



현업 개발자가 되기 위한 첫 걸음

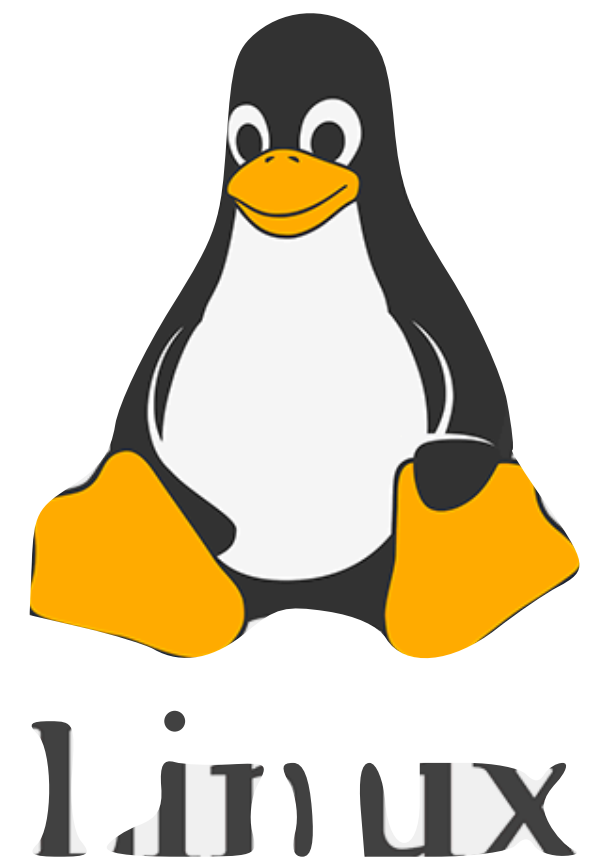
2207 김준호 2113 우준성



Git은 프로젝트 파일을 여러 개발자가 함께 작업할 때
프로젝트 관리하기 편하게 해줌
+
변경사항이 있으면
이를 추적해 관리해주는 프로그램

Git의 탄생

Git은 최초로 리눅스 토발즈가 리눅스 커널 개발에 이용하려고 개발



git프

코드를 버전 관리함으로써, 배포후 심각한 **버그 발생 시 빠르게 복구**하거나 수정된 코드만 파악하여, **버그를 빠르게 찾거나**, 한 프로젝트의 코드를 여러 사람이 **함께 협업** 할 수 있도록 도와주는 등 사용하면 많은 이점을 가짐





Repository(저장소)

저장소란 말그대로 파일이나 폴더를 저장해 두는 곳
원본 파일이 날라갔을 때 복구 가능

— 저장소는 로컬 저장소와 원격 저장소로 구분



김준호
rlawnsgh714

대소고 6기 + B1nd 6기

Edit profile

26 followers · 25 following

대구소프트웨어마이스터고등학교

Organizations



Find a repository...

