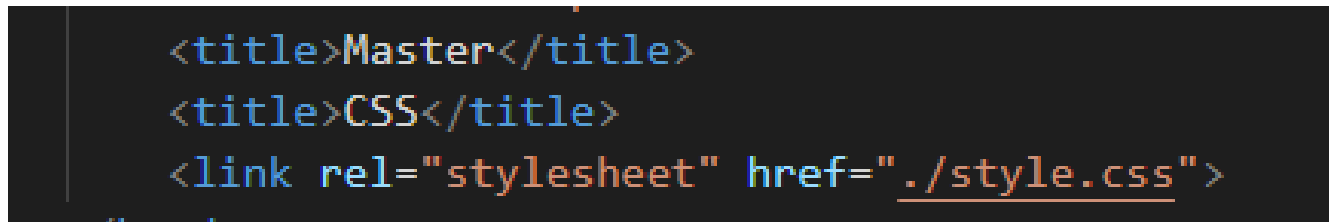
```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>CSS</title>
<link rel="stylesheet" href="./style.css">
```

# Conflict 상황 해결하기!



```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
Accept Current Change | Accept Incoming Change | Accept Both Changes | Compare Changes
<<<<<< HEAD (Current Change)
  <title>Master</title>
=====
  <title>CSS</title>
>>>>>> css (Incoming Change)
  <link rel="stylesheet" href="./style.css">
</head>
```

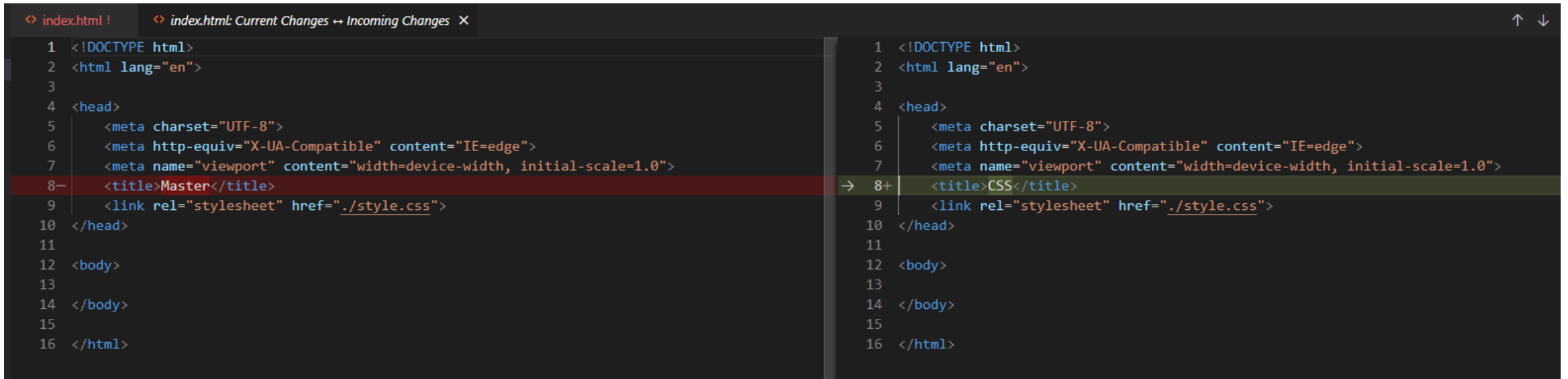
- Accept Both Change 를 누르면 master와 css의 코드를 전부 보존 합니다!



```
<title>Master</title>
<title>CSS</title>
<link rel="stylesheet" href="./style.css">
```

# Conflict 상황 해결하기!

- Compare Changes를 누르면 코드 비교 에디터가 켜집니다!
- Conflict 된 코드량이 많으면 해당 에디터가 도움이 됩니다!



The screenshot displays a code comparison interface with two side-by-side editors. The left editor, titled 'index.html !', shows the original code with line 8 containing `<title>Master</title>`. The right editor, titled 'index.html: Current Changes ↔ Incoming Changes', shows the proposed changes with line 8 containing `<title>CSS</title>`. A red horizontal bar highlights the conflict between the two versions of line 8. The rest of the code, including the DOCTYPE, lang attribute, head meta tags, link to style.css, and body tags, is identical in both versions.

```
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="UTF-8">
6   <meta http-equiv="X-UA-Compatible" content="IE=edge">
7   <meta name="viewport" content="width=device-width, initial-scale=1.0">
8-  <title>Master</title>
9   <link rel="stylesheet" href="./style.css">
10 </head>
11
12 <body>
13
14 </body>
15
16 </html>
```

```
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="UTF-8">
6   <meta http-equiv="X-UA-Compatible" content="IE=edge">
7   <meta name="viewport" content="width=device-width, initial-scale=1.0">
8+  <title>CSS</title>
9   <link rel="stylesheet" href="./style.css">
10 </head>
11
12 <body>
13
14 </body>
15
16 </html>
```

# Conflict 상황 해결하기!

- 일단 원하는 방향으로 **conflict**를 해결해 줍니다!
- 우리는 제목을 Master CSS 로 변경해 봅시다!

```
<meta name="viewport" content="width=device-width, initial-scale=1">
<title>Master CSS</title>
<link rel="stylesheet" href="style.css">
</head>
```

# Conflict 상황 해결하기!

```
tetz@DESKTOP-P7Q4OLL MINGW64 ~/Desktop/git (master|MERGING)
$
```

- 지금은 **merge** 중 이므로 커맨드 창에 **MERGING** 이라고 뜹니다!
- 코드 수정으로 끝이 아니라 수정 후, **커밋**을 해야 머징이 완료 됩니다!
- **Git add . → git commit -m “충돌 해결”**

```
tetz@DESKTOP-P7Q4OLL MINGW64 ~/Desktop/git (master)
$
```

- Merge 가 완료되고 MERGING 이라는 문구도 사라집니다!

# Conflict 상황 해결하기!

- Git branch
- 남은 css 브랜치는 삭제를 해봅시다!
- Git branch -d css
- Git branch

# 종합 실습

- JS 라는 브랜치를 만듭니다!
- JS 라는 브랜치로 이동 후, main.js 라는 파일을 만들고 alert 창으로 “!!” 라는 문구를 띄우는 코드를 삽입!
- Index.html 파일에 해당 js 파일을 연결해 줍니다!
- JS 브랜치에서 변경 사항 커밋!
- JS 브랜치의 내용을 Master 에 Merge!

# 종합 실습

- Master의 main.js에 가서 alert 창의 문구를 “??”로 변경!
- 변경 내용 커밋
- JS 브랜치로 이동하여, alert 창의 문구를 “!?”로 변경!
- 변경 내용 커밋
- Master에서 JS 브랜치를 머지!
- Conflict 상황 해결하고 머지 완료 하기!



