

ORIENTATION

NEXT X LIKELION 2022

인사말

NEXT X 멋쟁이사자처럼 10기 합격하신 여러분들 모두 환영합니다



(엄청난 경쟁률) : 1

NEXT X 멋쟁이사자처럼 이란??

간략한 소개!

1. 누구를 위한 학회인가요?

2. 어떤 것을 배우나요?

NEXT X 멋쟁이사자처럼 이란??

간략한 소개!



1. 누구를 위한 학회인가요?

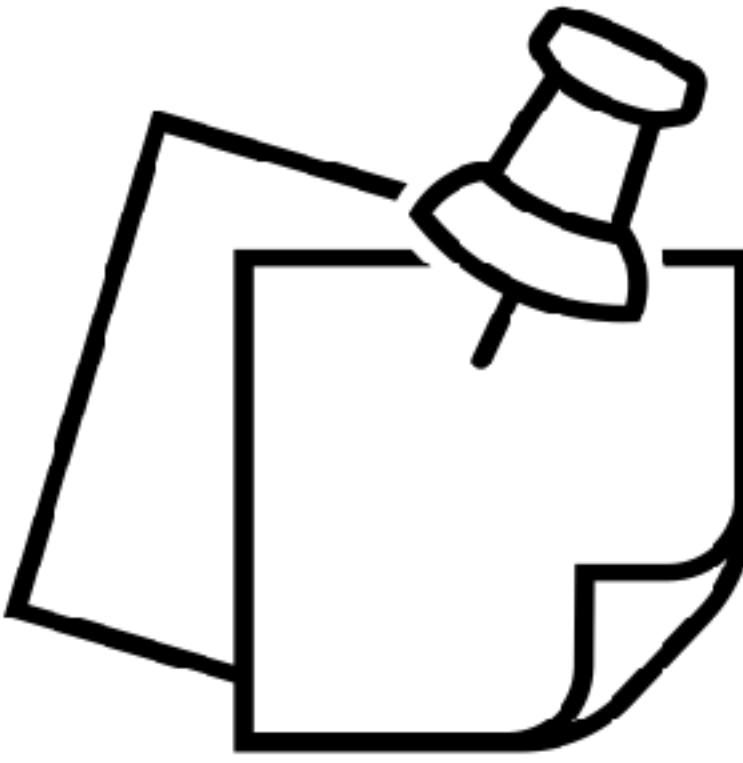
2. 어떤 것을 배우나요?

개발자..? 기업가..?

멋사에 오신 분들은?



나는 개발자..?



나는 기업가..?

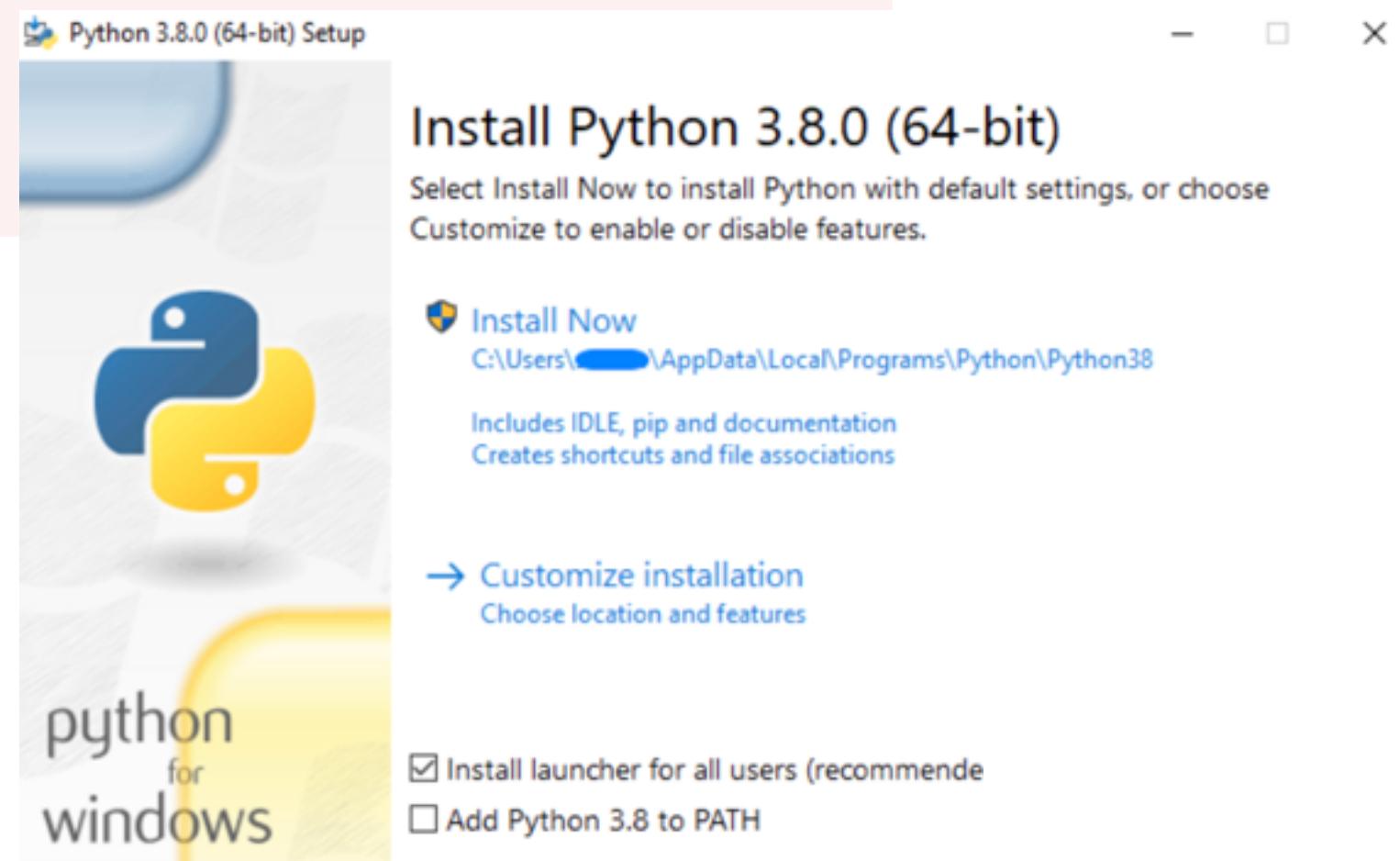
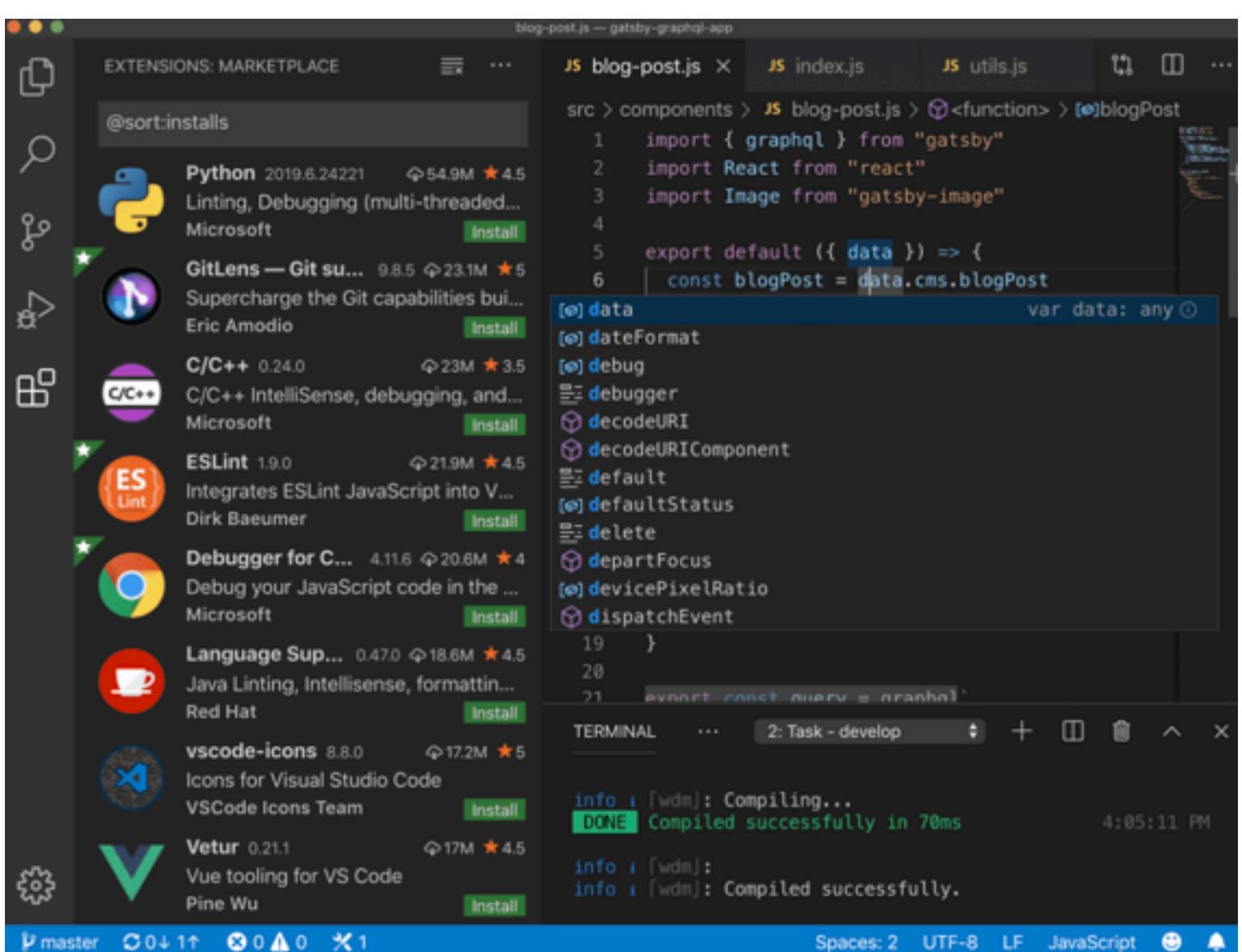
개발..진입 장벽..

개발 시작하기에는 너무 어려운 기본 세팅과 시작하자마자 마주치는 수많은 에러들

```
▶ React caught an error thrown by BuggyCounter. You should fix this error in your code. react-dom.development.js:7708
React will try to recreate this component tree from scratch using the error boundary you provided, ErrorBoundary.

Error: I crashed!

The error is located at:
in BuggyCounter (at App.js:26)
in ErrorBoundary (at App.js:21)
in div (at App.js:8)
in App (at index.js:5)
```



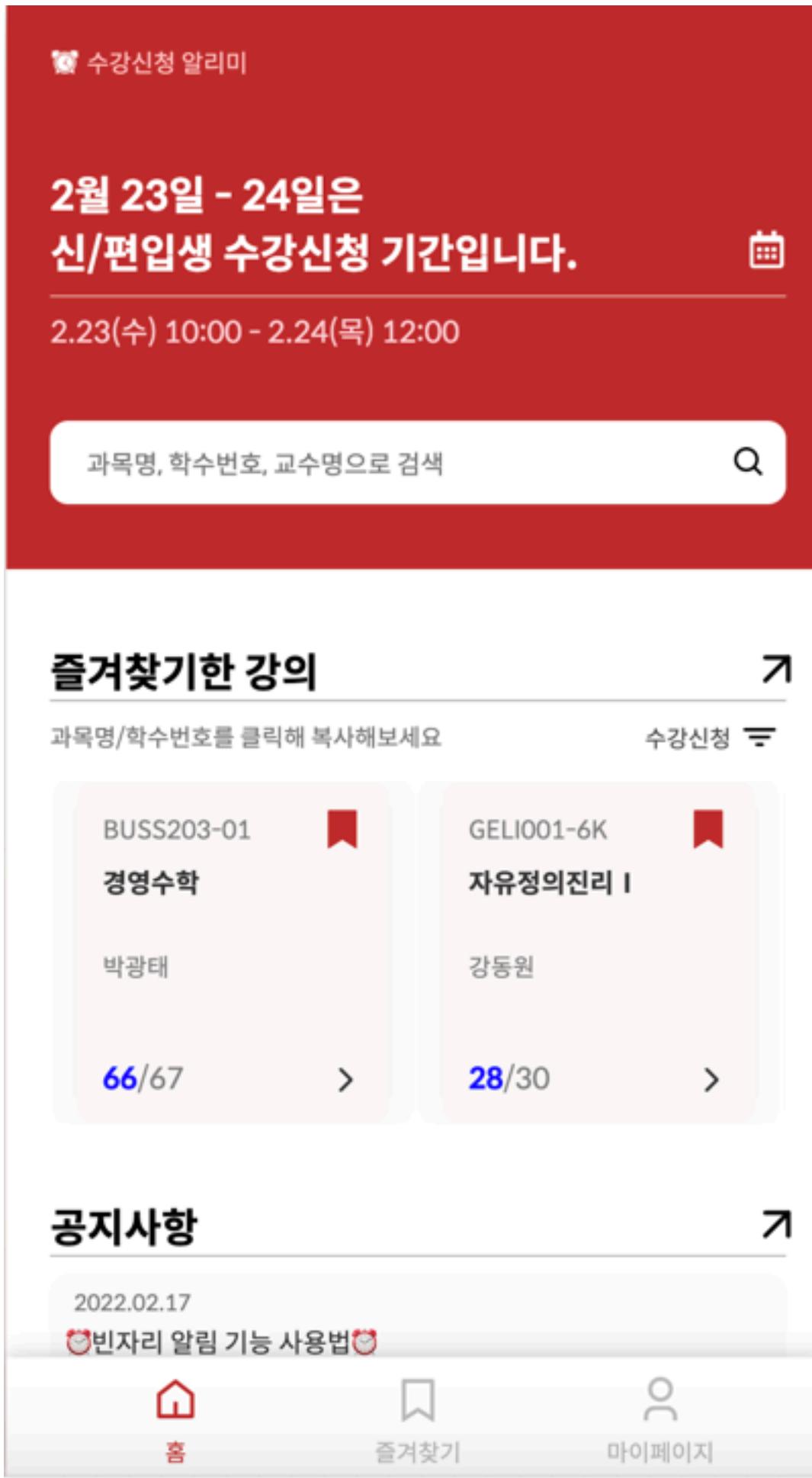
404

Not Found

The resource requested could not be found on this server!