Type ▾

Language ▾

Sort ▾

New

C Public

c언어 수업 들은 것들과 C언어로 문제 풀은 것들을 보관합니다

C Updated 12 days ago

Star ▾

Java Public

Java Updated 14 days ago

Star ▾

rlawnsgh714 Public

Updated 23 days ago

Star ▾

Spring Public

Updated on 24 Feb

Star ▾

WEB Public

HTML Updated on 17 Feb

Star ▾

typescript Public

typescript 공부한 것들을 모아 놓는 곳

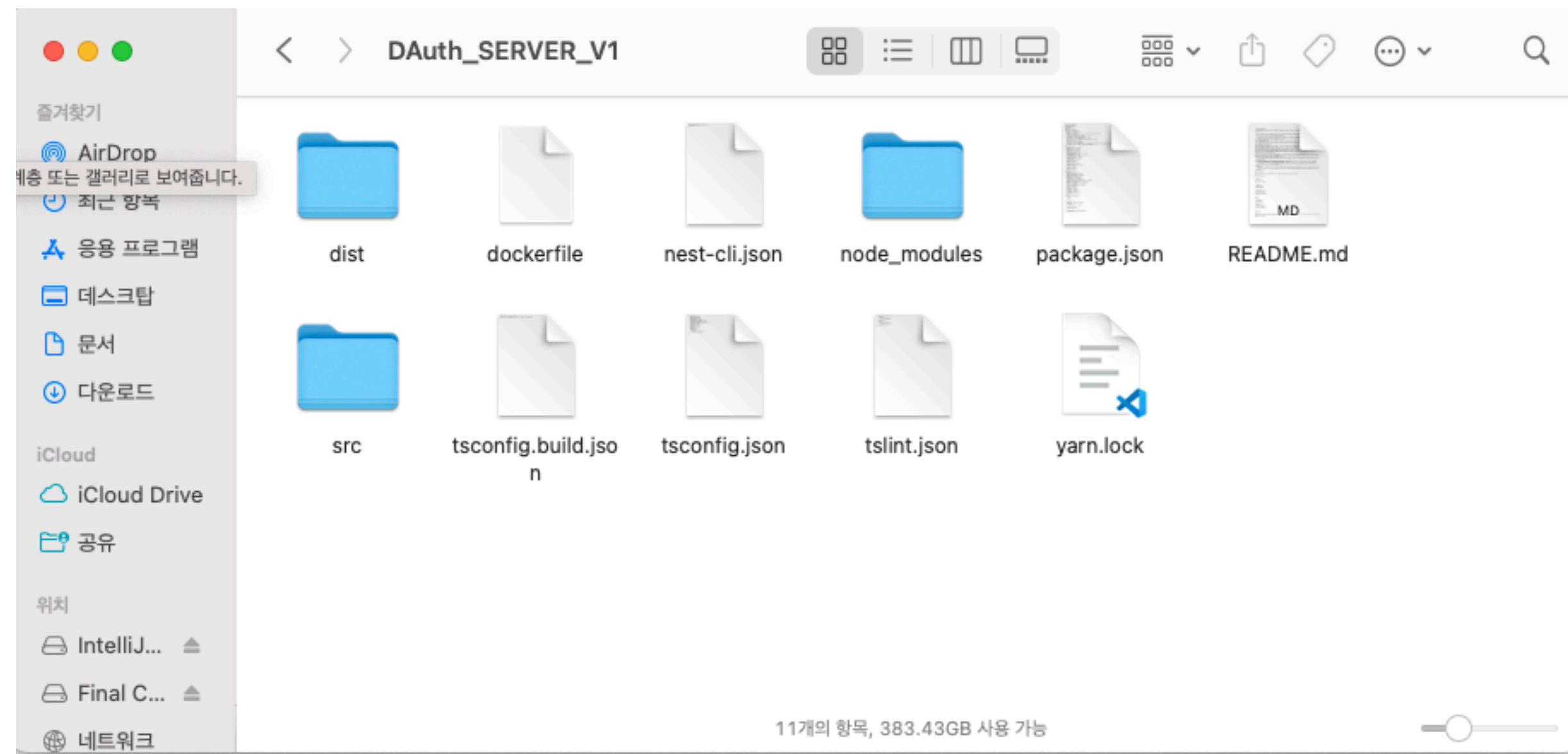
TypeScript ☆ 1 Updated on 9 Feb

Star ▾

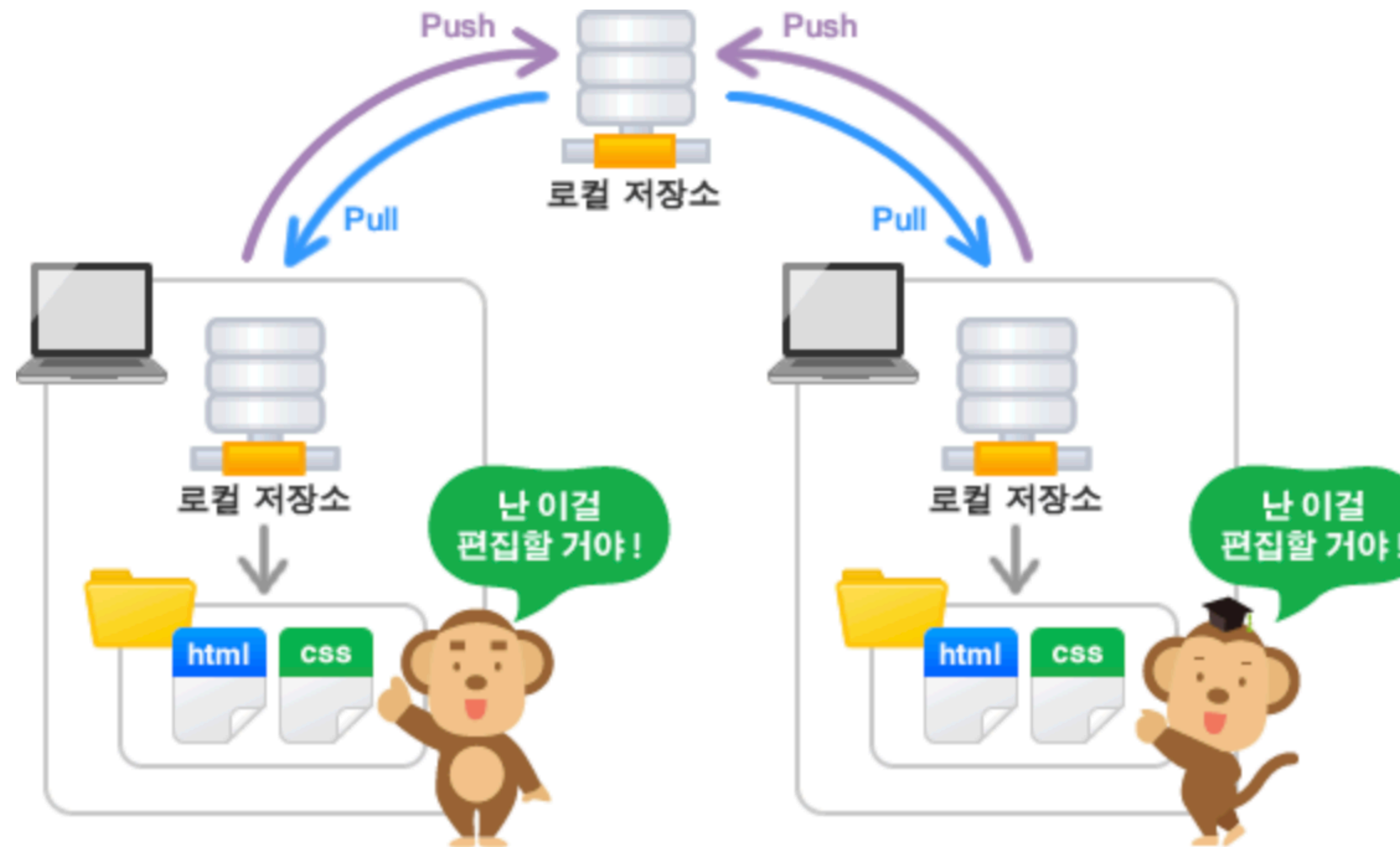
Local repository(로컬 저장소)



Local repository(로컬 저장소)



Remote repository(원격 저장소)



Remote repository(원격 저장소)

The screenshot shows a web browser window displaying the GitHub repository page for 'rlawnsgh714/Java'. The browser's address bar shows the URL 'github.com/rlawnsgh714/Java'. The repository page includes a search bar, navigation links for 'Pull requests', 'Issues', 'Marketplace', and 'Explore', and a sidebar with icons for 'Code', 'Issues', 'Pull requests', 'Actions', 'Projects', 'Wiki', 'Security', 'Insights', and 'Settings'. The main content area shows the repository's file structure, including folders like '.idea', 'java0405', 