대형 프로젝트를 위해

알아 두면 좋은 지식!

# Branch의 종류

main(master)

develop

feature

release

hotfix

# Branch - main(master)

- 제품으로 출시될 수 있는 브랜치
- 배포(Release) 이력을 관리하기 위해 사용
- 배포 가능한 상태만을 관리하는 브랜치

# Branch - develop

- 다음 출시 버전을 개발하는 브랜치
- 기능 개발을 위한 브랜치들을 병합하기 위해 사용
- 평소 개발을 진행하는 브랜치

# Branch - feature

- 기능 개발을 진행하는 브랜치
- 새로운 기능 개발 및 버그 수정을 할 때마다 'develop' 에서 분기
- 공유할 필요가 없어 로컬에서 진행 후 develop 에 merge 해 공유
- 이름 : feature/~~

```
git checkout -b feature/이름 develop
```

```
/* 개발name */
```

```
|
```

```
git checkout develop
```

```
git merge --no--ff feature/이름
```

```
git branch -d feature/이름
```

```
git push origin develop
```

# Branch - release

- 출시 버전을 준비하는 브랜치
- 배포를 위한 전용 브랜치
- 이름 : release-0.0

```
git checkout -b release-1.2 develop
```

```
/* 배포 사이클 */
```

```
git checkout master
```

```
git merge --no--ff release-1.2
```

```
git tag -1 1.2
```

```
git checkout develop
```

```
git merge --no--ff release-1.2
```

```
git branch -d release-1.2
```

# Branch - hotfix

- 출시 버전에서 발생한 버그 수정 브랜치
- 배포한 버전에 긴급하게 수정해야 할 필요가 있는 경우 사용
- Master에서 분기
- 이름 : hotfix-0.0.0

```
git checkout -b hotfix-1.2.1 master
```

```
|
```

```
/* 문제 수정 */
```

```
git checkout master
```

```
git merge --no--ff hotfix-1.2.1
```

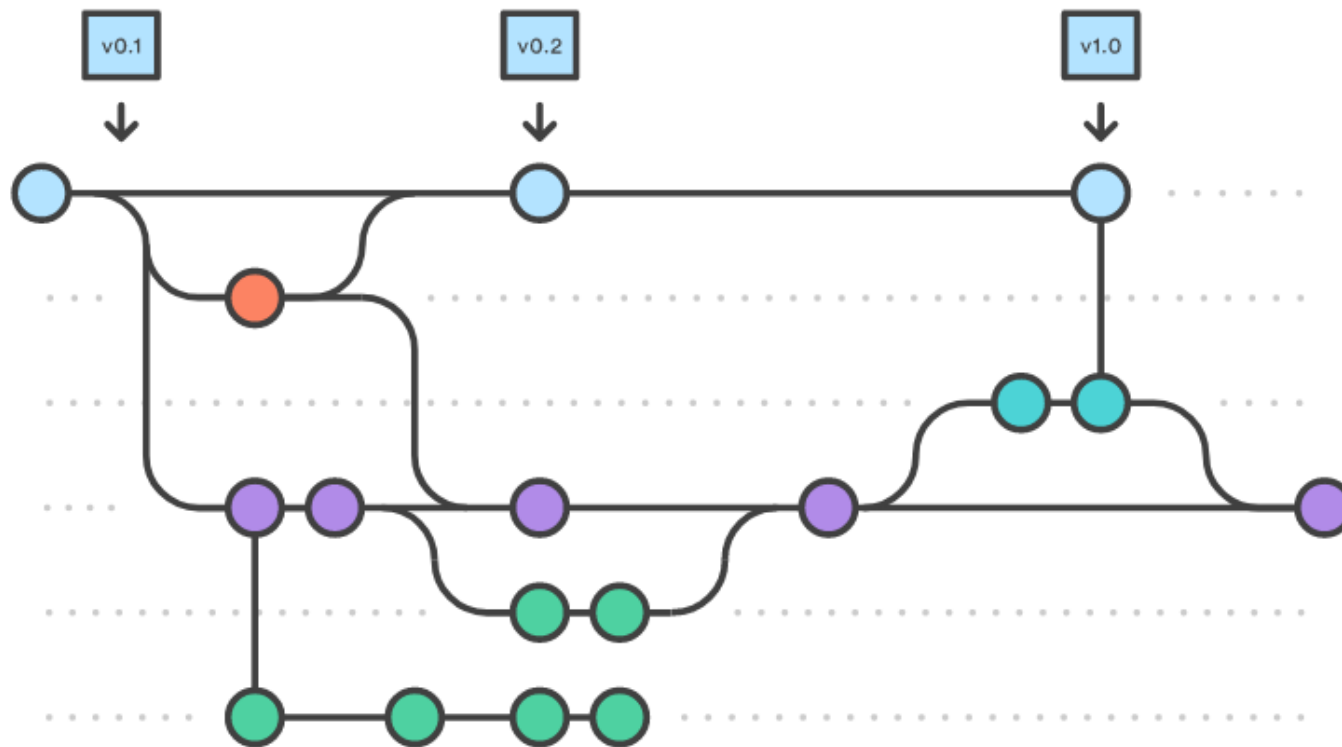
```
git tag -a 1.2.1
```

```
git checkout develop
```