아이디어 검증

단순 설문조사에서 벗어나자! 실제 MVP 제작을 통해 아이디어 검증 및 리뷰



NEXT X 멋쟁이사자처럼 이란??

간략한 소개!

1. 누구를 위한 학회인가요?



2. 어떤 것을 배우나요?

Session 일정

1년 동안의 계획

3월, 5월: git, html, css, python, Django, Javascript

5월 말: 자체 아이디어톤 + 해커톤 진행

여름방학: 서울대 연합 해커톤, 스페셜 세션 (블록체인 등등)

2학기: 조별 프로젝트 진행 +  NEXON MOD 해커톤

Session 장소



15일에 경영대에서 강의실 대관 관련해서 회의 후에
확정해서 연락 준다고 해서 그 전까지는 외부 세미나실
대관 형식으로 진행할 예정입니다 양해 부탁드립니다..

| 다음주 월요일 Session 장소



마루 180
주소: 강남구 역삼로 180

이것만은 꼭 지켜주세요

주의사항



질문은 구체적으로!

이거 안돼요ㅠㅠ (X)

Django DB에 문제가 있는 것
같습니다 ㅠㅠ (O)



정규 세션 결석은 2번까지



과제 미제출은 2번까지

본격적인 세션 시작!

1년동안 끝까지 화이팅합시다아



0번째 세션

Git / 파일 시스템 / 터미널

NEXT X LIKELION 전혜린

2022.03.10

아기다리 고기다리던 10기 멋사 첫 세션!-!

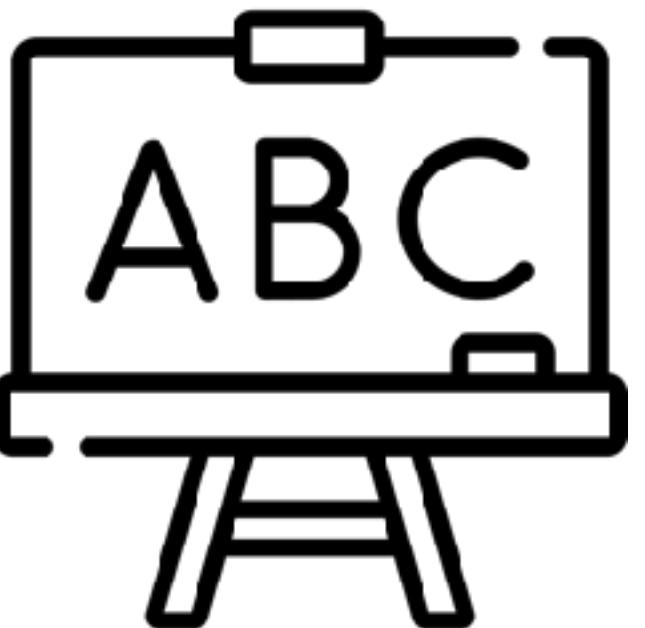
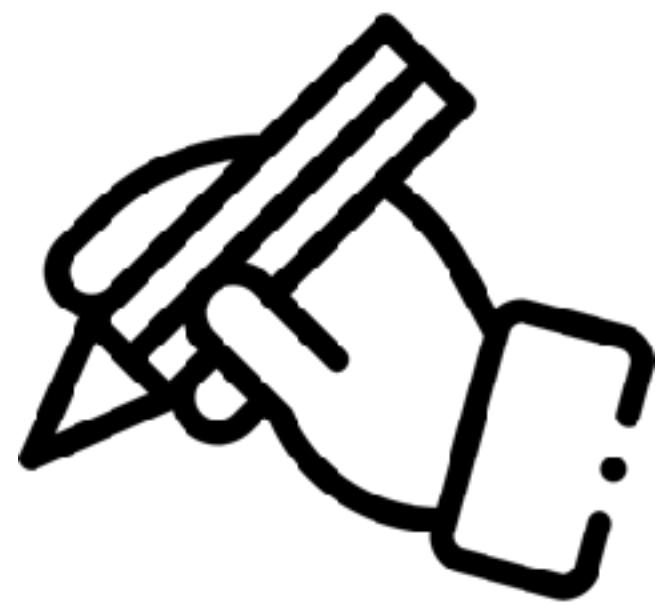
우리는 모두 멋쟁이 사자 YAY



와 개발이당-!

Intro

글 잘 쓰는 소설가가 되기 위해서?



연필을 사서 쥐고,

알파벳을 배우고,

문학을 익히고,

소설가가 된다!

Intro

코딩 잘하는 개발자가 되기 위해서?



연필을 사서 쥐고,



프로그래밍 도구

ex) Git, VSCode, etc.



알파벳을 배우고,



프로그래밍 언어

ex) Python, Javascript, etc.



문학을 익히고,



프로그래밍 지식

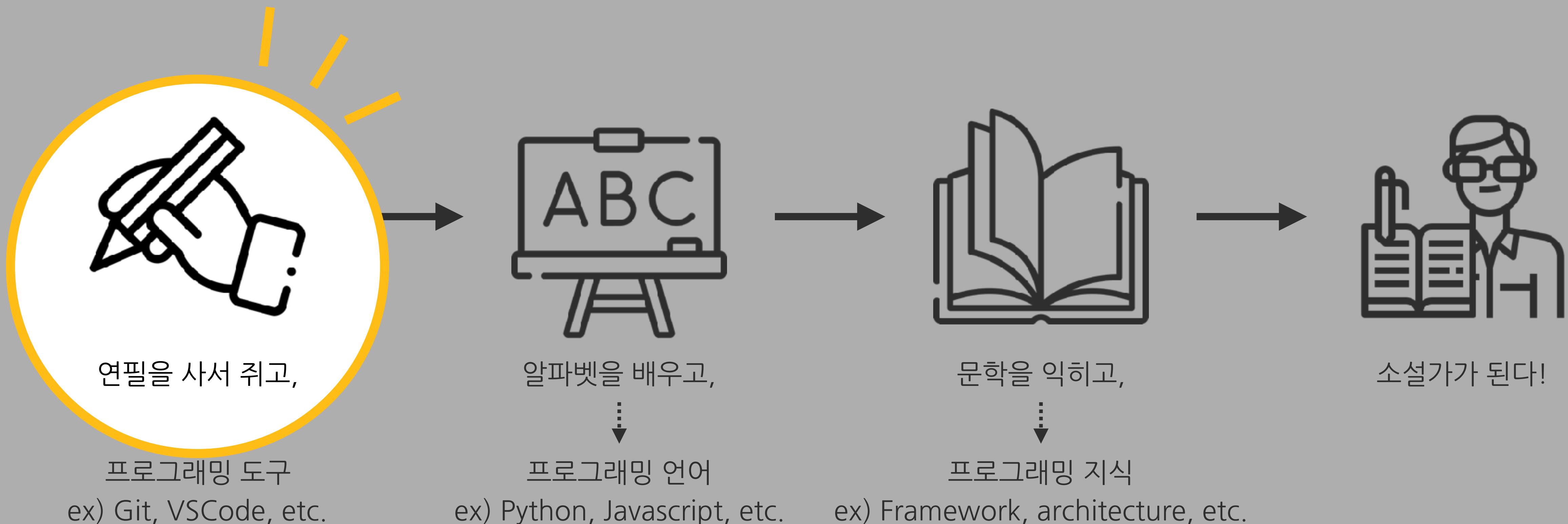
ex) Framework, architecture, etc.



소설가가 된다!

Intro

코딩 잘하는 개발자가 되기 위해서?

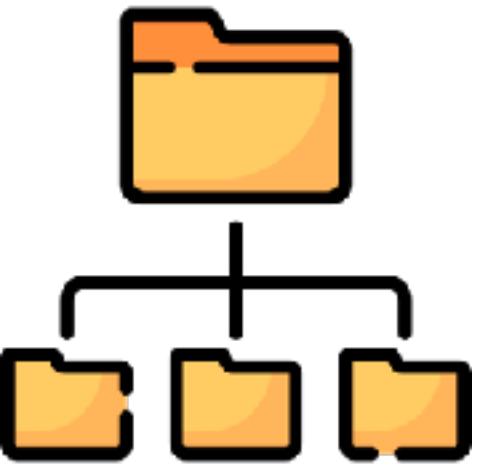


Index

Detail Curriculum



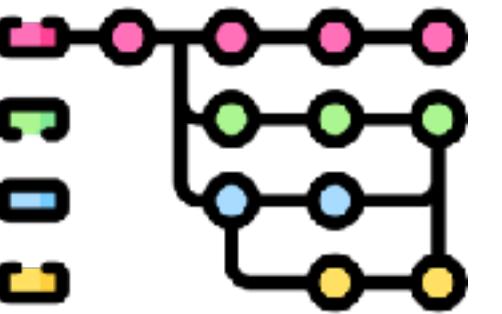
0. 프로그래밍 툴 설치 및 설정



1. FileSystem 알아보기



2. CLI 배우기



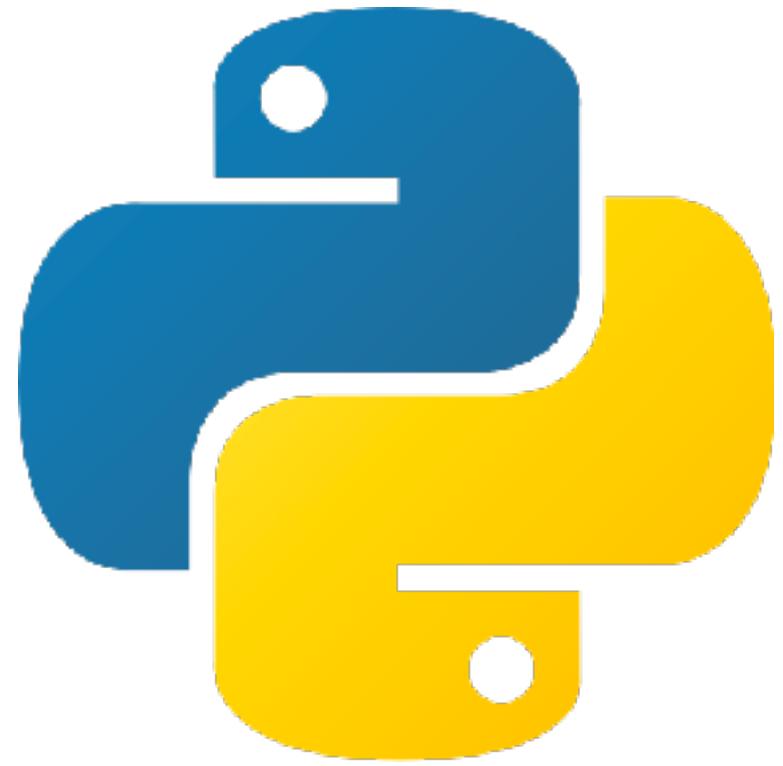
3. 로컬에서 Git 기초 버전 관리



4. Github에 푸시하기

0. 프로그래밍 툴 설치 및 설정

Python, Git, VSCode



Python
(쉬운 프로그래밍 언어)



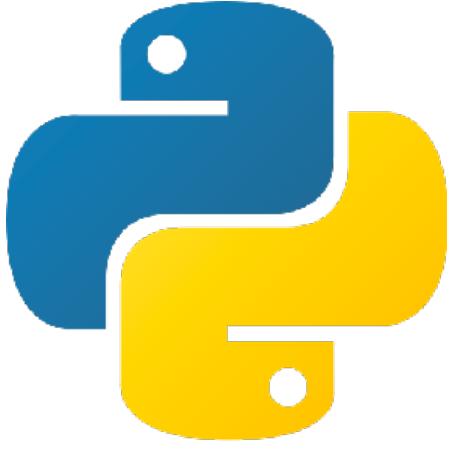
Git
(버전 관리 시스템)



Visual Studio Code
(텍스트 에디터)

0-1. Python 설치

Python 설치 유무 확인

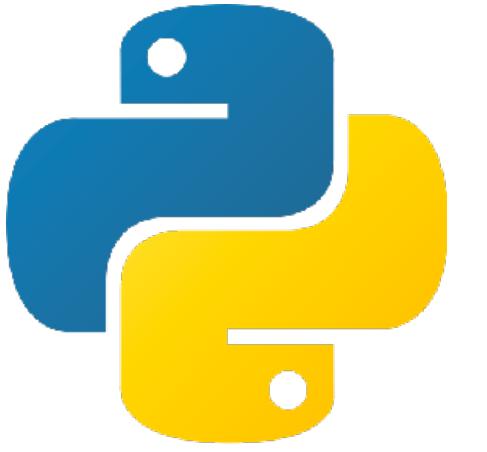


macOS 유저는 **Terminal**, Window 유저는 **PowerShell**에서
`python --version` 으로 설치 유무 검색