'javaclass', and 'src', and files like '.gitignore', 'Java.iml', and 'README.md'. The 'README.md' file is open, showing the title 'Java'. On the right side, there are sections for 'About', 'Releases', 'Packages', and 'Languages', with a progress bar indicating the language distribution: Java 98.8% and CSS 1.2%.

Chrome 파일 수정 보기 방문 기록 북마크 프로필 탭 창 도움말

Feature/posting 작업 by whit x Create a New Repository x Controllers | NestJS - A progre x fetch(가져오기)[원격 저장소] | 누구 x rlawnsgh714/Java x +

github.com/rlawnsgh714/Java

Search or jump to... / Pull requests Issues Marketplace Explore

rlawnsgh714 / Java Public

Unpin Unwatch 1 Fork 0 Star 0

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

master 1 branch 0 tags

Go to file Add file Code

rlawnsgh714 java 4월 5일자 코드 1814b93 14 days ago 5 commits

.idea	Initial commit	2 months ago
java0405	java 4월 5일자 코드	14 days ago
javaclass	jsp 수업정리3주차	21 days ago
src	Java-기초-문법-1	2 months ago
.gitignore	Initial commit	2 months ago
Java.iml	Initial commit	2 months ago
README.md	Create README.md	19 days ago

README.md

Java

About

No description, website, or topics provided.