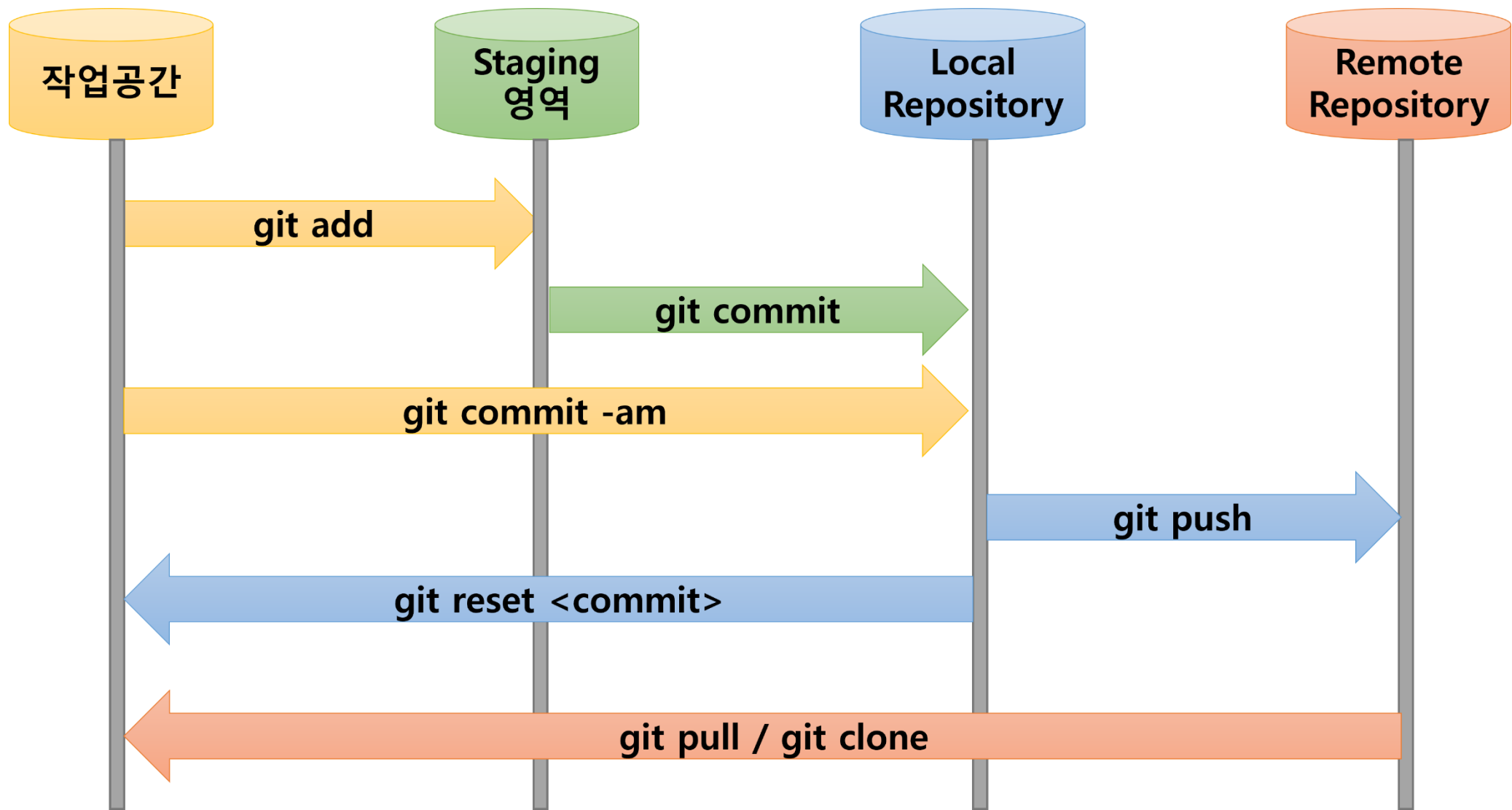
```
git merge --no--ff hotfix-1.2.1
```

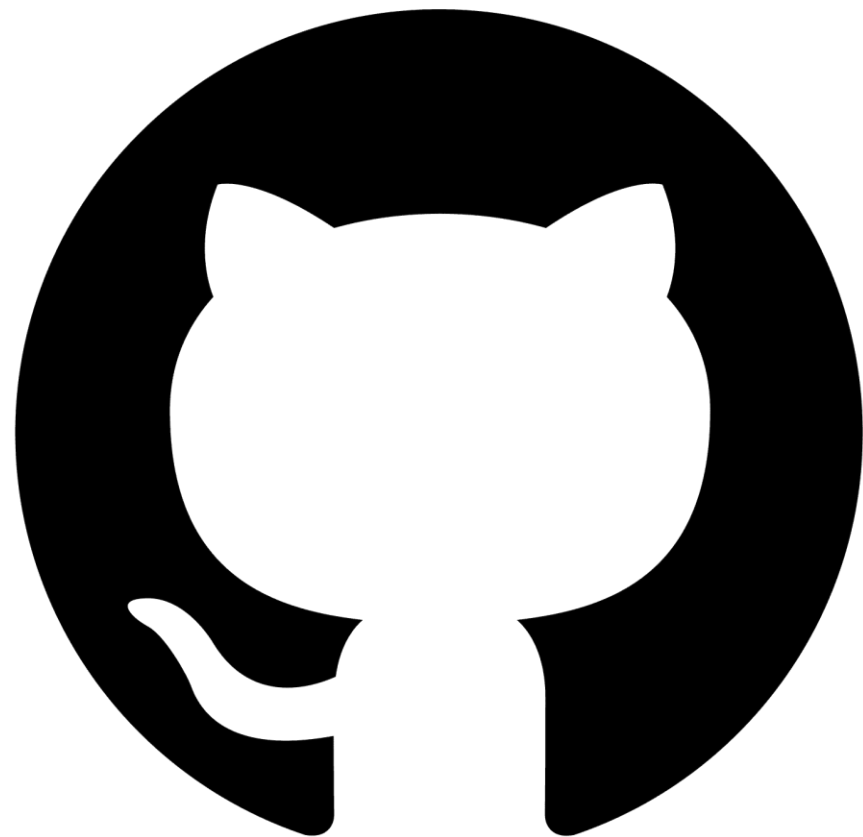
```
git branch -d hotfix-1.2.1
```

# Branch 종류





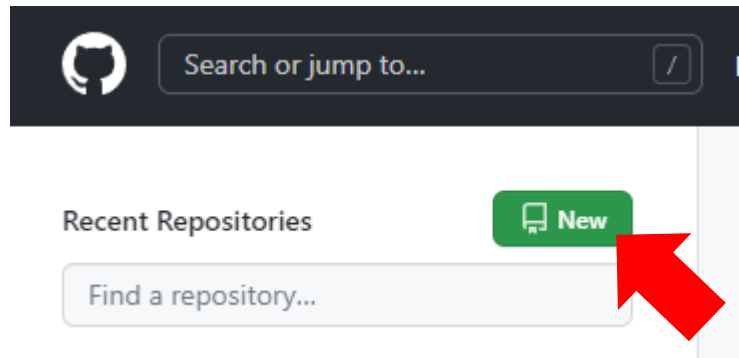




**GitHub**

# 온라인 협업을 시작해 봅시다!

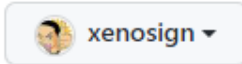
- 조장 분은 깃허브에 포트폴리오를 만들 Repo를 생성해 봅시다!



## Create a new repository

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

Owner \*




Repository name \*



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


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.