파이썬 3.9 이미 있으면 설치 x, 없으면 설치 그 그

0-1. Python 설치

Python 설치 주소

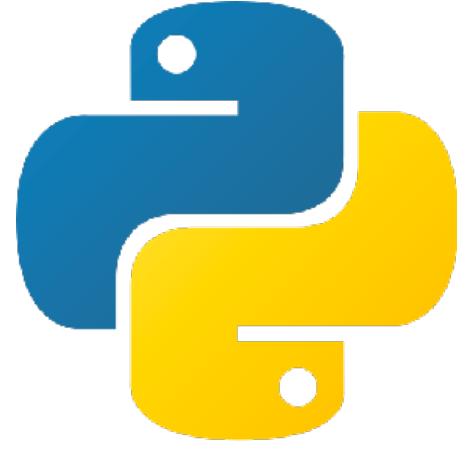


www.python.org/downloads/

v. 3.9.8

0-1. Python 설치

Python 설치 for Mac



MacOS 유저

Universal2 installer로 다운 받기

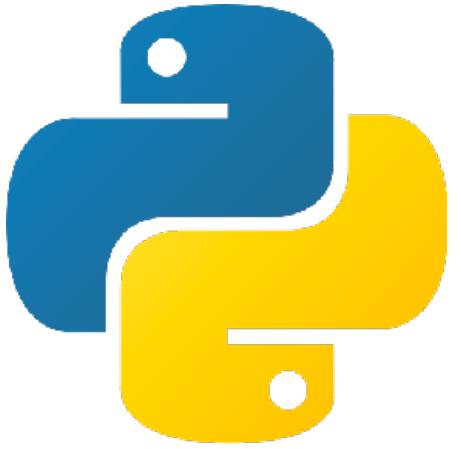
맥은 파이썬 버전 변경하기 추가 설정 필요

```
ls -l /usr/local/bin/python*
```

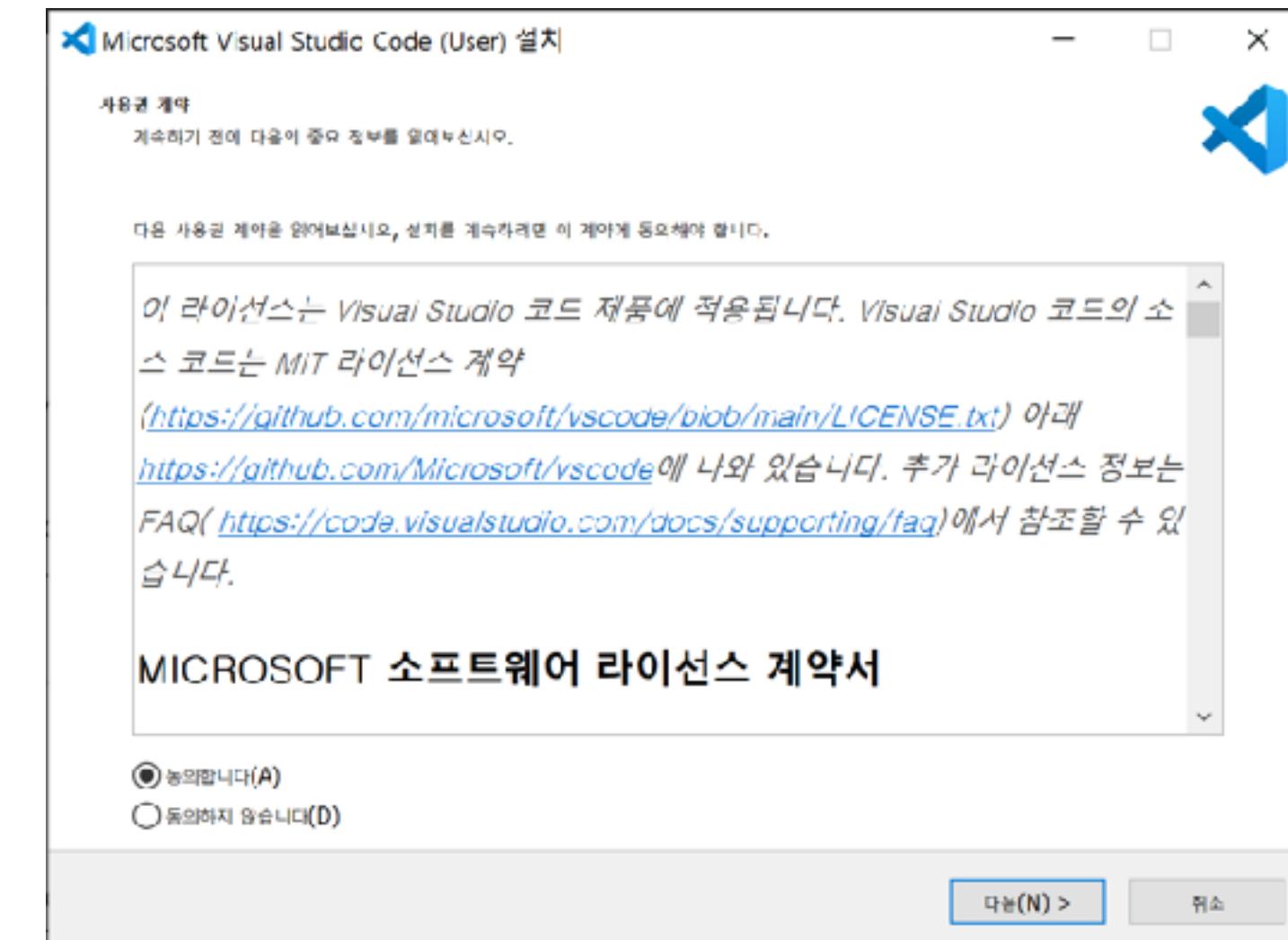
```
ln -s -f /usr/local/bin/python3.9 /usr/local/bin/python
```

0-1. Python 설치

Python 설치 for Window



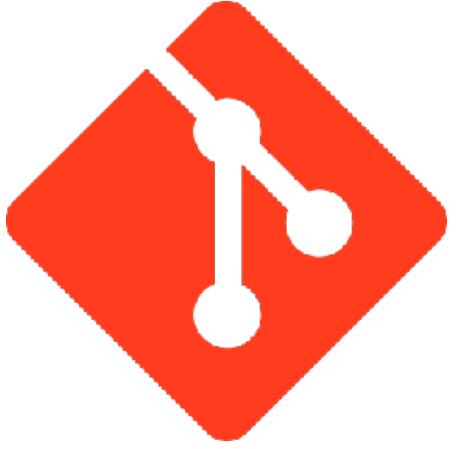
Window 유저



python 3.9 경로 추가 후 install 누르기

0-2. Git 설치

Git 설치 유무 확인



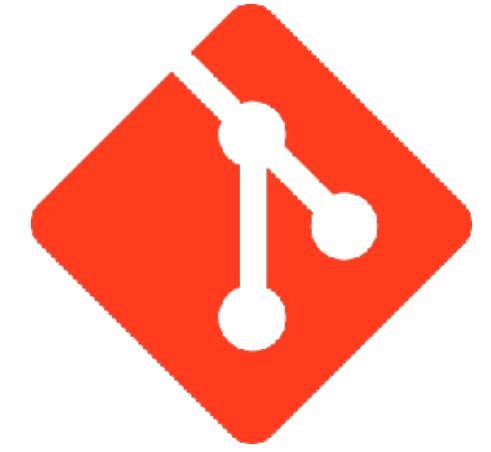
macOS 유저는 **Terminal**에서 `git --version` 으로 설치 유무 검색.
Window 유저는 왼쪽 하단 검색창에서 **Git Bash** 프로그램 유무 검색.

이미 있으면 설치 X, 없으면 설치 ↗

ONLY Window!
(Mac은 터미널에 이미 내장되어 있음)

0-2. Git 설치

Git 설치 for Window



<https://www.git-scm.com/downloads>



Window 유저

Download for Windows

[Click here to download](#) the latest (2.35.1) 32-bit version of Git for Windows. This is the most recent [maintained build](#). It was released [about 1 month ago](#), on 2022-02-01.

Other Git for Windows downloads

[Standalone Installer](#)

[32-bit Git for Windows Setup.](#)

[64-bit Git for Windows Setup.](#)

[Portable \("thumbdrive edition"\)](#)

[32-bit Git for Windows Portable.](#)

[64-bit Git for Windows Portable.](#)

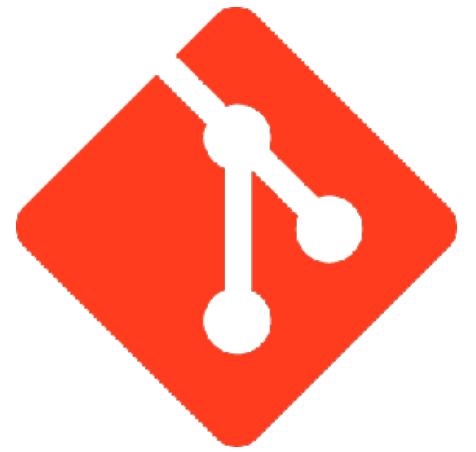
Using winget tool

Install [winget tool](#) if you don't already have it, then type this command in command prompt or Powershell.

```
winget install --id Git.Git -e --source winget
```

0-2. Git 설치

Git 설치 for Window



The image shows the Git 2.31.0 Setup process on Windows. It consists of three main windows:

- License Agreement Window:** Shows the "GNU General Public License" (Version 2, June 1991). It includes the license text, a "Preamble", and a note about free disk space required (263.5 MB). Buttons for "Next" and "Cancel" are at the bottom.
- Destination Location Window:** Allows selecting the installation path. The default is "C:\Program Files\Git". A "Browse..." button is available. A note states: "At least 263.5 MB of free disk space is required." Buttons for "Back", "Next", and "Cancel" are at the bottom.
- Terminal Emulator Configuration Window:** Asks which terminal emulator to use with Git Bash. Two options are shown:
 - Use MinTTY (the default terminal of MSYS2)**: Described as using a resizable window, non-rectangular selections, and a Unicode font. It notes that Windows console programs like interactive Python must be launched via 'wintpy'.
 - Use Windows' default console window**: Described as using the default console window of Windows ("cmd.exe"), which works well with Win32 console programs but has limited scroll-back and requires configuration for Unicode fonts.A red circle highlights the second option, and a red arrow points from the "Next" button in the previous window to this one. A red arrow also points to the "Use Windows' default console window" radio button. Korean text "클릭!" (Click!) is placed near the highlighted button.

Large arrows indicate the flow from the first window to the second, and from the second to the third. Korean text "계속 next 누르다가.." (Keep pressing Next after...) is placed between the second and third windows.

계속 next 누르고 install 하면 끝!

0-2. Git 초기 설정

Git initialization



맥은 Terminal,
윈도우는 Git Bash에서

```
-zsh

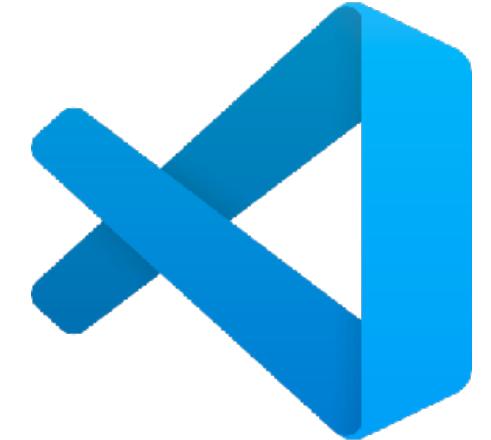
git config --global user.name ${user.name}
ex) 자신의 깃헙 아이디가 hello123이면
    git config --global user.name hello123

git config --global user.email ${user.email}
ex) 자신이 깃헙를 가입한 이메일이 hello123@korea.ac.kr이면
    git config --global user.email hello123@korea.ac.kr

git config --list
-> git username과 email 확인
```

0-3. VSCode 설치

VSCode 설치 주소



<https://code.visualstudio.com/>



Window 유저

설치 후 git bash에서
code . 쳐서 vscode 뜨면 설치 성공

* code . 의미: vscode를 현재 경로에서 열겠다.
(‘.’은 현재 경로를 의미하고, code는 (vs)code를 가리킴.)



MacOS 유저

다운로드 후 code . 따로 설정 필요!



```
>shell command  
Shell Command: Install 'code' command in PATH  
Shell Command: Uninstall 'code' command from PATH
```

Command Palette ($\uparrow \text{⌘}P$) 연 다음 shell command 검색 & 설치

0-3. VSCode 설정

VSCode extension & theme



Live Server & material icon theme 설치하기

Extension: Live Server — Session_0

Extension: Live Server

Live Server v5.7.5

Ritwick Dey | 19,465,365 | ★★★★★(364)

Launch a development local Server with live reload feature for...

Reload Required | Disable | Uninstall | ⚙️

This extension is enabled globally.