Readme 0 stars 1 watching 0 forks

Releases

No releases published
[Create a new release](#)

Packages

No packages published
[Publish your first package](#)

Languages

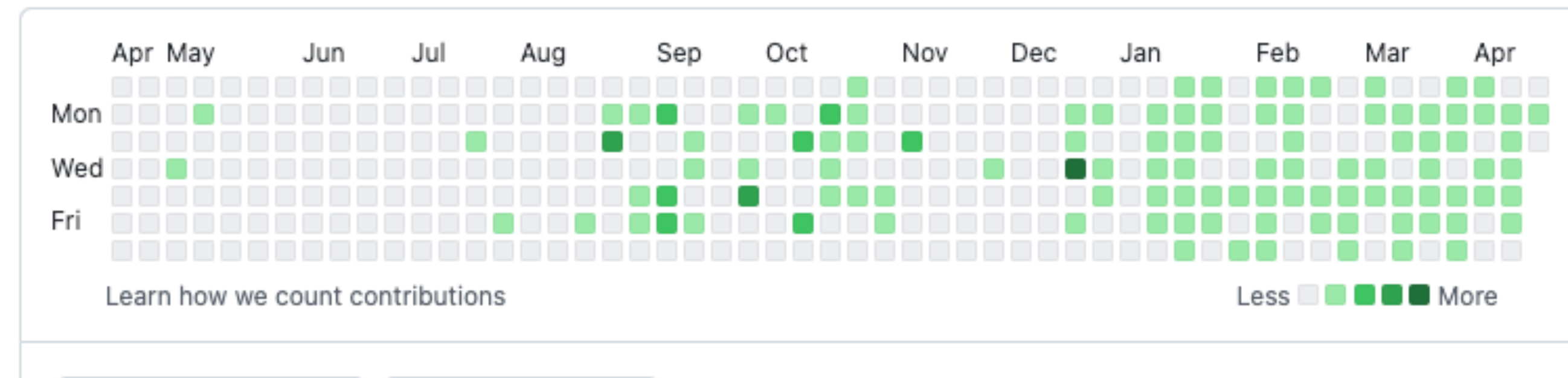
Java 98.8% CSS 1.2%

© 2022 GitHub, Inc. Terms Privacy Security Status Docs Contact GitHub Pricing API Training Blog About