# 온라인 협업을 시작해 봅시다!

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

 Set up in Desktop

or

HTTPS

SSH

`https://github.com/xenosign/portfolio2.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 "# portfolio2" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/xenosign/portfolio2.git
git push -u origin main
```



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

```
git remote add origin https://github.com/xenosign/portfolio2.git
git branch -M main
git push -u origin main
```

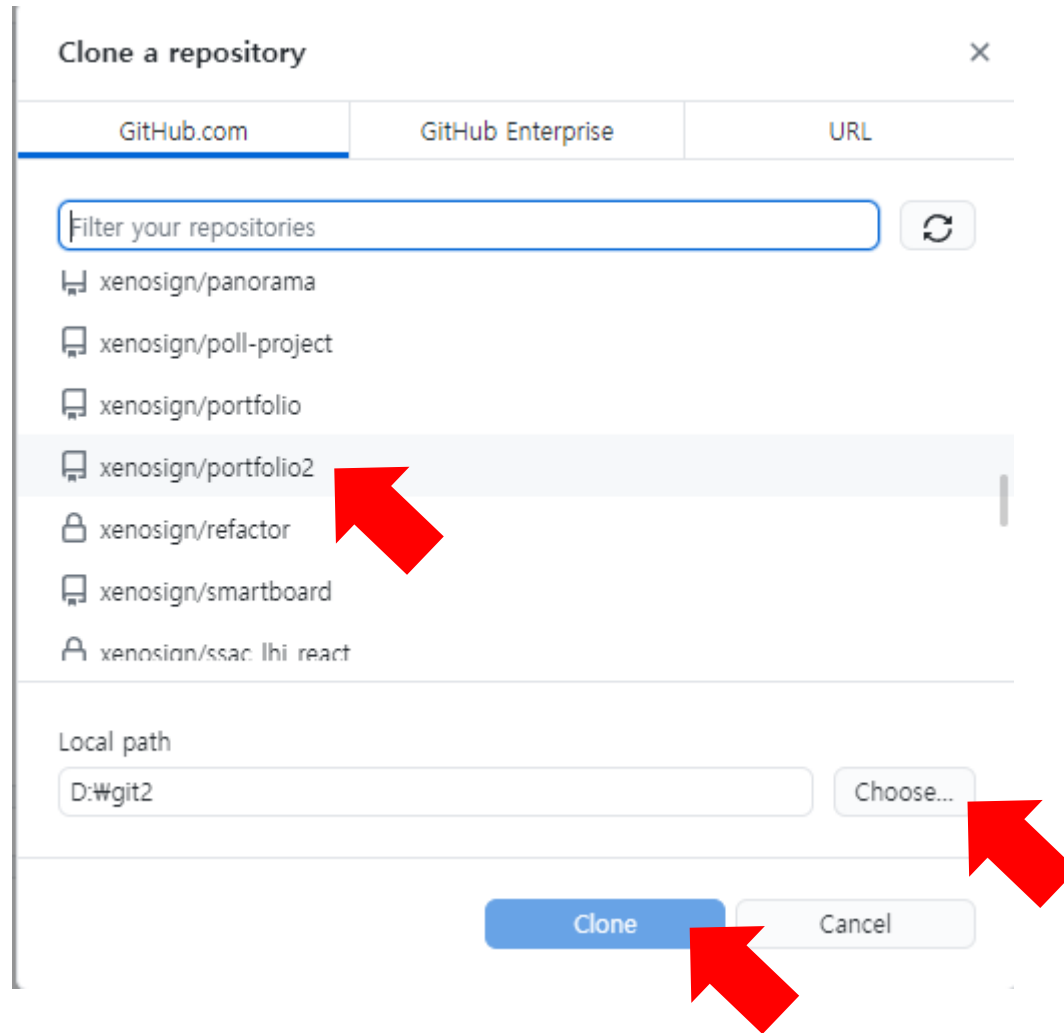
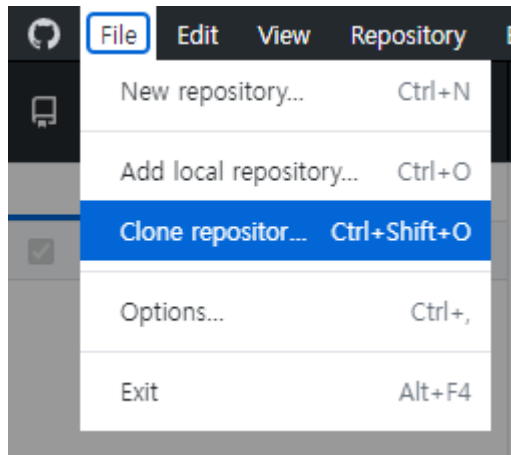


## ...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

[Import code](#)

# 온라인 협업을 시작해 봅시다!



# 새로운 깃 저장소!

- Index.html 파일 생성 → 기본 코드 생성
- 커밋
- `Git push origin main / git push --all`
- 또는 Github Desktop으로 처리!

## No local changes

There are no uncommitted changes in this repository. Here are some friendly suggestions for what to do next.



### Publish your branch

The current branch (main) hasn't been published to the remote yet. By publishing it to GitHub you can share it, open a pull request, and collaborate with others.

Always available in the toolbar or `Ctrl` `P`

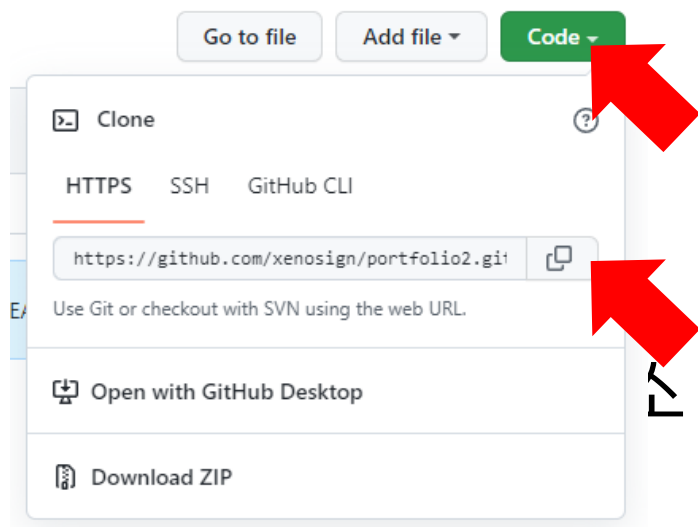
Publish branch



# 팀원 Part

# 팀원들은, 해당 repo를 clone 합시다!

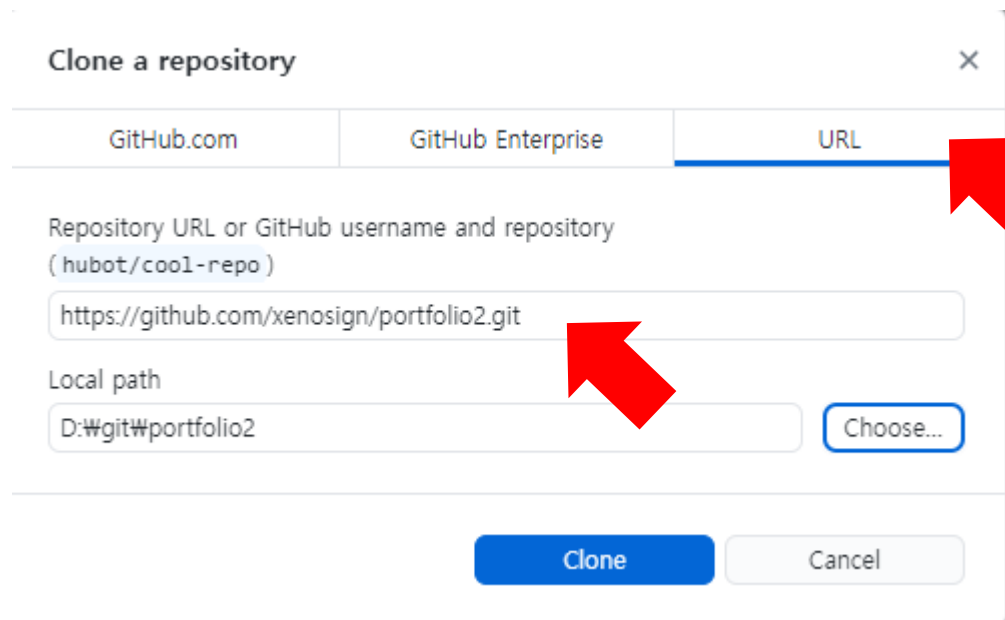
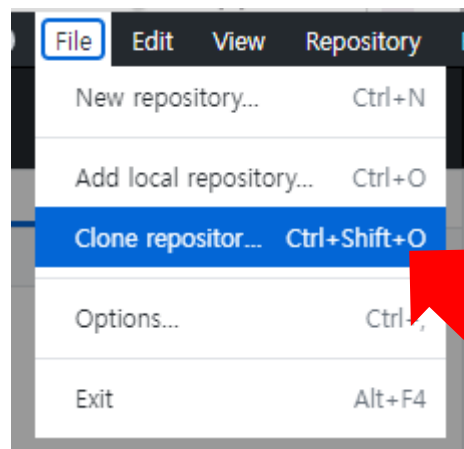
- 조장이 만든 git 주소를 받으세요! 해당 주소로 접속!





# 팀원들은, 해당 repo를 clone 합시다!

- 조장이 만든 git 주소를 받으세요! 해당 주소로 접속!



# 팀원 Part, 클론한 Repo에서!

- 새로운 브랜치 생성!
- Git branch css
- Git switch css / git checkout css
- Style.css 파일 생성!
- Reset 코드 추가!
- 커밋!
- Git push --all

ENTER

# 팀원 Part, Error 발생!

- 권한이 없기 때문입니다!
- 아무리 Public Repo라고 해도 아무나 push 하면 안되겠죠?
- Clone은 아무나 가능하지만, push는 허용된 인원만 가능합니다!
- 그럼 팀원을 초대해 봅시다!

# 조장 Part

# 조장 Part, 팀원 초대!

The screenshot shows the GitHub repository settings for 'xenosign / portfolio2' (Public). The 'Settings' tab is selected in the top navigation bar. In the left sidebar, the 'General' section is expanded, and the 'Collaborators' option is highlighted. The main content area is titled 'Manage access' and displays a message: 'You haven't invited any collaborators yet'. Below this message is a green 'Add people' button. Three red arrows are overlaid on the image: one pointing to the 'Settings' tab, one pointing to the 'Collaborators' option in the sidebar, and one pointing to the 'Add people' button.

xenosign / portfolio2 Public

<> Code Issues Pull requests Actions Projects Wiki Security Insights Settings

General

Access

Collaborators

Moderation options

Code and automation

Branches

Tags

Actions

Webhooks

Environments


Pages


Manage access

You haven't invited any collaborators yet


Add people


# 조장 Part, 팀원 초대!








Add a collaborator to portfolio2

**Tetsuro Takemoto**  
Tetz • Invite collaborator

**Tetz**  
TetzSpreatics • Invite collaborator

**tetzzz**  
Invite collaborator

**tetziida**  
Invite collaborator

**tetzerakg7**  
Invite collaborator

Select a collaborator above

## Manage access

Add people

☐ Select all

Type ▾

☐



**Tetz**

Awaiting TetzSpreatics's response

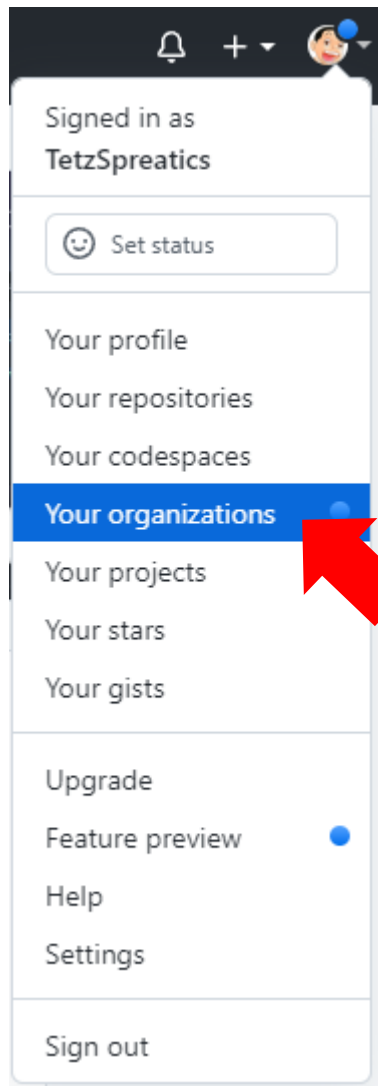
Pending Invite 

Remove

# 팀원 Part




# 팀원 Part, 초대 수락!




## Organizations

New organization

 **portfolio2** ⚠ Invitation expires in 7 days. ●

Join

Decline

 **spr-engineering** member and collaborator on 2 repositories

Leave



**xenosign** invited you to collaborate

Accept invitation

Decline

🔒 Owners of portfolio2 will be able to see:


- Your public profile information
- [Certain activity](#) within this repository
- Country of request origin
- Your access level for this repository
- Your IP address

Is this user sending spam or malicious content?

[Block xenosign](#)

# 팀원 Part, 다시 Push!

- `Git push --all`
- 그리고 깃허브로 가봅시다!

 css had recent pushes less than a minute ago

Compare & pull request



# 팀원 Part, 깃헙에서 Merge!

## Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#).

base: main ← compare: css ✓ Able to merge. These branches can be automatically merged.

css 파일 추가

Write Preview

H B I ≡ <> 🔗 ≡ ≡ ☑ @ ↻ ↶

style.css 파일 추가 및 Reset 작업!

Attach files by dragging & dropping, selecting or pasting them.

Create pull request

Reviewers

xenosign

Assignees

xenosign

Labels

None yet

Projects

None yet

Milestone

No milestone

# 팀원 Part, 깃헙에서 Merge!

- Reviewers

- 말 그대로 해당 코드를 리뷰해야 할 사람들을 추가 할 수 있습니다!

- Assignees

- 해당 코드의 Merge 여부를 결정할 사람을 지정합니다!
- 보통의 팀에서는 팀장급(또는 프로젝트 리더)이 Assignees로 등록 되며, Reviewers는 팀원들을 포함합니다!
- 그리고 merge 룰 을 정할 수 있는데요.
  - Reviewers 의 승인이 있어야만 Merge가 가능하게 등 다양하게 정할 수 있습니다.

# css 파일 추가 #1

[Edit](#)[Code](#)[Open](#)

TetzSpreatics wants to merge 1 commit into `main` from `css`