Details Feature Contributions Changelog Runtime Status

[Wanna try LIVE SERVER++ (BETA) ? It'll enable live changes without saving file. <https://github.com/ritwickdey/vscode-live-server-plus-plus>]

Categories

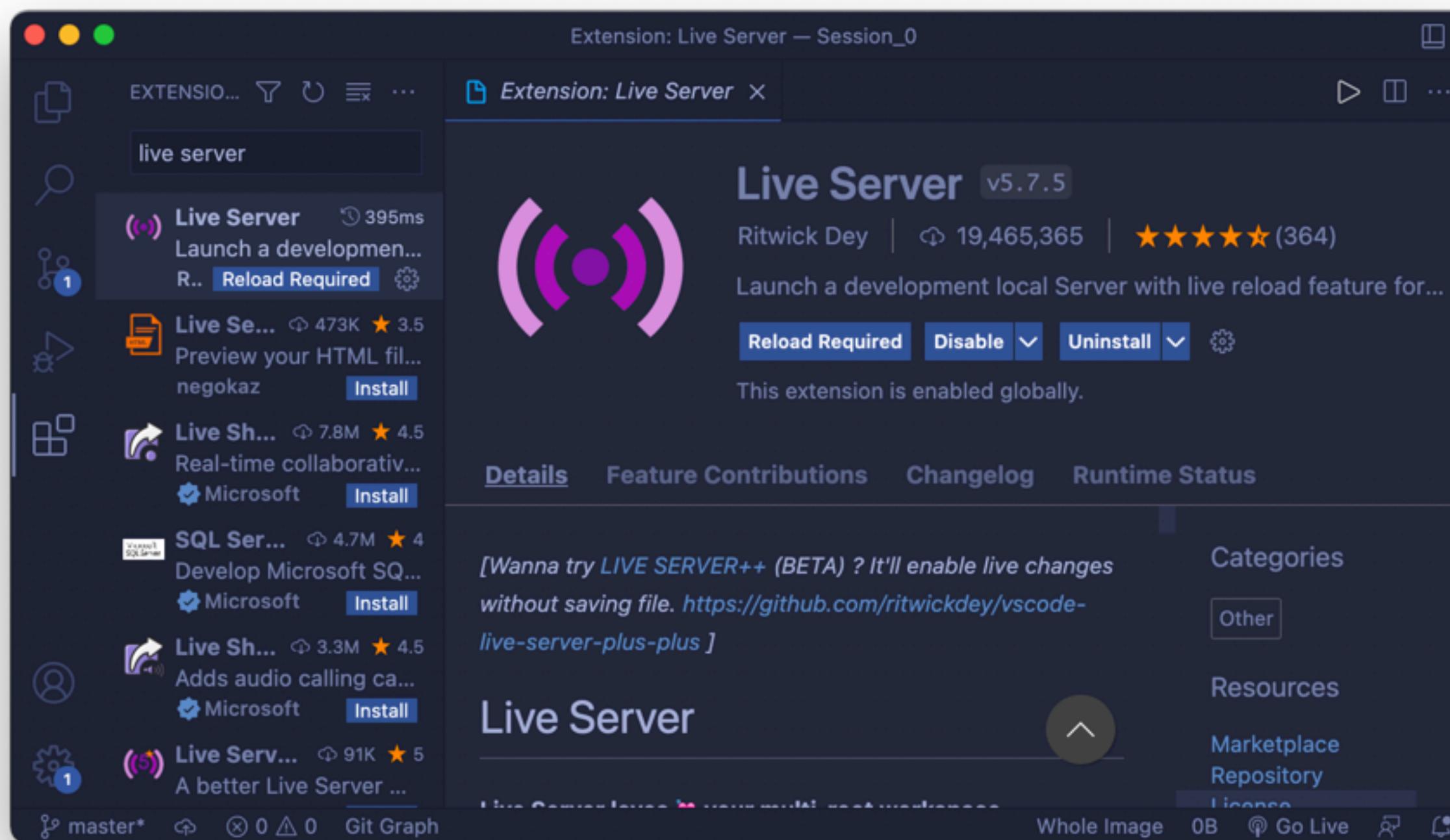
Resources

Marketplace Repository License

Live Server

Whole Image 0B Go Live ⌂ ⌂

master* ↗ 0 × 0 △ 0 Git Graph



Extension: Material Icon Theme — Session_0

Extension: Material Icon Theme

Material Icon Theme v4.14.1

Philipp Kief | 11,324,958 | ★★★★★(244)

Material Design Icons for Visual Studio Code

Reload Required | Set File Icon Theme | Disable | Uninstall | ⚙️

This extension is enabled globally.

Details Feature Contributions Changelog Runtime Status

Categories

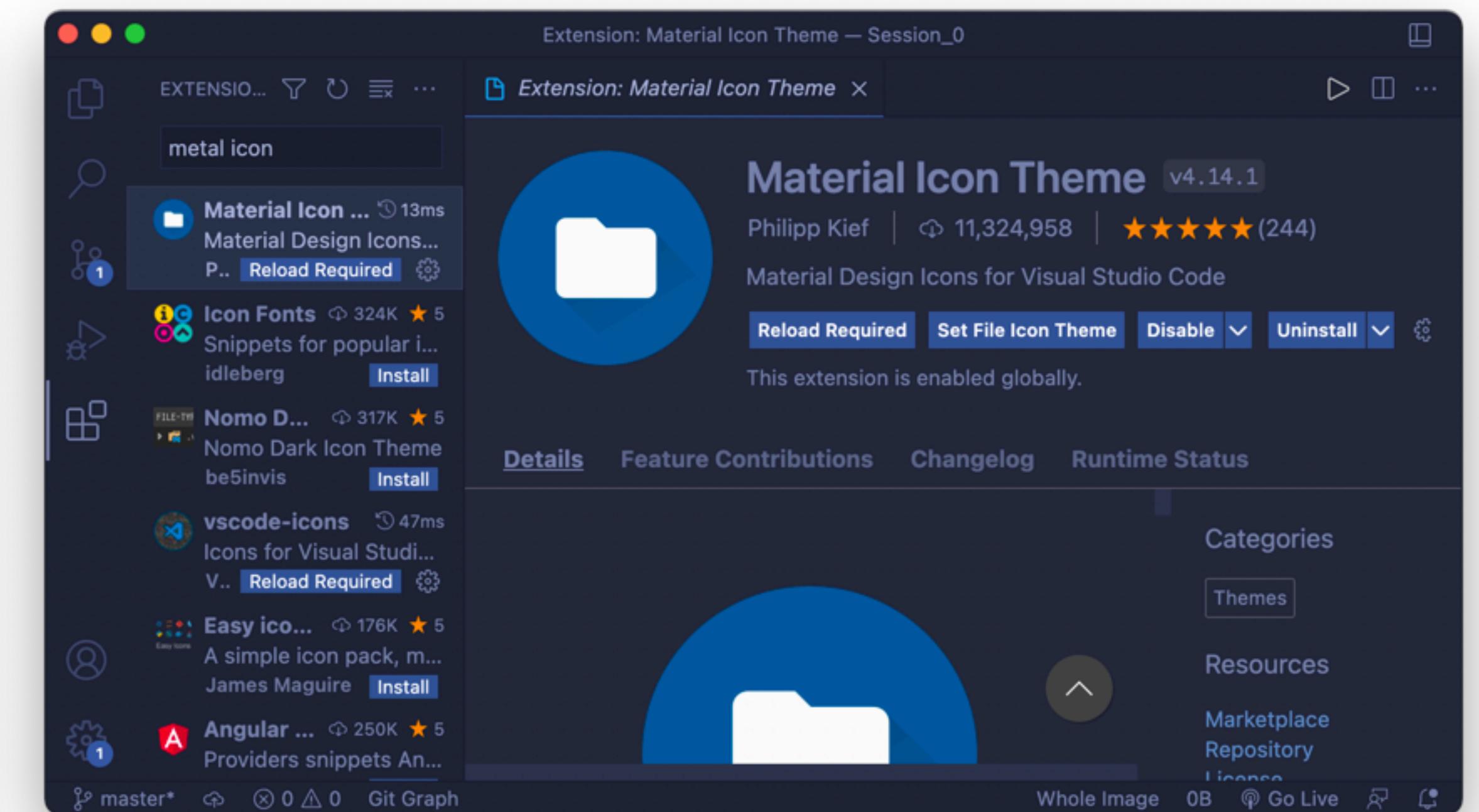
Themes

Resources

Marketplace Repository License

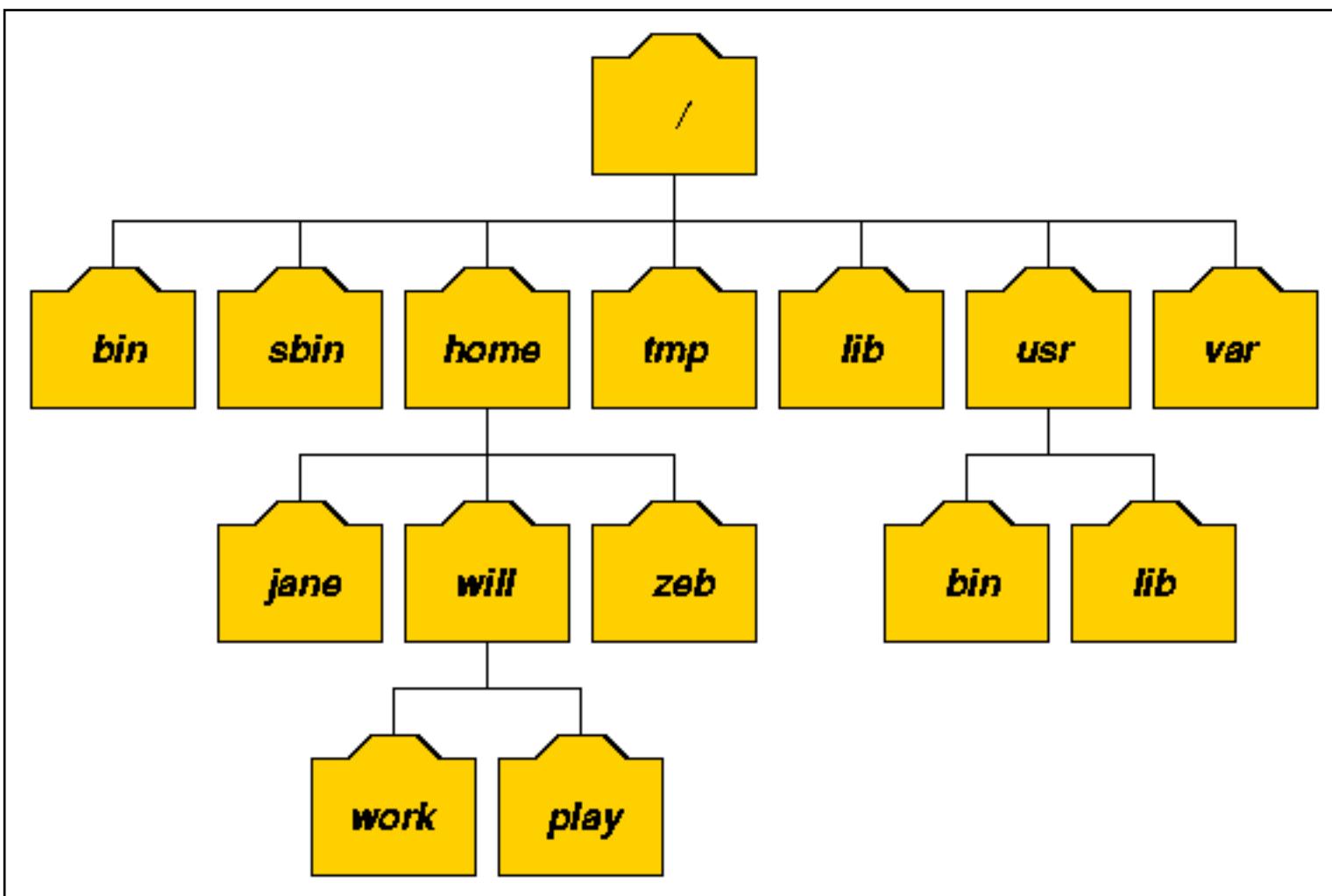
Whole Image 0B Go Live ⌂ ⌂

master* ↗ 0 × 0 △ 0 Git Graph



1. FileSystem 알아보기

File System



디렉토리 구조: 계층 트리 구조
하위 폴더에 접근하려면 현재 경로에서부터 차례대로

ex1) /home/will/play

ex2) /usr/bin

2. CLI (Command Line Interface) 배우기

CLI (Command Line Interface)



CLI (Command Line Interface)

텍스트 기반의 명령어 입력으로 시스템 조작
↔ GUI (Graphic User Interface) 버튼, 아이콘, 이미지 등과 같은 운영 체제와 상호 작용할 수 있는 비주얼을 제공

Q) 그래서 뭐가 좋은데요?