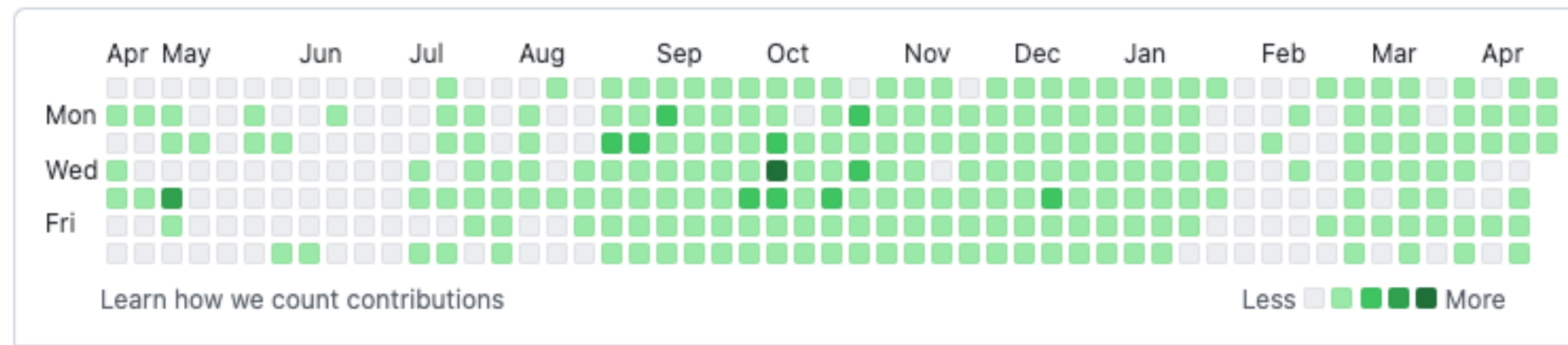
(번외) 잔디 채우기

339 contributions in the last year

Contribution settings ▼



1,324 contributions in the last year



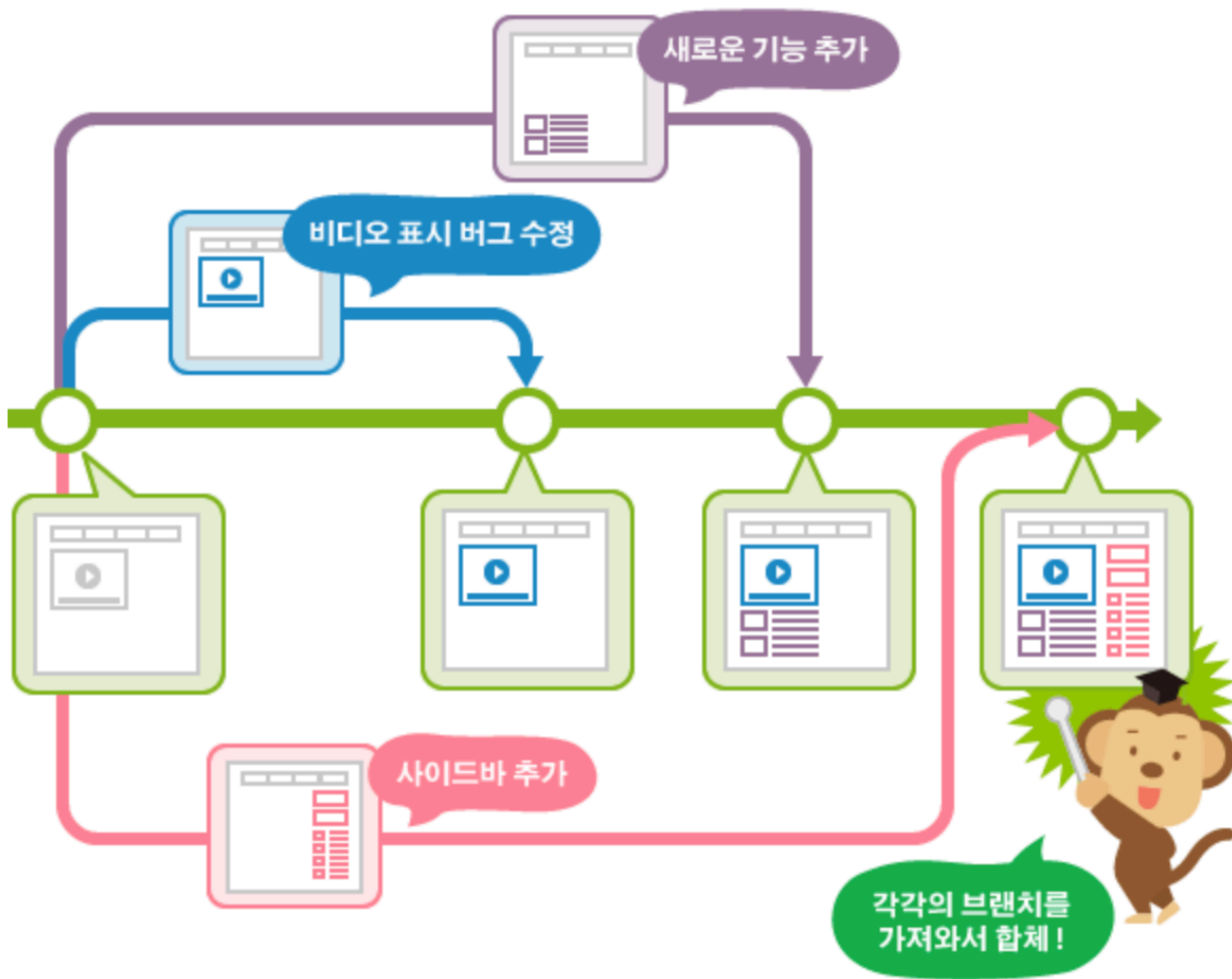
Branch(나뭇가지)



branch의 역할은 **독립적으로** 어떤 작업을 진행하기 위한 존재한다.

또한 이렇게 만들어진 브랜치는 **다른 브랜치와 병합**함으로써, 작업한 내용을 다시 새로운 하나의 브랜치로 모을 수 있다.

(협업할 때 빛을 발함. 역할 별로 코딩 + 코드 충돌 예방)



실습 go!

Git download

<https://git-scm.com>

Github

<https://github.com>

회원가입 하기

오른쪽 상단에 있는


Create repository로

원격 저장소 만들기

Create a new repository

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

Owner *

 rlawnsg714 ▾

Repository name *

저장소 이름

Great repository names are short and memorable. Need inspiration? How about **urban-waddle**?

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:

Skip this step if you're importing an existing repository.