[Conversation](#) 0[Commits](#) 1[Checks](#) 0[Files changed](#) 2

+4 -0



TetzSpreatics commented now

Collaborator



style.css 파일 추가 및 Reset 작업!



css 파일 추가

5242b11



TetzSpreatics requested a review from xenosign now



TetzSpreatics assigned xenosign now

Add more commits by pushing to the `css` branch on `xenosign/portfolio2`.



Review requested

[Show all reviews](#)

Review has been requested on this pull request. It is not required to merge. [Learn more](#).



1 pending reviewer



This branch has no conflicts with the base branch

Merging can be performed automatically.

Merge pull request



You can also [open this in GitHub Desktop](#) or view [command line instructions](#).

Reviewers



xenosign



Still in progress? [Convert to draft](#)

Assignees



xenosign

Labels



None yet

Projects



None yet

Milestone



No milestone

Development



Successfully merging this pull request may close these issues.

None yet

Notifications

[Customize](#)

# 팀원 Part, 깃헙에서 Merge!

- 현재는 룰을 정하지 않아서 바로 Merge가 가능한 상황입니다!

Add more commits by pushing to the `css` branch on `xenosign/portfolio2`.



Merge pull request #1 from `xenosign/css`

css 파일 추가

Confirm merge

Cancel



# 팀원 Part, 깃헙에서 Merge!

- Merge가 완료되어 main 브랜치에 style.css 파일이 생성

main 2 branches 0 tags Go to file Add file Code

TetzSpreatics Merge pull request #1 from xenosign/css 6e6d956 19 seconds ago 3 commits

index.html	css 파일 추가	17 minutes ago
style.css	css 파일 추가	17 minutes ago

Help people interested in this repository understand your project by adding a README. Add a README

- 하지만 아직 우리 Local에는 main의 변경 사항이 반영이 안되어 있습니다!

# 팀원 Part, pull 받기!

- Vscode로 돌아 갑니다!
- Git switch main / git checkout main
- Git pull --all
- 변경 사항 반영 완료!



# 팀원 Part, pull 받기!

The screenshot shows the Visual Studio Code interface with the following elements:

- Menu Bar:** File, Edit, View, Repository, Branch, Help.
- Toolbar:** Current repository: portfolio2, Current branch: main, Pull origin (Last fetched 1 minute ago, 2 ↓).
- Changes Panel:** 0 changed files.
- Main Area:**
  - No local changes:** There are no uncommitted changes in this repository. Here are some friendly suggestions for what to do next.
  - Pull 2 commits from the origin remote:** The current branch (main) has commits on GitHub that do not exist on your machine. Always available in the toolbar when there are remote changes or `Ctrl Shift P`. A red arrow points to the **Pull origin** button.
  - Open the repository in your external editor:** Select your editor in [Options](#). Repository menu or `Ctrl Shift A`. Button: **Open in Visual Studio Code**.
  - View the files of your repository in Explorer:** Repository menu or `Ctrl Shift F`. Button: **Show in Explorer**.
  - Open the repository page on GitHub in your browser:** Repository menu or `Ctrl Shift G`. Button: **View on GitHub**.
- Source Control Panel:** Summary (required), Description, Commit to main.

# 조장 Part

**Main 브랜치**

**보호하기**

# 조장 Part, 브랜치 보호하기!

The screenshot shows the GitHub repository settings for 'portfolio2' (Public). The top navigation bar includes links for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. A red arrow points to the 'Settings' tab. The left sidebar lists various settings categories: General, Access, Collaborators, Moderation options, Code and automation, Branches (highlighted with a red arrow), Tags, Actions, Webhooks, Environments, Pages, and Security. The main content area is titled 'Default branch' and shows the 'main' branch as the default. Below this, the 'Branch protection rules' section displays a message: 'You haven't protected any of your branches'. It explains that protected branch rules can be used to disable force pushing, prevent deletions, and require status checks. A red arrow points to the 'Add branch protection rule' button. An illustration of a penguin is also present in the bottom right of the 'Branch protection rules' section.

xensign / portfolio2 Public

<> Code Issues Pull requests Actions Projects Wiki Security Insights Settings

General

Access

Collaborators

Moderation options

Code and automation

Branches

Tags

Actions

Webhooks

Environments

Pages

Security

### Default branch

The default branch is considered the "base" branch in your repository, against which all pull requests and code commits are automatically made, unless you specify a different branch.

main

### Branch protection rules

You haven't protected any of your branches

Define a protected branch rule to disable force pushing, prevent branches from being deleted, and optionally require status checks before merging. [Learn more about protected branches](#)

Add branch protection rule

# 조장 Part, 브랜치 보호하기!

- 최소 한명의 Reviewer가 Approve 리뷰를 남겨야만 머지가 가능하도록 설정!