비교 근거	CLI	GUI
기본	명령 행 인터페이스는 사용자가 명령을 통해 시스템과 통신 할 수 있게합니다.	그래픽 사용자 인터페이스는 이미지, 아이콘 등이 포함 된 그래픽을 사용하여 사용자가 시스템과 상호 작용할 수 있게합니다.
사용 된 장치	건반	마우스와 키보드
수행 업무의 용이함	수술을 수행하기 어렵고 전문 지식이 필요 합니다.	작업을 수행하기 쉽고 전문 지식이 필요하지 않습니다.
정도	높은	낮은
적응성	불합리한	보다 융통성있는
메모리 사용량	낮은	높은
외관	변경할 수 없습니다.	사용자 정의 변경을 사용할 수 있음
속도	빠른	느린
통합 및 확장성	잠재적 개선의 범위	묶인

2. CLI (Command Line Interface) 배우기

Basic Command

[경로 탐색] **pwd**: print (current) working directory

[경로 이동] **cd**: change directory

[파일 확인] **ls**: list segments

ls -a: list segments of all

ls -l: list directory with long listing

ls -al: list directory of all with long listing

[파일 생성] **touch**: create a file

mkdir: make a directory

[파일 제거] **rm**: remove a file

rm -rf: remove forcibly including directory and the files in it

2. CLI (Command Line Interface) 배우기

pwd - print (current) working directory

pwd

현재 절대 경로를 확인합니다.

```
chunhyerin@iMac Session_0 % pwd  
/Users/chunhyerin/Desktop/Session_0
```

2. CLI (Command Line Interface) 배우기

cd - change directory

cd 원하는_경로

원하는 절대 경로, 상대 경로 주소값으로
이동합니다.

```
chunhyerin@iMac Session_0 % cd ..  
chunhyerin@iMac Desktop % cd Session_0  
chunhyerin@iMac Session_0 %
```

상위 폴더인 Desktop 으로 이동
Session_0 으로 이동

절대경로 - root directory에서 해당 directory까지의 경로
(Pwd 를 사용했을 때 나오는 경로)

상대경로 - 현재 directory에서 해당 directory까지의 경로

“.” - 현재 경로를 의미
“..” - 위의 경로를 의미

2. CLI (Command Line Interface) 배우기

ls - 자세하게 알아보기

ls 현재 디렉토리 위치에 내용 나열

ls -a 숨김 파일 (.., .git, .gitignore 등)
포함해서 모두 표시

ls -l 좀 더 자세한 정보 (수정 날짜 등)
포함해서 출력

ls -al ls -a, ls -l 둘 다 같이 섞어서
출력

```
chunhyerin@iMac Session_0 % ls
README.md      Session0.png  Session0.txt
chunhyerin@iMac Session_0 % ls -a
.
..
.git           README.md    Session0.png  Session0.txt
chunhyerin@iMac Session_0 % ls -l
total 8
-rw-r--r--  1 chunhyerin  staff   103  3 10 10:20 README.md
-rw-r--r--  1 chunhyerin  staff     0  3 10 09:04 Session0.png
-rw-r--r--  1 chunhyerin  staff     0  3 10 09:04 Session0.txt
chunhyerin@iMac Session_0 % ls -al
total 8
drwxr-xr-x  6 chunhyerin  staff  192  3 10 09:32 .
drwx-----+ 7 chunhyerin  staff  224  3 10 09:03 ..
drwxr-xr-x 12 chunhyerin  staff  384  3 10 09:35 .git
-rw-r--r--  1 chunhyerin  staff   103  3 10 10:20 README.md
-rw-r--r--  1 chunhyerin  staff     0  3 10 09:04 Session0.png
-rw-r--r--  1 chunhyerin  staff     0  3 10 09:04 Session0.txt
```

2. CLI (Command Line Interface) 배우기

touch - create a file

touch 파일_이름

현재 디렉터리에서 “파일_이름”을
가진 파일을 생성합니다.

```
chunhyerin@iMac Session_0 % touch hello.txt
chunhyerin@iMac Session_0 % ls
README.md           Session0.png      Session0.txt      hello.txt
```

2. CLI (Command Line Interface) 배우기

mkdir - make a directory

mkdir 폴더_이름

현재 디렉터리에서 “폴더_이름”을
가진 폴더를 생성합니다.

```
chunhyerin@iMac Session_0 % mkdir newFolder
chunhyerin@iMac Session_0 % ls
README.md          Session0.png      Session0.txt      newFolder
```

2. CLI (Command Line Interface) 배우기

rm - remove

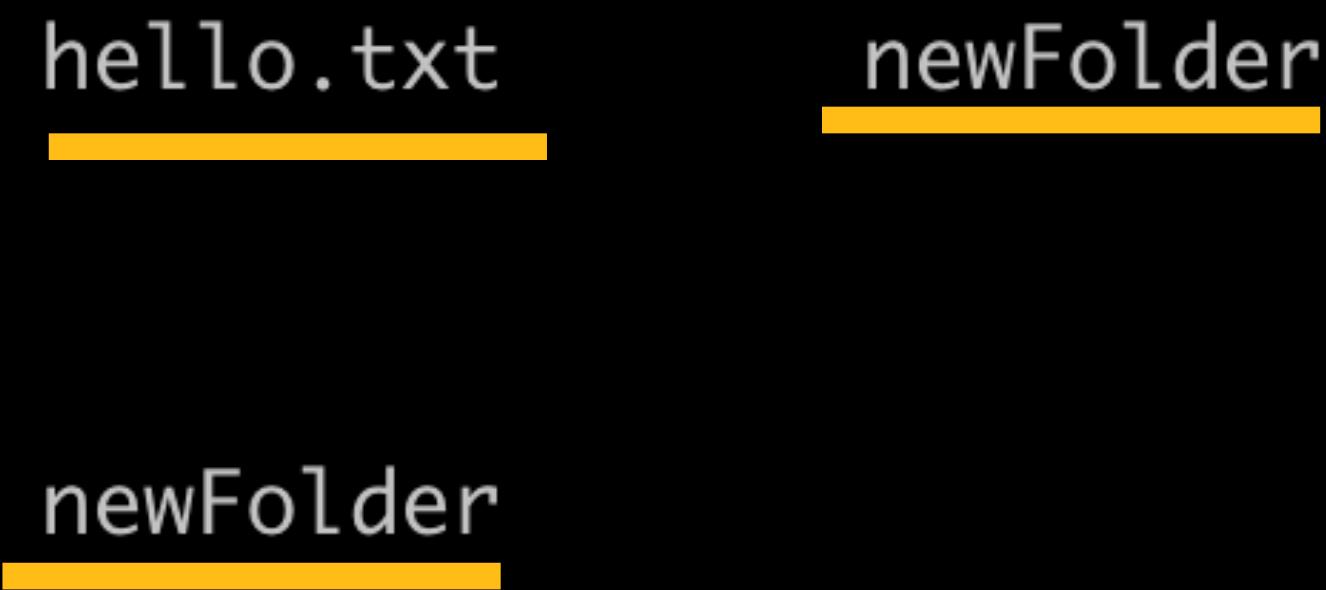
rm 파일_이름

현재 디렉터리에서 “파일_이름”을 가진 파일을 삭제합니다.

rm - rf 폴더_이름

현재 디렉터리에서 “폴더_이름”을 가진 파일을 삭제합니다.

```
chunhyerin@iMac Session_0 % ls  
README.md      Session0.png      Session0.txt  
  
chunhyerin@iMac Session_0 % rm hello.txt  
  
chunhyerin@iMac Session_0 % ls  
README.md      Session0.png      Session0.txt  
  
chunhyerin@iMac Session_0 % rm -rf newFolder  
  
chunhyerin@iMac Session_0 % ls  
README.md      Session0.png      Session0.txt
```



3. 로컬에서 Git 기초 버전관리

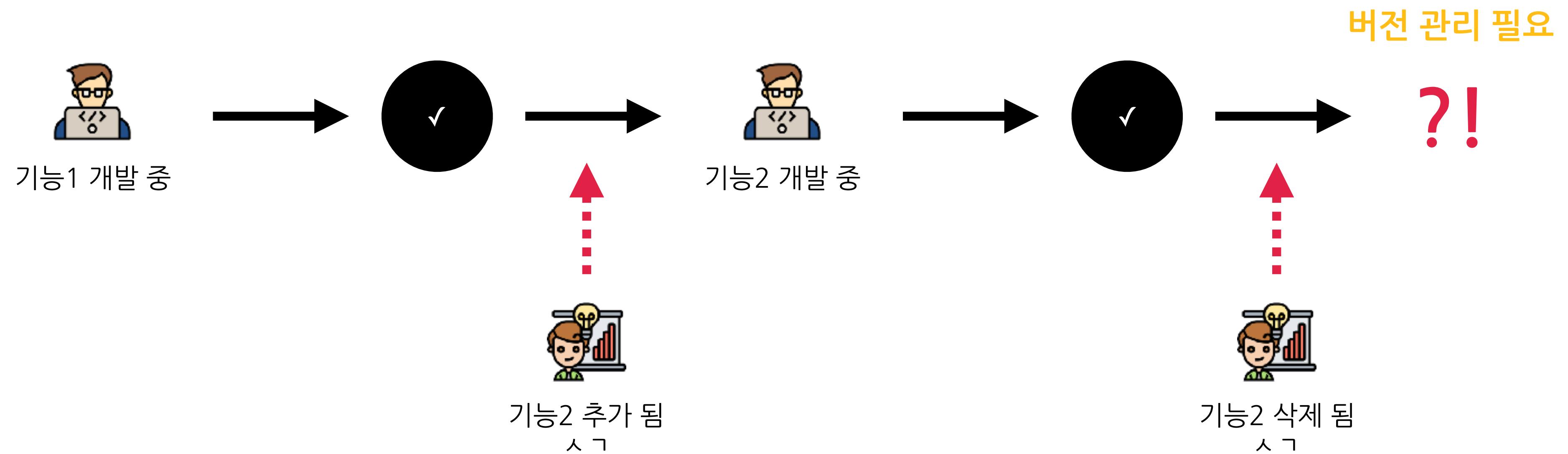
What is Git?

커밋이요? 머지요? 뭐요?



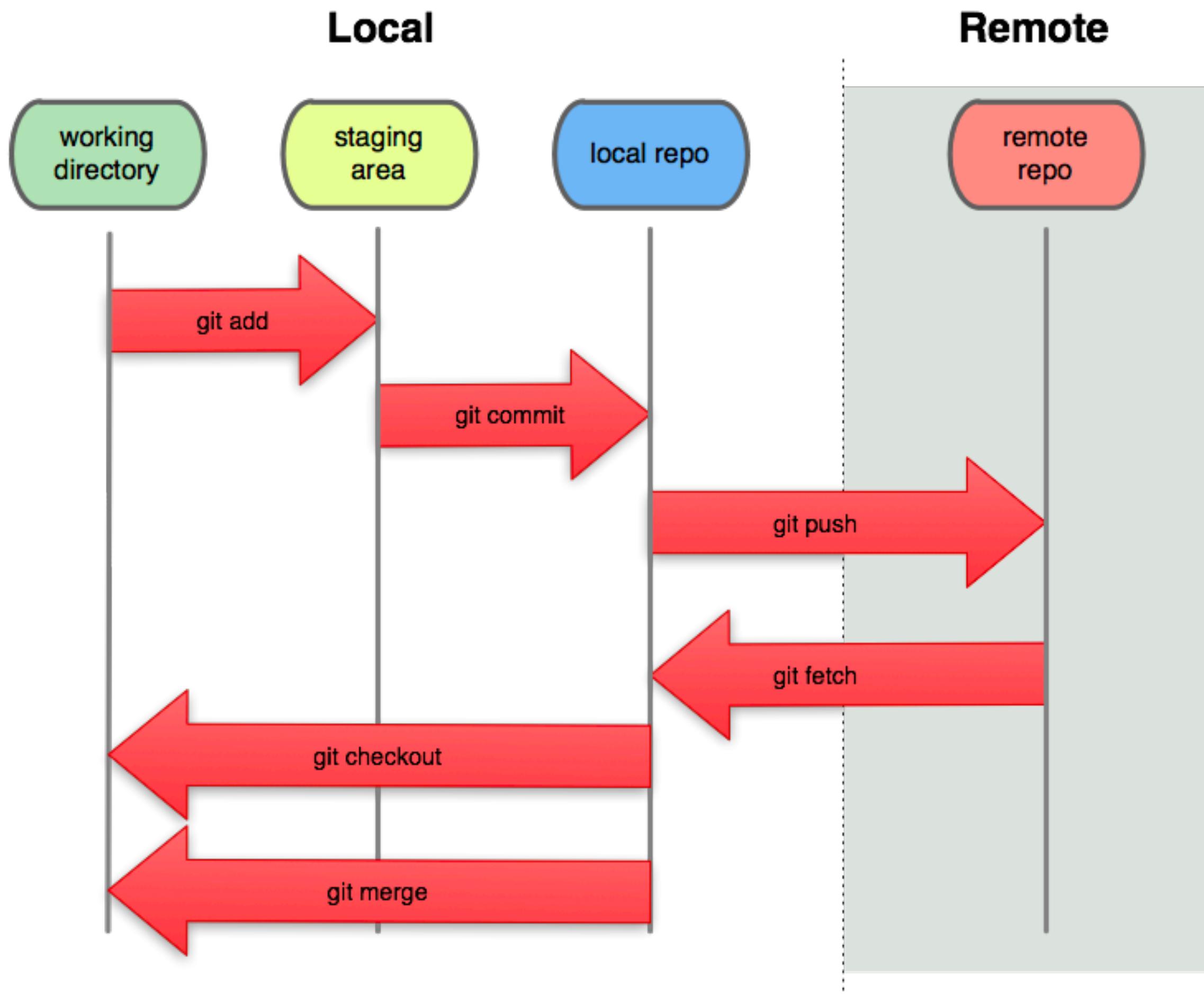
3. 로컬에서 Git 기초 버전관리

Why Git?