☐

Add a README file

This is where you can write a long description for your project. [Learn more.](#)

Add .gitignore

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

.gitignore template: None ▾

Choose a license

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

License: None ▾

설명-간단히

공개 여부

저장소에 대한 설명을 기입



폴더 생성

1. Cmd 열기

2. 원하는 위치에 들어가기

ex) `cd document`

3. 원하는 위치에 `mkdir` “폴더명”

ex) `mkdir test`

git remote

repository를 만든 맨 처음에는

github의 원격저장소와 로컬저장소를 연결시켜야 한다

git remote add origin 경로명

ex) git remote add origin <https://github.com/rlawnsgh714/typescript>

git init

Repository를 생성 후 git을 사용하기 위해
git init 초기화를 해 줘야 사용이 가능하다

git add

add 는 git 파일에 해당 파일을 더하겠다는 뜻
Git add . -> 모든 파일 추가

★add 만으로는 파일에 저장이 된 것은 아니다

git commit

Commit은 파일 및 폴더의 추가/변경 사항을
저장소에 기록하기 위해 있다

Git commit -m “설명”

ex) git commit -m “add user feature”

git push

자신이 커밋한 것들을 원격 저장소로 보내는 명령어

git push origin “브랜치명”

ex) git push origin main

git pull

원격 저장소를 공유해 여러 사람이 함께 작업을 하면, 모두가 같은 원격 저장소에 푸시(Push)합니다. 그럼 다른 사람이 원격 저장소에 올려놓은(Push) 변경 내용을 내 로컬 저장소에도 적용(Pull)할 필요가 있습니다.

git pull origin “브랜치명”

ex) git pull origin main

한번에 정리1

mkdir로 파일 생성

cd 파일명

git init으로 git 초기화

git remote로 저장소와 연결

```
kimjunho — -zsh — 80x24
Last login: Tue Apr 19 15:49:12 on ttys003
[kimjunho@gimjunhoui-MacBookPro ~ % mkdir test
[kimjunho@gimjunhoui-MacBookPro ~ % git init
힌트: Using 'master' as the name for the initial branch. This default branch nam
e
힌트: is subject to change. To configure the initial branch name to use in all
힌트: of your new repositories, which will suppress this warning, call:
힌트:
힌트:   git config --global init.defaultBranch <name>
힌트:
힌트: Names commonly chosen instead of 'master' are 'main', 'trunk' and
힌트: 'development'. The just-created branch can be renamed via this command:
힌트:
힌트:   git branch -m <name>
/Users/kimjunho/.git/ 안의 빈 깃 저장소를 다시 초기화했습니다
kimjunho@gimjunhoui-MacBookPro ~ % git remote add origin rlawnsgh714/typescript
```