The screenshot shows the GitHub 'Branches' settings page. On the left sidebar, 'Branches' is selected. The main content area shows the 'main' branch selected under 'Branch name pattern \*'. Below this, the 'Protect matching branches' section is expanded. Two checkboxes are checked: 'Require a pull request before merging' and 'Require approvals'. Under 'Require approvals', a dropdown menu is open showing 'Required number of approvals before merging: 1'. The number '1' is highlighted in blue, and a red arrow points to it. Another red arrow points to the 'main' branch name in the 'Branch name pattern' field.

Your GitHub Free plan can only enforce rules on its public repositories.

Branch name pattern \*

main

Protect matching branches

☒ **Require a pull request before merging**  
When enabled, all commits must be made to a non-protected branch that matches this rule.

☒ **Require approvals**  
When enabled, pull requests targeting a matching branch cannot be merged.

Required number of approvals before merging: 1 ▼

- ☒ 1
- ☐ 2
- ☐ 3
- ☐ 4
- ☐ 5
- ☐ 6

☐ **Require a pull request to be merged**  
When enabled, pull requests must be approved before merging.

# 팀원 Part

# 팀원 Part, 새로운 브랜치 생성!

- Git branch js
- Git switch js / git checkout js
- Main.js 파일 생성 → alert 코드 삽입
- Index.html에 main.js 링크
- 커밋 → 푸쉬

# 팀원 Part, pull request 수행

js had recent pushes 1 minute ago

Compare & pull request

main 3 branches 0 tags

Go to file Add file Code

TetzSpreatics Merge pull request #1 from xenosign/css 6e6d956 18 minutes ago 3 commits

index.html	css 파일 추가	35 minutes ago
style.css	css 파일 추가	35 minutes ago

Help people interested in this repository understand your project by adding a README.




Add a README




# 팀원 Part, pull request 수행

## Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#).










 base: main  compare: js  **Able to merge.** These branches can be automatically merged.




JS 파일 추가


Write


Preview

H B I         


main.js 파일 생성 및 링크 완료!

Attach files by dragging & dropping, selecting or pasting them. 


Create pull request 


Reviewers 


Suggestions

 xenosign [Request](#)


At least 1 approving review is required to merge this pull request.

Assignees 


 xenosign

Labels 

None yet

Projects 

None yet



# 팀원 Part, pull request 수행

## JS 파일 추가 #2

[Edit](#)[Code](#)

[Open](#) TetzSpreatics wants to merge 1 commit into `main` from `js`

Conversation 0

Commits 1

Checks 0

Files changed 2

+2 -0



TetzSpreatics commented now

Collaborator



main.js 파일 생성 및 링크 완료!



JS 파일 추가

b97c969



TetzSpreatics assigned xenosign now

Add more commits by pushing to the `js` branch on `xenosign/portfolio2`.



### Review required

At least 1 approving review is required by reviewers with write access. [Learn more.](#)



### Merging is blocked

Merging can be performed automatically with 1 approving review.



Merge pull request



You can also [open this in GitHub Desktop](#) or view [command line instructions](#).

Reviewers



Suggestions



xenosign

[Request](#)

At least 1 approving review is required to merge this pull request.

Still in progress? [Convert to draft](#)

Assignees



xenosign

Labels



None yet

Projects



None yet

Milestone



No milestone


# 조장 Part

# 조장 Part, Review 수행

Repository: [xenosign / portfolio2](#) Public

Navigation: [Code](#) [Issues](#) [Pull requests](#) 1 [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#)

Filters: [is:pr is:open](#) [Labels](#) 9 [Milestones](#) 0 [New pull request](#)

<input type="checkbox"/>	1 Open	✓ 1 Closed	Author	Label	Projects	Milestones	Reviews	Assignee	Sort
<input type="checkbox"/>	<a href="#">JS 파일 추가</a>								
	#2 opened 2 minutes ago by TetzSpreatics • Review required								

# 조장 Part, Review 수행

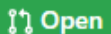
This pull request is waiting on your review.

Add your review

## JS 파일 추가 #2

Edit

<> Code



Open

TetzSpreatics wants to merge 1 commit into `main` from `js`

Conversation 0

Commits 1

Checks 0

Files changed 2

+2 -0



TetzSpreatics commented 4 minutes ago

Collaborator



main.js 파일 생성 및 링크 완료!



JS 파일 추가

b97c969



TetzSpreatics assigned xenosign 4 minutes ago



xenosign self-requested a review 31 seconds ago

Reviewers



xenosign

At least 1 approving review is required to merge this pull request.

Still in progress? Convert to draft

Assignees



xenosign

Labels



# 조장 Part, Review 수행

## JS 파일 추가 #2

Edit <> Code

Open TetzSpreatics wants to merge 1 commit into main from js

Conversation 1 Commits 1 Checks 0 Files changed 2

+2 -0

Changes from all commits File filter Conversations

0 / 2 files viewed Review changes

Filter changed files

index.html  
main.js