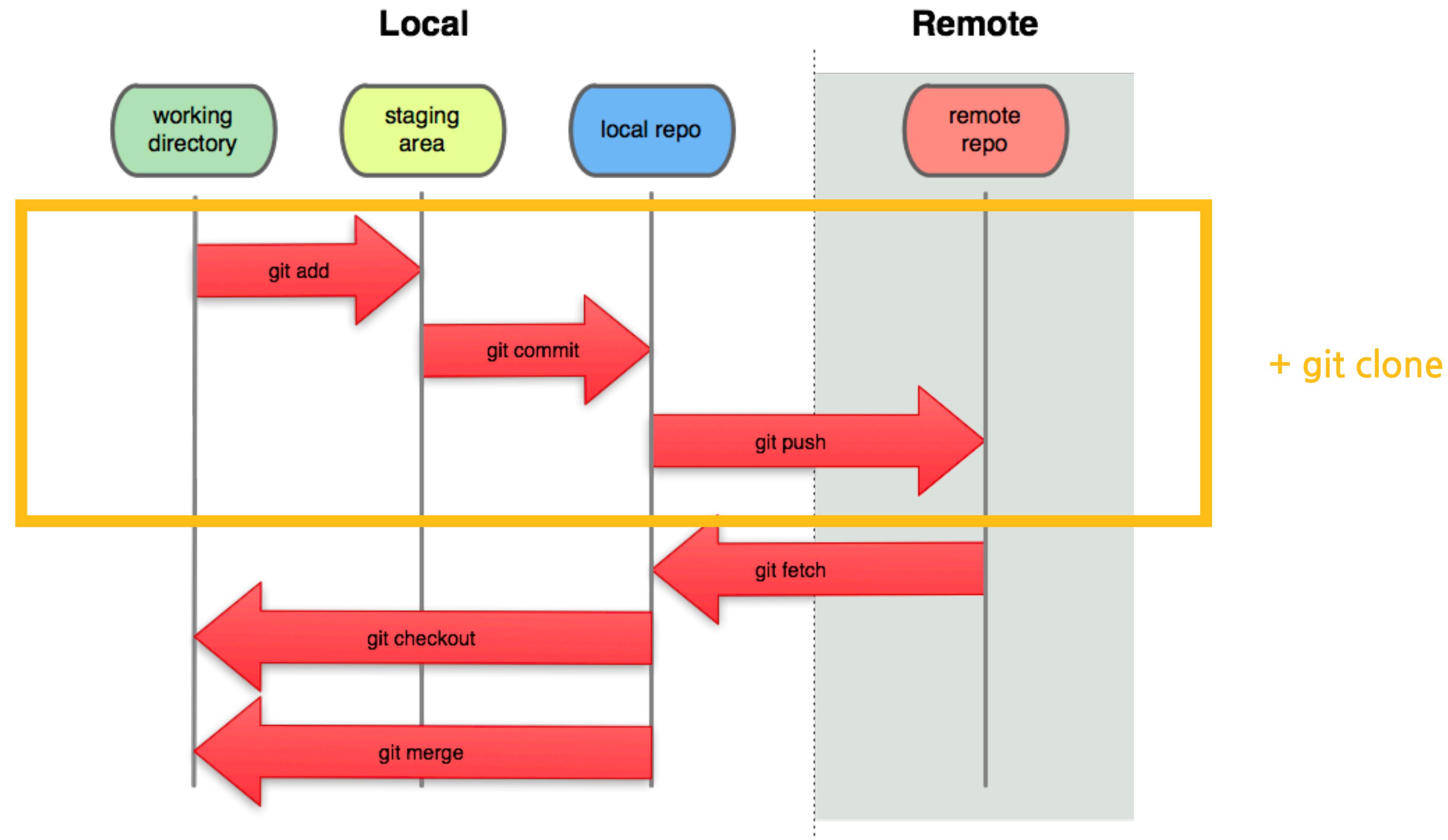
3. 로컬에서 Git 기초 버전관리

Git 기능 정리



3. 로컬에서 Git 기초 버전관리

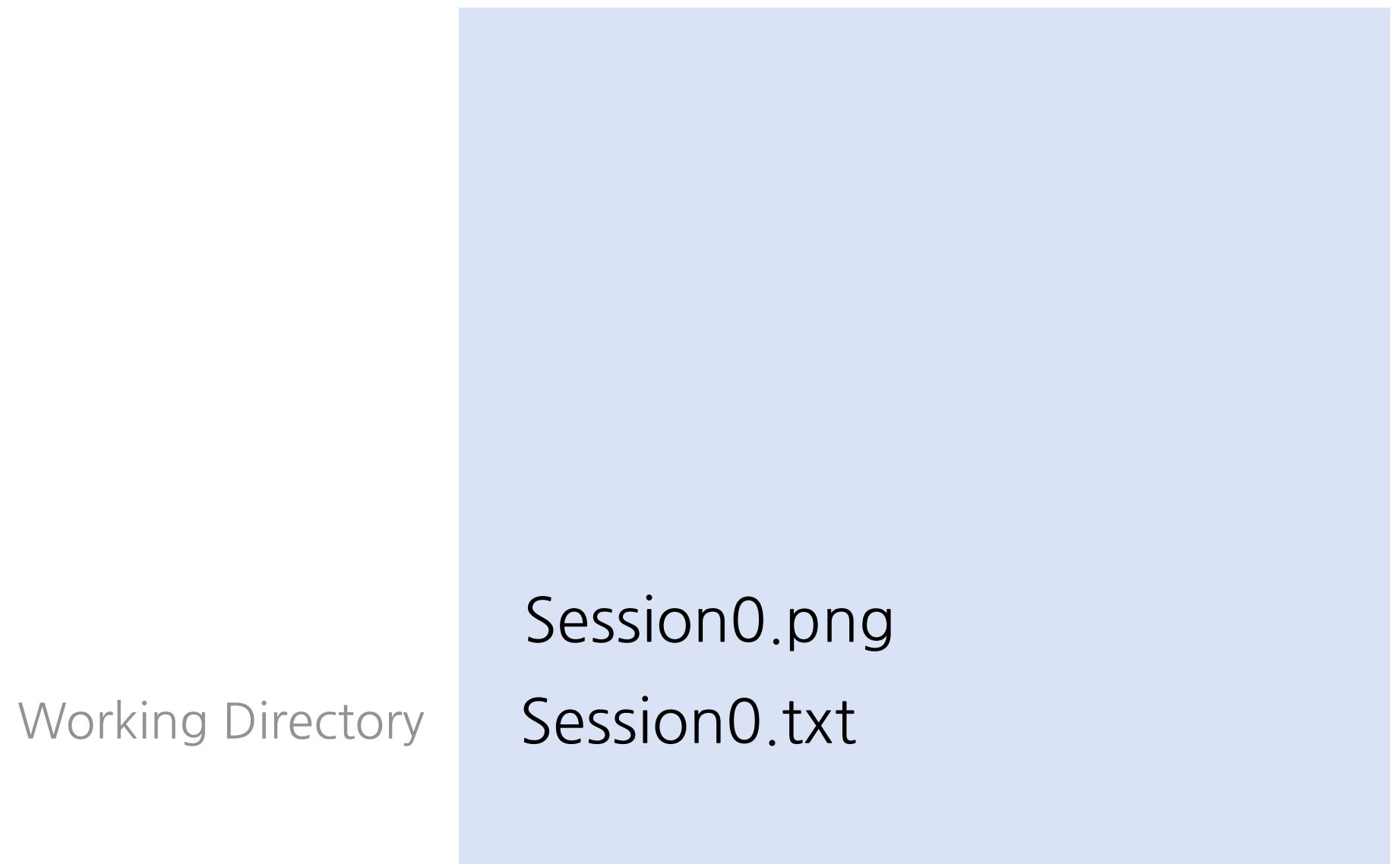
Git 기능 정리



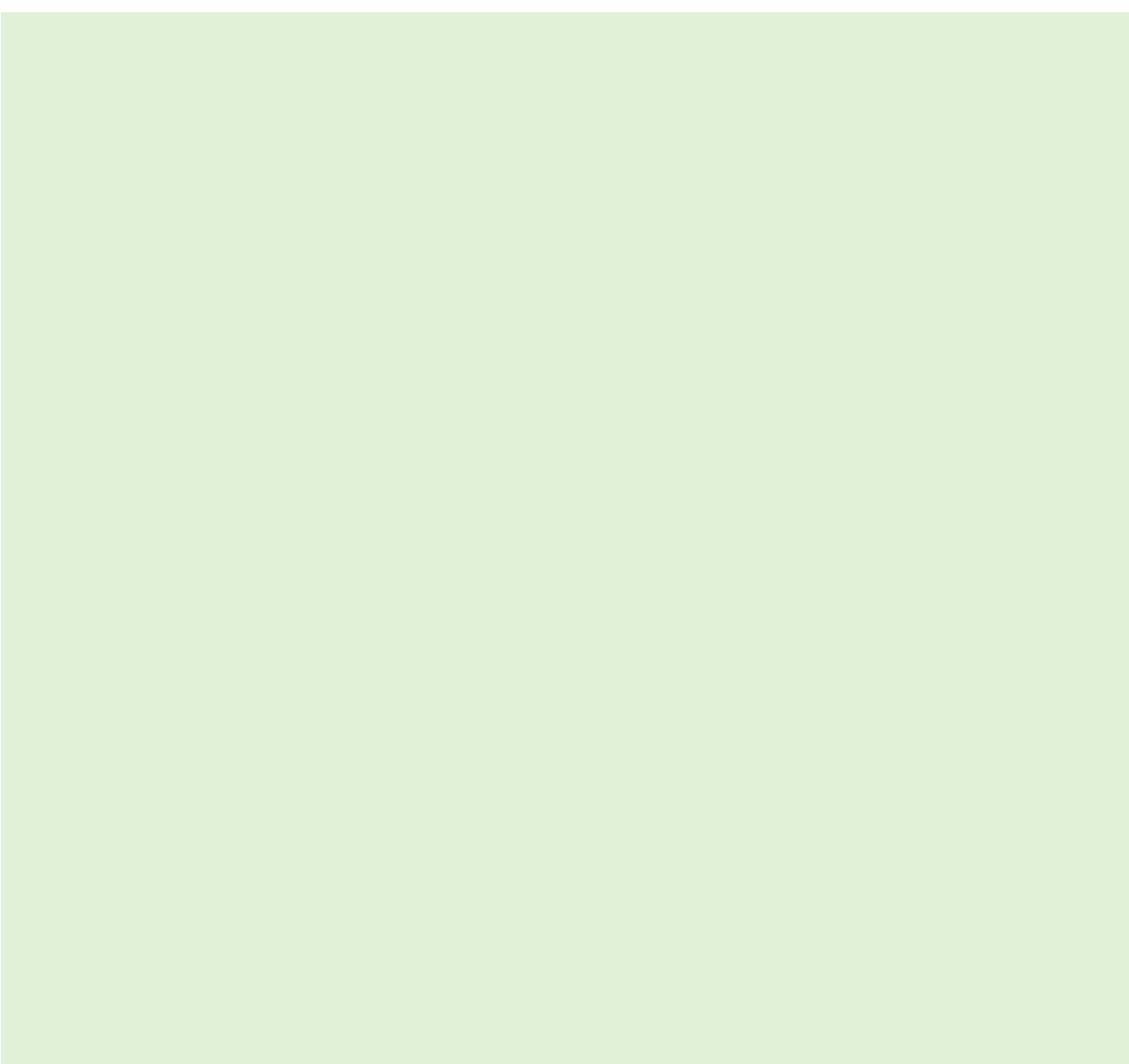
3. 로컬에서 Git 기초 버전관리

아무것도 안 했을 때 초기 상태

Local

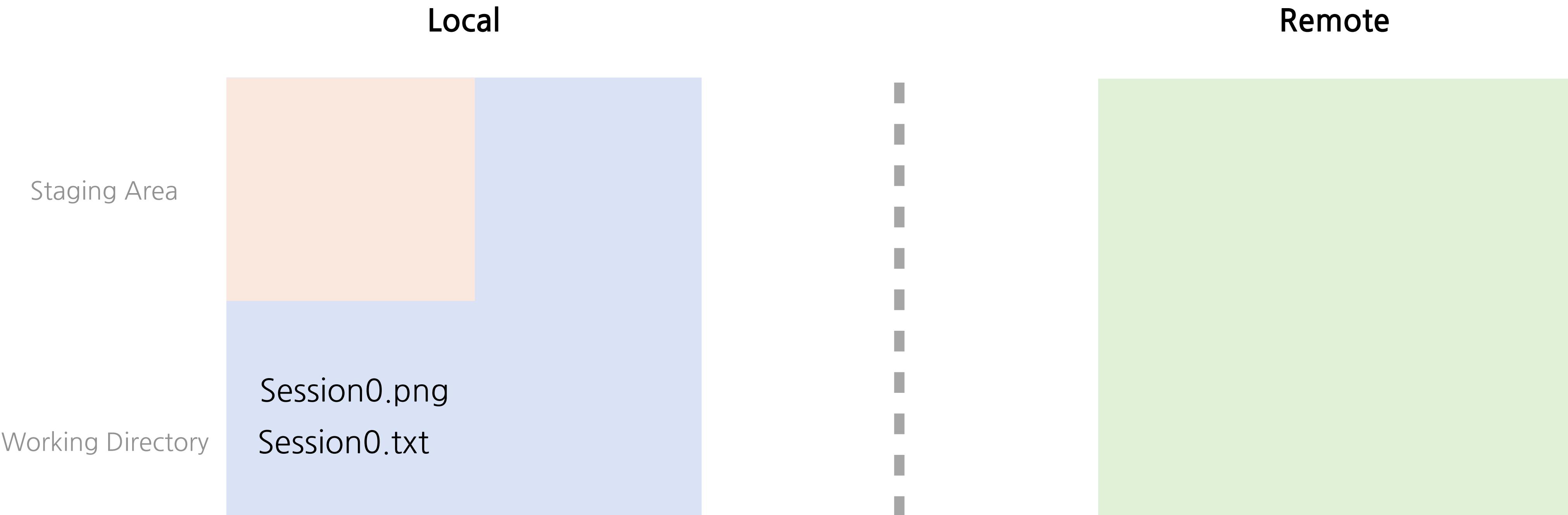


Remote



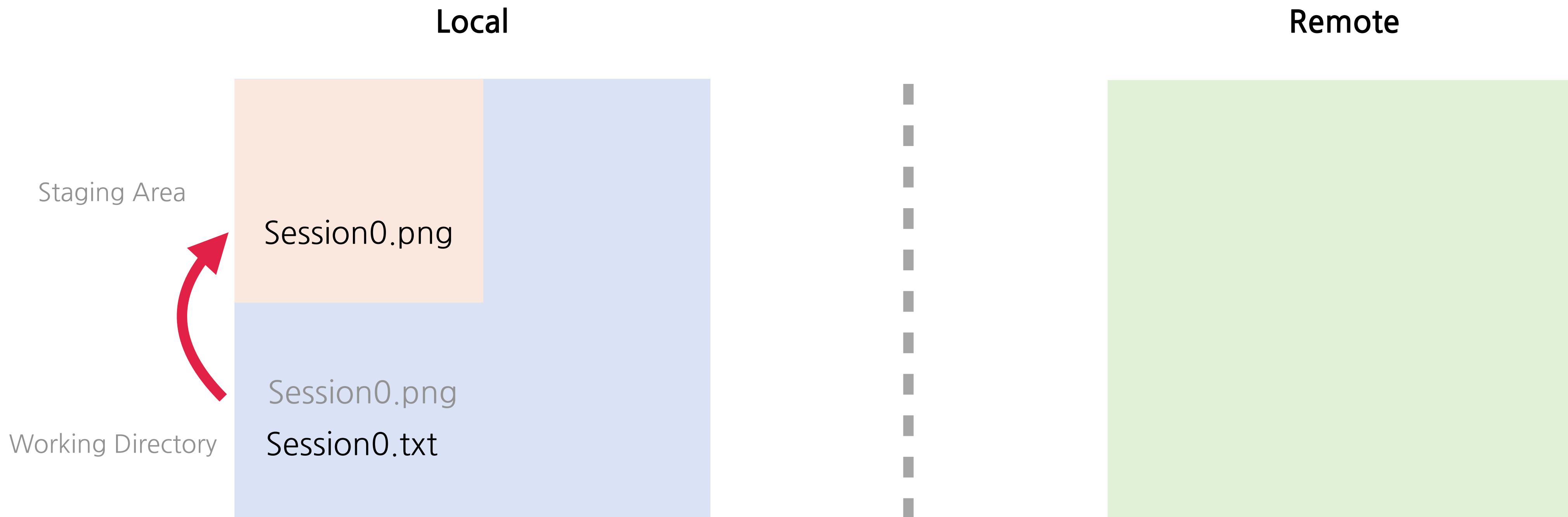
3. 로컬에서 Git 기초 버전관리

스테이징 영역 활성화 및 추적 시작



3. 로컬에서 Git 기초 버전관리

스테이징 영역에 파일 추가



git add \${file_name}

`git add Session0.png`

3. 로컬에서 Git 기초 버전관리

Local Repository에 확정



git commit -m “\${커밋메세지}”
git commit -m “first commit”