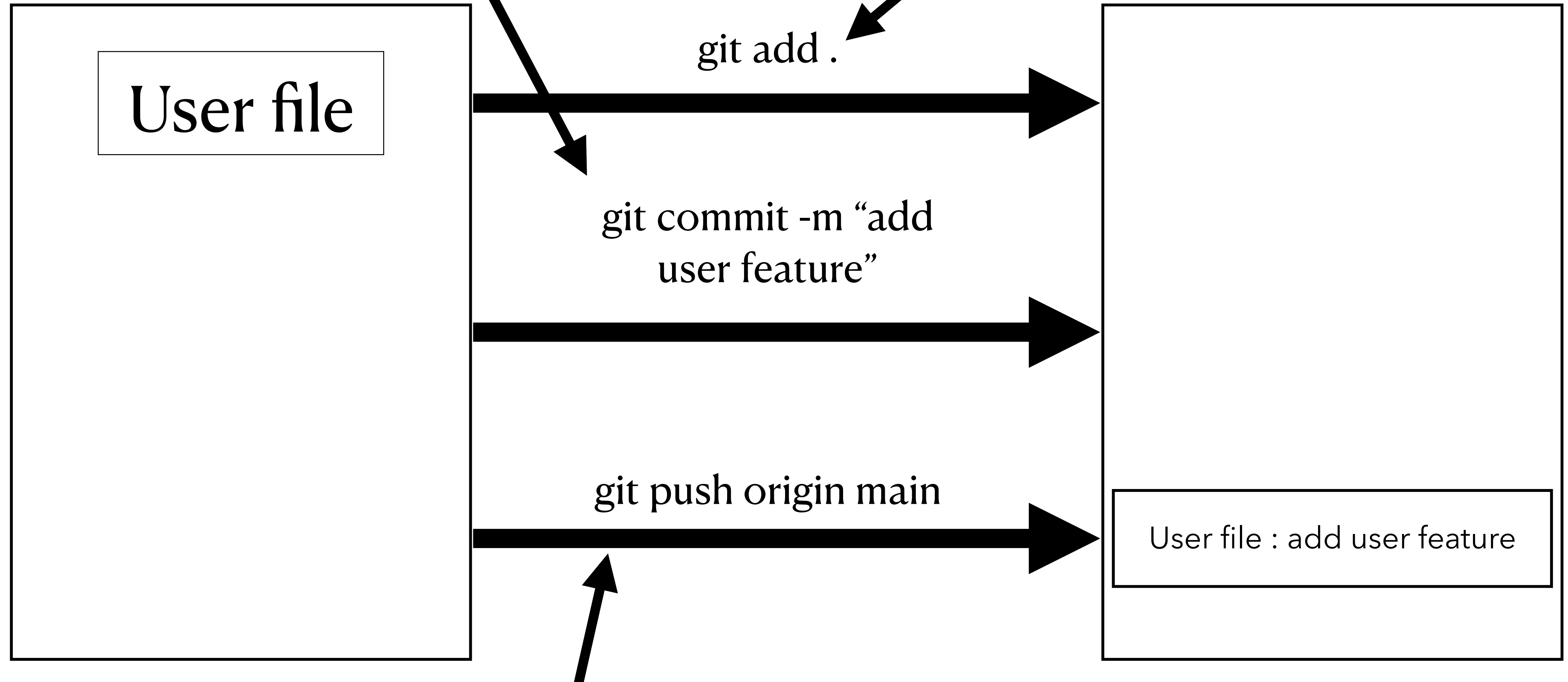
파일의 변동사항을 저장소에 기록

한번에 정리2

모든 파일을 git 파일에 추가

내 컴퓨터

원격 저장소(github)



원격 저장소로 보냄

한번에 정리3

내 컴퓨터

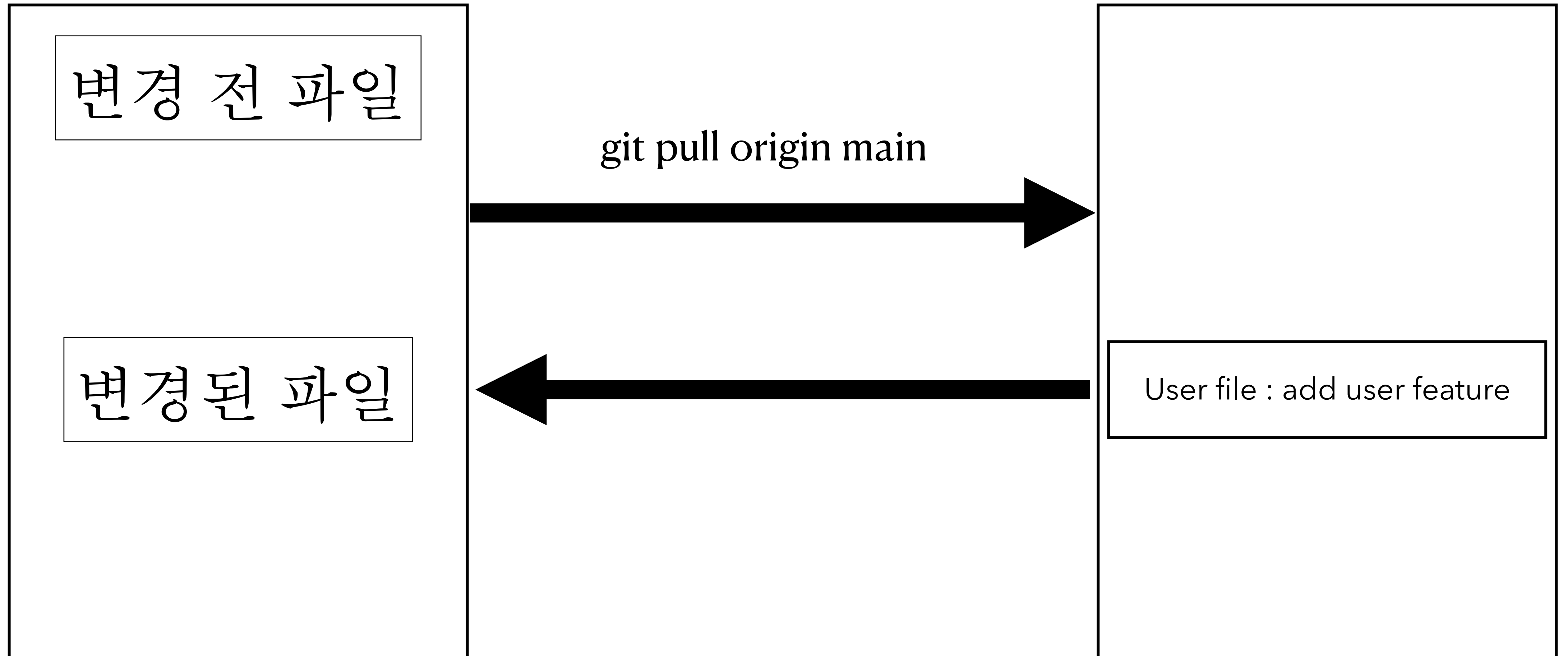
원격 저장소(github)