```
@@ -7,6 +7,7 @@
7      <meta name="viewport" content="width=device-width, initial-scale=1.0">
8      <title>Document</title>
9      <link rel="stylesheet" href="style.css">
10     + <script src="main.js"></script>
10    11    </head>
11    12
12    13    <body>
```

```
@@ -0,0 +1 @@
1 + alert("!!");
```

Finish your review

Write Preview

승인합니다!

Attach files by dragging & dropping, selecting or pasting them.

☐ Comment  
Submit general feedback without explicit approval.


☒ Approve  
Submit feedback to approve merging these changes.


☐ Request changes  
Submit feedback that must be addressed before merging.

Submit review


# 조장 Part, Merge 승인!

Add more commits by pushing to the `js` branch on `xenosign/portfolio2`.




 **Changes approved** [Show all reviewers](#)

1 approving review by reviewers with write access. [Learn more.](#)

 1 approval

▼


 **This branch has no conflicts with the base branch**

Merging can be performed automatically.

**Merge pull request**

You can also [open this in GitHub Desktop](#) or view [command line instructions](#).

Add more commits by pushing to the `js` branch on `xenosign/portfolio2`.



Merge pull request #2 from `xenosign/js`

JS 파일 추가

**Confirm merge**

Cancel

# 조장 Part, Merge 승인!

main ▾


3 branches

0 tags




Go to file

Add file ▾

Code ▾

 xenosign Merge pull request #2 from xenosign/js ...

ce63f45 19 seconds ago ⌚ 5 commits

 index.html	JS 파일 추가	11 minutes ago
 main.js	JS 파일 추가	11 minutes ago
 style.css	css 파일 추가	44 minutes ago



# 팀원 Part

# 팀원 Part, pull 받기!

The screenshot shows the Visual Studio Code interface with the following elements:

- Menu Bar:** File, Edit, View, Repository, Branch, Help.
- Toolbar:** Current repository: portfolio2, Current branch: main, Pull origin (Last fetched 1 minute ago, 2 ↓).
- Changes Panel:** 0 changed files.
- Main Area:**
  - No local changes:** There are no uncommitted changes in this repository. Here are some friendly suggestions for what to do next.
  - Pull 2 commits from the origin remote:** The current branch (main) has commits on GitHub that do not exist on your machine. Always available in the toolbar when there are remote changes or `Ctrl Shift P`. A red arrow points to the **Pull origin** button.
  - Open the repository in your external editor:** Select your editor in [Options](#). Repository menu or `Ctrl Shift A`. Button: **Open in Visual Studio Code**.
  - View the files of your repository in Explorer:** Repository menu or `Ctrl Shift F`. Button: **Show in Explorer**.
  - Open the repository page on GitHub in your browser:** Repository menu or `Ctrl Shift G`. Button: **View on GitHub**.
- Source Control Panel:** Summary (required), Description, Commit to main.

# Conflict

# 조장 Part

main

portfolio2 / index.html

Go to file

...



xenosign Update index.html

Latest commit a4ac843 now

History

1 contributor

17 lines (13 sloc) | 340 Bytes

Raw

Blame