3. 로컬에서 Git 기초 버전관리

Git 기록 확인

```
chunhyerin@iMac Session_0 % git log  
commit c3f15a64ed342864b782469bfb2ebae2525a3eba (HEAD -> master)  
Author: deborah91298 <deborah91298@korea.ac.kr>  
Date:   Thu Mar 10 09:34:09 2022 +0900
```

첫 세션 푸시

git log

3. 로컬에서 Git 기초 버전관리

Git 기록 파일 전부 추적



git add .
git commit -m “second commit”

“.” - 현재 경로를 의미.
즉, 현재 경로에 있는 모든
파일 및 폴더를 포함함

4. Github에 푸시하기

Github 란?



Github

협업 지원, 오픈 소스의 성지, 코드 관리 페이지

4. Github에 푸시하기

README.md 란?

README.md

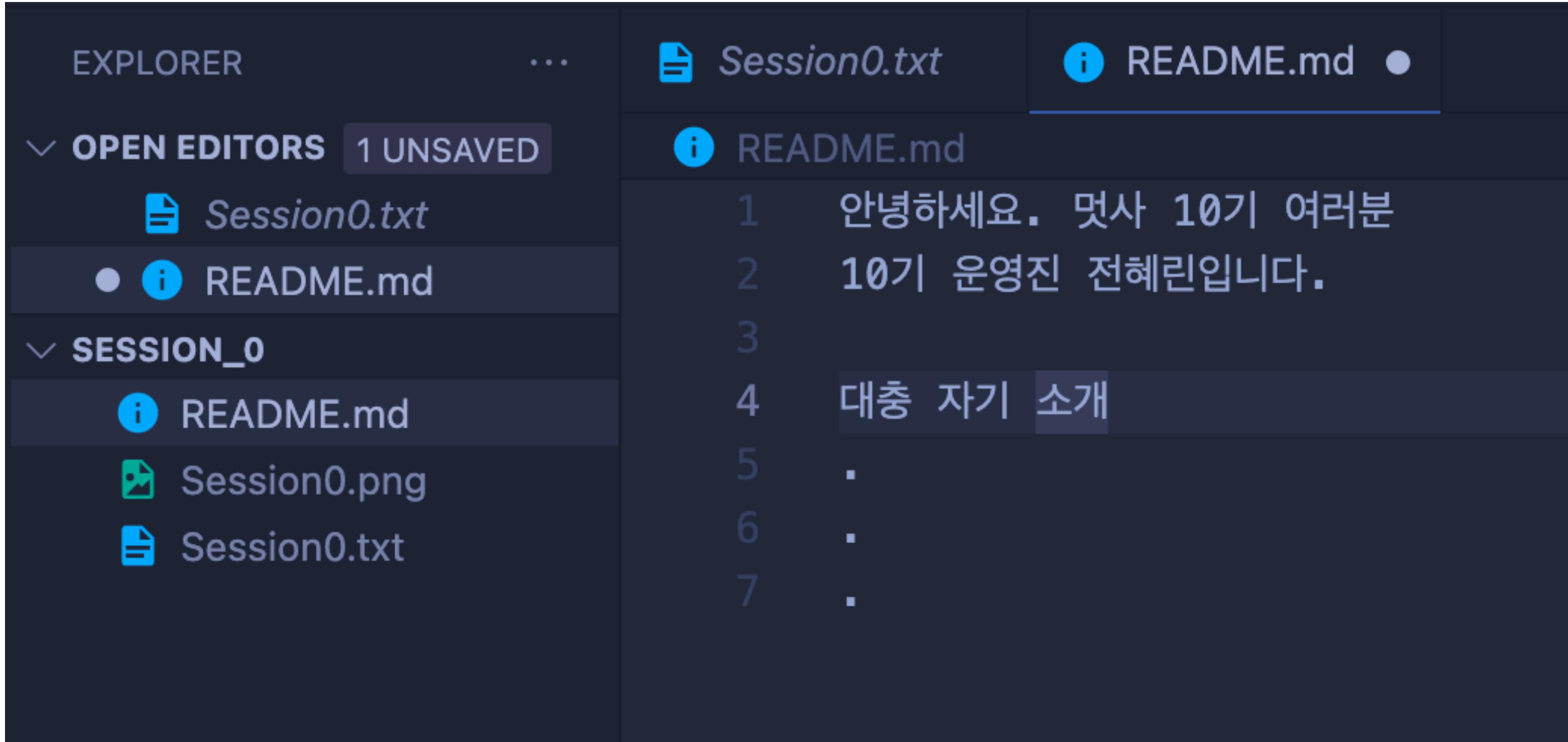
Github에 프로젝트를 올릴 때, 프로젝트에 대한 설명뿐 아니라 사용 방법, LICENSE 등의 내용을 기술하는 파일이다.



4. Github에 푸시하기

README.md 작성

code . 해당 경로에서, (vs)code를 열어라! (앞에서 배웠죠?)



4. Github에 푸시하기

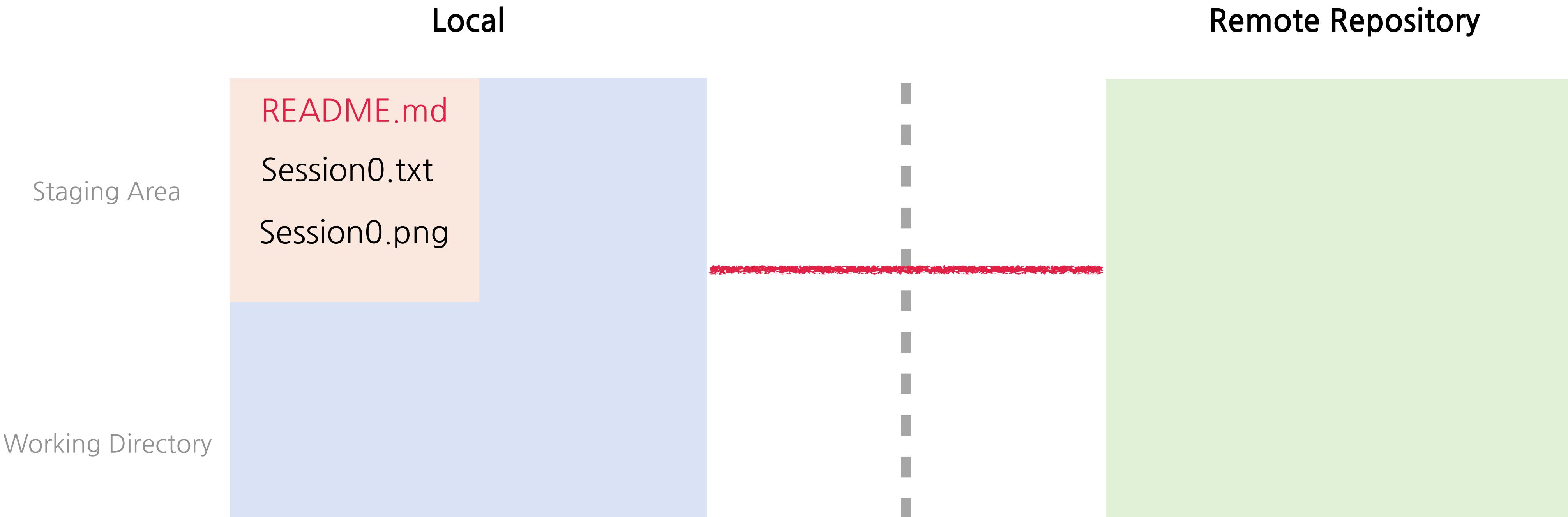
Remote Repository 현 상황

만들어져 있는데 연결이 안 됐네?



4. Github에 푸시하기

Local Repository와 Remote Repository 연결



git remote add origin “본인 Repository 주소”

ex) git remote add origin https://github.com/deborah91298/Next_Likelion.git

4. Github에 푸시하기

Local Repository와 Github Repository 연결 확인

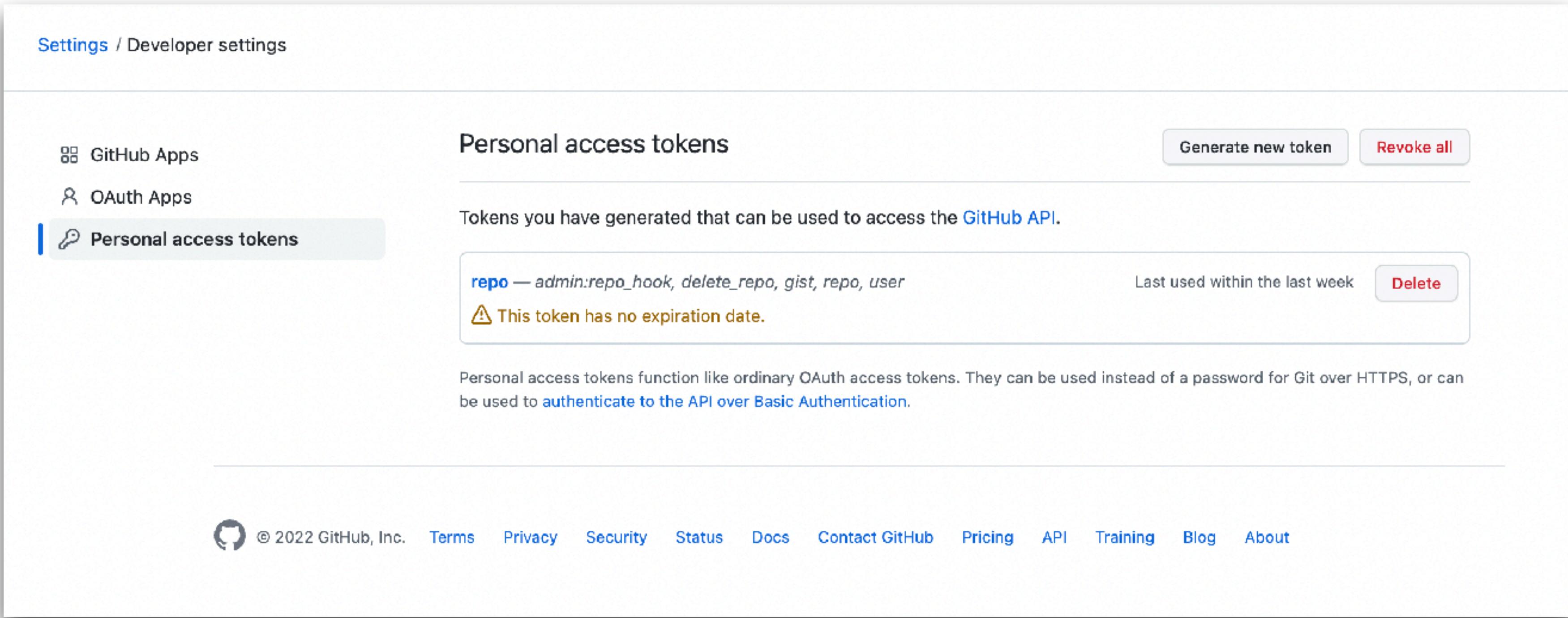
```
chunhyerin@iMac Session_0 % git remote -v  
origin  https://github.com/deborah91298/Next_Likelion.git (fetch)  
origin  https://github.com/deborah91298/Next_Likelion.git (push)
```

git remote -v

4. Github에 푸시하기

Github Repository에 접근 가능한 Personal Access Token 생성

My profile > Settings > Developer Settings > Personal access tokens > Generate new token



4. Github에 푸시하기

Github Repository에 접근 가능한 Personal Access Token 생성

The screenshot shows two overlapping GitHub modal windows. The top window is titled 'Note' and contains fields for 'repo' (selected), 'Expiration' (no expiration date), and a 'Select scopes' section. The 'repo' scope is checked, along with several others like 'repo:status', 'repo_deployment', 'public_repo', 'repo:invite', 'security_events', and 'admin:repo_hook'. The bottom window is titled 'Delete personal access token' and lists scopes: 'user', 'delete_repo', and various enterprise and GPG key scopes. It includes 'Update token' and 'Cancel' buttons. A note at the bottom states: 'Any applications or scripts using this token will no longer be able to access the GitHub API. You cannot undo this action.' with a 'Delete this token' button.