변경 전 파일

변경된 파일

git pull origin main

User file : add user feature






명령어 총정리!

git 명령어




분류	명령어	내용
새로운 저장소 생성 ★	\$git init	.git 하위 디렉토리 생성 폴더를 만든 후, 그 안에서 명령 실행 => 새로운 git 저장소 생성
저장소 복제/다운로드 ★	\$git clone <URI>	기존 소스 코드 복제/다운로드
	\$git clone /로컬저장소/경로	로컬 저장소 복제
	\$git clone 사용자명@호스트 :/원격저장소/경로	원격 저장소 복제

git 명령어

분류	명령어	내용
추가 및 확정	 \$git add<파일명> \$git add *	커밋의 단일 파일의 변경사항을 포함 (인덱스에 추가된 상태)
	\$git add -A	커밋의 파일 변경 사항을 한번에 모두 포함
	 \$git commit -m "커밋 메시지"	커밋 생성 (실제 변경사항 확정)
	 \$git status	파일 상태 확인

분류	명령어	내용
가지치기 작업 branch	★ \$git branch	브랜치 목록
	\$git branch <브랜치 이름>	브랜치 생성
★	\$ git checkout -b <브랜치이름>	브랜치 생성 & 이동
	\$ git checkout master	master branch로 되돌아 옴
	\$ git branch -d <브랜치이름>	브랜치 삭제
	\$ git push origin <브랜치이름>	만든 브랜치를 원격 서버에 전송
	\$ git push -u <remote > <브랜치이름>	새 브랜치를 원격 저장소로 push
	\$ git pull <remote > <브랜치이름>	원격에 저장된 git 프로젝트의 현재 상태를 다운받고 + 현재 위치한 브랜치로 병합

분류	명령어	내용
변경 사항 발행 push	★ \$ git push origin master	변경사항 원격 서버에 업로드
	★ \$ git push < remote > <브랜치이름>	커밋을 원격 서버에 업로드
	\$ git push -u < remote > <브랜치이름>	커밋을 원격 서버에 업로드
	★ \$ git remote add origin <등록된 원격 서버 주소>	클라우드 주소 등록 및 발행 (git에게 새로운 원격 서버 주소 알림)
	\$ git remote remove <등록된 클라우드 주소>	클라우드 주소 삭제

분류	명령어	내용
갱신 및 병합 merge	 \$git pull	원격 저장소의 변경 내용이 현재 디렉토리에 가져와지고(fetch) 병합(merge)됨
	 \$ git merge <다른 브랜치이름>	현재 브랜치에 다른 브랜치의 수정사항 병합
	 \$ git add <파일명>	각 파일을 병합할 수 있음
	\$ git diff <브랜치이름> <다른 브랜치이름>	변경 내용 merge 전에 바뀐 내용을 비교할 수 있음

분류	명령어	내용
태그 tag 작업	\$ git log	현재 위치한 브랜치 커밋 내용 확인 및 식별자 부여됨
로컬 변경사항 return 작업	\$ git checkout -- <파일명>	로컬의 변경 사항을 변경 전으로 되돌림
	\$ git fetch origin	원격에 저장된 git프로젝트의 현 상태를 다운로드

Pull & Fetch

Pull

- `git pull`: `git remote` 명령을 통해 서로 연결된 원격 저장소의 최신 내용을 로컬 저장소로 가져오면서 병합한다. `git push`의 반대 성격

Fetch

- `git fetch`: 로컬 저장소와 원격 저장소의 변경 사항이 다를 때 이를 비교 대조하고 `git merge` 명령어와 함께 최신 데이터를 반영하거나 충돌 문제 등을 해결한다.

선배들 Github

준호 : rlawnsgh714

준성 : whitebear05

상은 : sangau124

동현 : ldh3907

우재 : woojae05

민재 : snack655

모르는거 있으면 바인드실로 :)