```
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5     <meta charset="UTF-8">
6     <meta http-equiv="X-UA-Compatible" content="IE=edge">
7     <meta name="viewport" content="width=device-width, initial-scale=1.0">
8     <title>test</title>
9     <link rel="stylesheet" href="style.css">
10    <script src=""></script>
11 </head>
12
13 <body>
14
15 </body>
16
17 </html>
```



## Commit changes

Create index.html

Add an optional extended description...

- ☒ Commit directly to the `main` branch.
- ☐ Create a new branch for this commit and start a pull request. [Learn more about pull requests.](#)

Commit changes

Cancel



# 팀원 Part


# 팀원 Part, 새로운 브랜치 생성!

- Git branch conflict
- Git switch conflict / git checkout conflict
- Index.html의 제목을 conflict 로 수정!
- 커밋 → 푸쉬



## Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#).


base: **main**
←
compare: **conflict**
✖ Can't automatically merge. Don't worry, you can still create the pull request.



Update index.html

Write

Preview

**H**
**B**
*I*
 [List Icons]
 [Link Icon]
 [Check Icon]
 [Mention Icon]
 [Share Icon]
 [Undo Icon]

Leave a comment

Attach files by dragging & dropping, selecting or pasting them.



Create pull request



Reviewers



No reviews—at least 1 approving review is required.

Assignees



No one—assign yourself

Labels



None yet

Projects



None yet

Milestone




No milestone

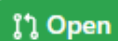
Development



Use [Closing keywords](#) in the description to automatically close issues


 Remember, contributions to this repository should follow our [GitHub Community Guidelines](#).

# Update index.html #3



Open

TetzSpreatics wants to merge 1 commit into `main` from `conflict`



Conversation 0



Commits 1



Checks 0



Files changed 1



TetzSpreatics commented now

Collaborator



No description provided.



Update index.html

6f7bf49



TetzSpreatics requested a review from **xenosign** now



TetzSpreatics assigned **xenosign** now

Add more commits by pushing to the `conflict` branch on **xenosign/portfolio2**.



**Review required**

[Show all reviewers](#)

At least 1 approving review is required by reviewers with write access. [Learn more](#).



1 pending reviewer



**This branch has conflicts that must be resolved**

Resolve conflicts

Use the [web editor](#) or the [command line](#) to resolve conflicts.

**Conflicting files**

index.html



# Update index.html #3

Resolving conflicts between `conflict` and `main` and committing changes → `conflict`

1 conflicting file

 index.html  
index.html

index.html

```
1  <!DOCTYPE html>
2  <html lang="en">
3
4  <head>
5      <meta charset="UTF-8">
6      <meta http-equiv="X-UA-Compatible" content="IE=edge">
7      <meta name="viewport" content="width=device-width, initial-scale=1.0">
8  <<<<<< conflict
9      <title>conflict</title>
10 |=====
11      <title>test</title>
12 >>>>>> main
13      <link rel="stylesheet" href="style.css">
14      <script src="main.js"></script>
15  </head>
16
17  <body>
18
19  </body>
20
21  </html>
22
```



# Update index.html #3

Resolving conflicts between `conflict` and `main` and committing changes → `conflict`

1 conflicting file

index.html

index.html

index.html

1 <!DOCTYPE html>

2 <html lang="en">

3

4 <head>

5 <meta charset="UTF-8">

6 <meta http-equiv="X-UA-Compatible" content="IE=edge">

7 <meta name="viewport" content="width=device-width, initial-scale=1.0">

8 <title>conflict</title>

9 <link rel="stylesheet" href="style.css">

10 <script src="main.js"></script>

11 </head>

12

13 <body>

14

15 </body>

16

17 </html>

18

1 conflict

Prev ^

Next v

⚙

Mark as resolved

Commit merge

✓ Resolved



---

Add more commits by pushing to the `conf1ict` branch on `xenosign/portfolio2`.



### Review required

At least 1 approving review is required by reviewers with write access. [Learn more.](#)

[Show all reviewers](#)



1 pending reviewer



### Merging is blocked

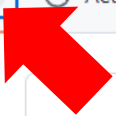
Merging can be performed automatically with 1 approving review.

Merge pull request



You can also [open this in GitHub Desktop](#) or view [command line instructions](#).

# 조장 Part



Label issues and pull requests for new contributors [Dismiss](#)

Now, GitHub will help potential first-time contributors [discover issues](#) labeled with **good first issue**

Filters 🔽  Labels 9 Milestones 0 [New pull request](#)

<input type="checkbox"/>	🔗 1 Open ✓ 2 Closed	Author ▾	Label ▾	Projects ▾	Milestones ▾	Reviews ▾	Assignee ▾	Sort ▾
<input type="checkbox"/>	<b>🔗 Update index.html</b> #3 opened 3 minutes ago by <a href="#">xenomatics</a> • Review required							





TetzSpreatics requested your review on this pull request.


Add your review

## Update index.html #3

Edit

<> Code

 Open TetzSpreatics wants to merge 2 commits into `main` from `conflict` 

 Conversation 0

 Commits 2

 Checks 0

 Files changed 1

+1 -1    



0 / 1 files viewed



Review changes

Finish your review

Write

Preview

H B I         

Leave a comment

Attach files by dragging & dropping, selecting or pasting them.



☐ Comment

Submit general feedback without explicit approval.

☒ Approve

Submit feedback and approve merging these changes.

☐ Request changes

Submit feedback that must be addressed before merging.

Submit review

Add more commits by pushing to the `conflict` branch on `xensign/portfolio2`.



### Changes approved

1 approving review by reviewers with write access. [Learn more.](#)

[Show all reviewers](#)



1 approval



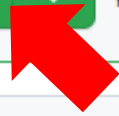
### This branch has no conflicts with the base branch

Merging can be performed automatically.

Merge pull request



You can also [open this in GitHub Desktop](#) or view [command line instructions](#).



Add more commits by pushing to the `conflict` branch on `xensign/portfolio2`.



Merge pull request #3 from `xensign/conflict`

Update index.html

Confirm merge

Cancel



# 팀원 Part

Pull !!!!!!!

성공적인 프로젝트 진행을 위해

# 반드시 지켜야 할 3가지



1. 일단 pull 받고 시작하기

2. 일단 pull 받고 시작하기

3. 일단 pull 받고 시작하기