Note

repo

What's this token for?

Expiration

This token has no expiration date. To set a new expiration date, you must regenerate the token.

Select scopes

Scopes define the access for personal tokens. [Read more about OAuth scopes.](#)

<input checked="" type="checkbox"/> repo	Full control of private repositories
<input checked="" type="checkbox"/> repo:status	Access commit status
<input checked="" type="checkbox"/> repo_deployment	Access deployment status
<input checked="" type="checkbox"/> public_repo	Access public repositories
<input checked="" type="checkbox"/> repo:invite	Access repository invitations
<input checked="" type="checkbox"/> security_events	Read and write security events
<input type="checkbox"/> workflow	Update GitHub Action workflows
<input type="checkbox"/> write:packages	Upload packages to GitHub Package Registry
<input type="checkbox"/> read:packages	Download packages from GitHub Package Registry
<input type="checkbox"/> delete:packages	Delete packages from GitHub Package Registry
<input type="checkbox"/> admin:org	Full control of orgs and teams, read and write org projects
<input type="checkbox"/> write:org	Read and write org and team membership, read and write org projects
<input type="checkbox"/> read:org	Read org and team membership, read org projects
<input type="checkbox"/> admin:public_key	Full control of user public keys
<input type="checkbox"/> write:public_key	Write user public keys
<input type="checkbox"/> read:public_key	Read user public keys
<input checked="" type="checkbox"/> admin:repo_hook	Full control of repository hooks
<input checked="" type="checkbox"/> write:repo_hook	Write repository hooks
<input checked="" type="checkbox"/> read:repo_hook	Read repository hooks
<input type="checkbox"/> admin:org_hook	Full control of organization hooks
<input checked="" type="checkbox"/> gist	Create gists
<input type="checkbox"/> notifications	Access notifications

user

read:user

user:email

user:follow

delete_repo

write:discussion

read:discussion

admin:enterprise

manage_runners:enterprise

manage_billing:enterprise

read:enterprise

admin:gpg_key

write:gpg_key

read:gpg_key

Update token

Cancel

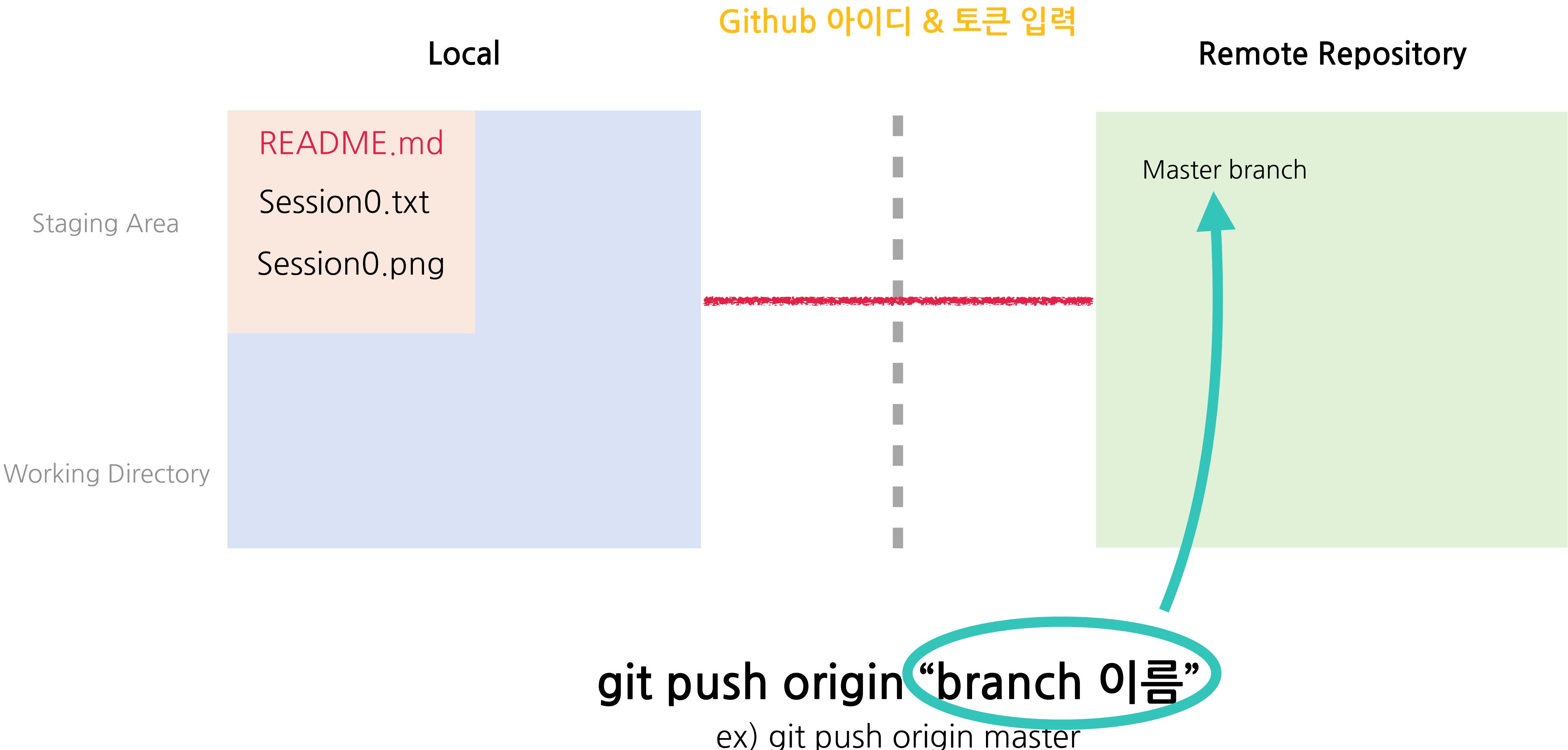
Delete personal access token

Any applications or scripts using this token will no longer be able to access the GitHub API. You cannot undo this action.

Delete this token

4. Github에 푸시하기

Github Repository에 코드 올리기



4. Github에 푸시하기

Github Repository에 올린 코드 확인



master ▾ 1 branch 0 tags

Go to file Add file ▾ Code ▾

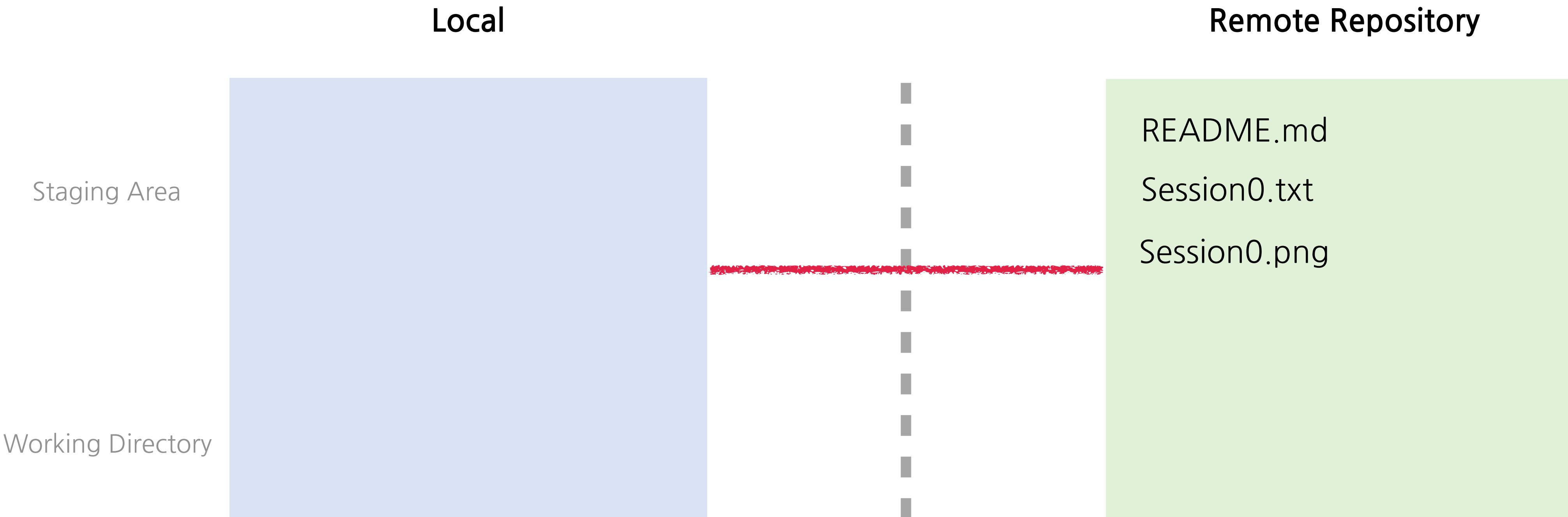
 deborah91298 첫 세션 푸시 c3f15a6 11 minutes ago 1 commit

 README.md	첫 세션 푸시	11 minutes ago
 Session0.png	첫 세션 푸시	11 minutes ago
 Session0.txt	첫 세션 푸시	11 minutes ago

Add a README with an overview of your project. [Add a README](#)

4. Github에 푸시

코드를 올릴 수 있으면 받아올 수도 있어야겠죠?



git clone “깃헙 주소”

ex) git clone https://github.com/deborah91298/Next_Likelion.git

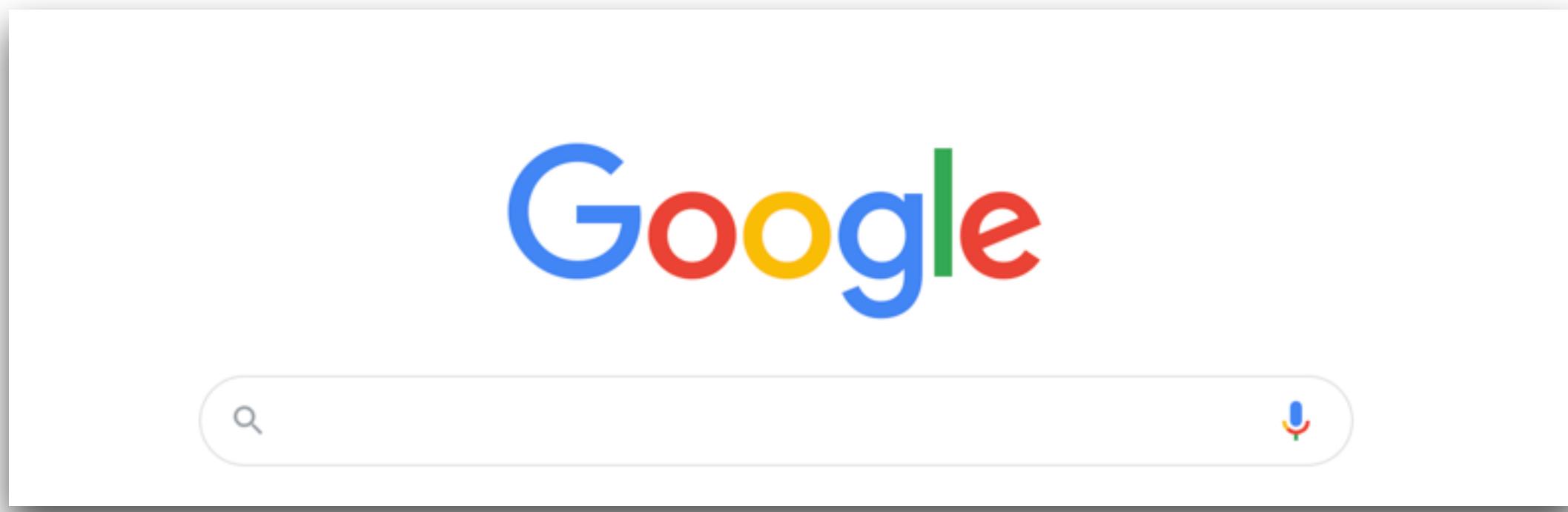
*기존 폴더와 다른 폴더에서 git clone(복사) 해주세요.

| 도전!) README.md 꾸며보기

README.md 추가 기능

1. 그림 넣기
2. 링크 추가
3. 제목별로 크기 다르게 하기

스스로 시도해봅시당-!



과제

Session0_HW

이번 세션 과제

1. 본인 깃헙에서 repository 판 다음에 Session0_HW 디렉토리 밑에 본인 만의 README 작성해보기 (~3.14.월 오후 6시)

전체 과제

1. Codecademy에서 Learn HTML / Learn CSS 수강 (~3.17.목 오후 6시)
2. Codecademy에서 python 수강 (차후 다시 공지 예정)

과제

Session0_HW

The screenshot shows a GitHub repository page for 'deborah91298 / NEXTLikeLion'. The 'Code' tab is selected. A dropdown menu shows 'master'. Below it is a commit history table.

Author	Commit Message	Date
deborah91298	session17_hw	on 16 Jul 2021
Session0_HW	folder name changed	12 months ago
Session11_HW	session11_hw	10 months ago
Session12_HW	session12_hw	10 months ago
Session15_HW	session15_hw	8 months ago
Session17_HW	session17_hw	8 months ago
Session1_HW	image's name changed	12 months ago
Session2_HW	smth changed?	12 months ago
Session3_HW	multiplication code changed	12 months ago
Session4_HW	facebook.html changed	11 months ago
Session5_HW	show_section modified	11 months ago
Session6_HW	session6 HW	11 months ago
Session7_HW	session7_hw	10 months ago
Session8_HW	comments added	10 months ago
Session9_HW	session9_hw	10 months ago