

Github으로 프로젝트 관리하기

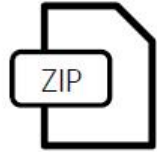
장준환

@github.com/junhwanjang

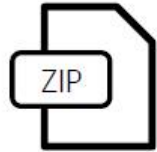


Git 과 Github

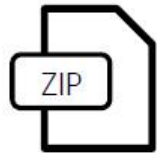
01 왜 쓰는가?



프로젝트_최종.zip



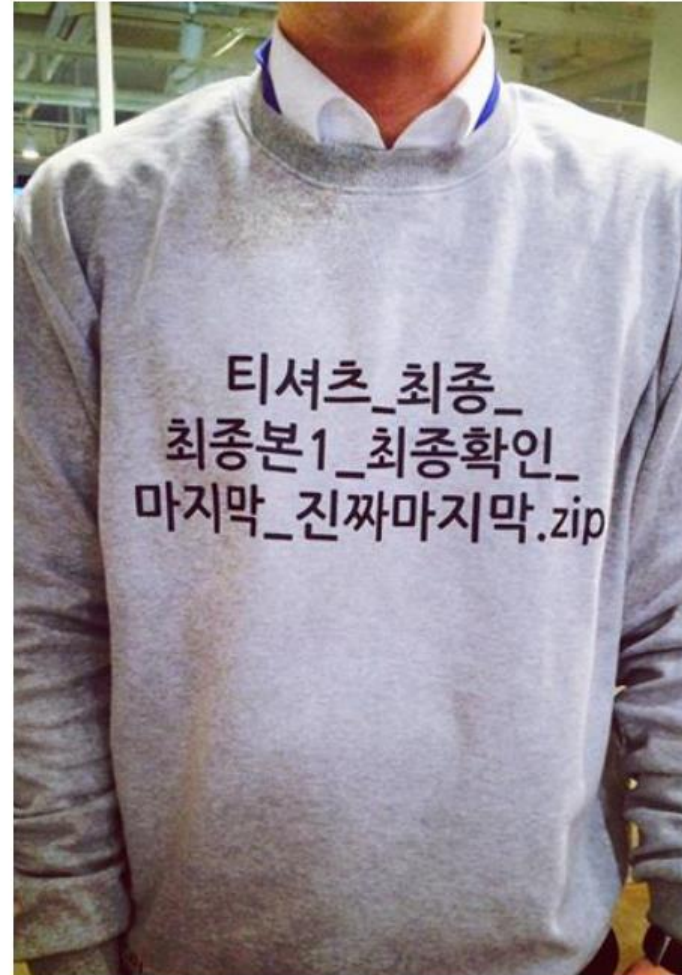
프로젝트_최종_최종2.zip



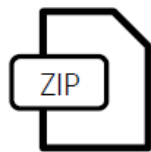
프로젝트_최종_최종2_마지막.zip



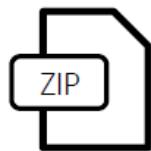
프로젝트_최종_최종2_마지막_진짜제출용.zip



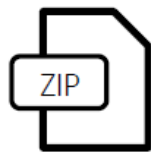
01 왜 쓰는가?



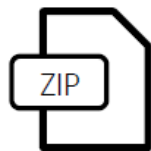
프로젝트_20160401.zip



프로젝트_20160405.zip

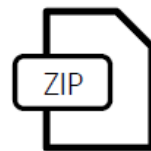


프로젝트_20160510.zip

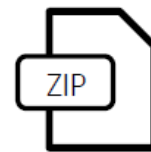


프로젝트_20160514.zip

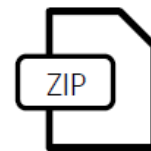
하지만 혼자가 아니라면 어떻게 될까?



프로젝트_20160401_이준영.zip



프로젝트_20160401_송태웅.zip



프로젝트_20160401_이준영+송태웅_통합.zip

소스코드에서 무엇이 변경되었는지?

누가 소스코드를 변경했는지?

언제 소스코드가 변경되었는지?

왜 그렇게 바뀐건지?

이전 버전으로의 복구는 어떻게?

...

그래서 씁니다. **버전 관리 시스템**



소스코드 버전관리 시스템

리누스 토발즈가 개발

빠름!

완전한 분산 환경 지원

대규모 프로젝트에 용이함



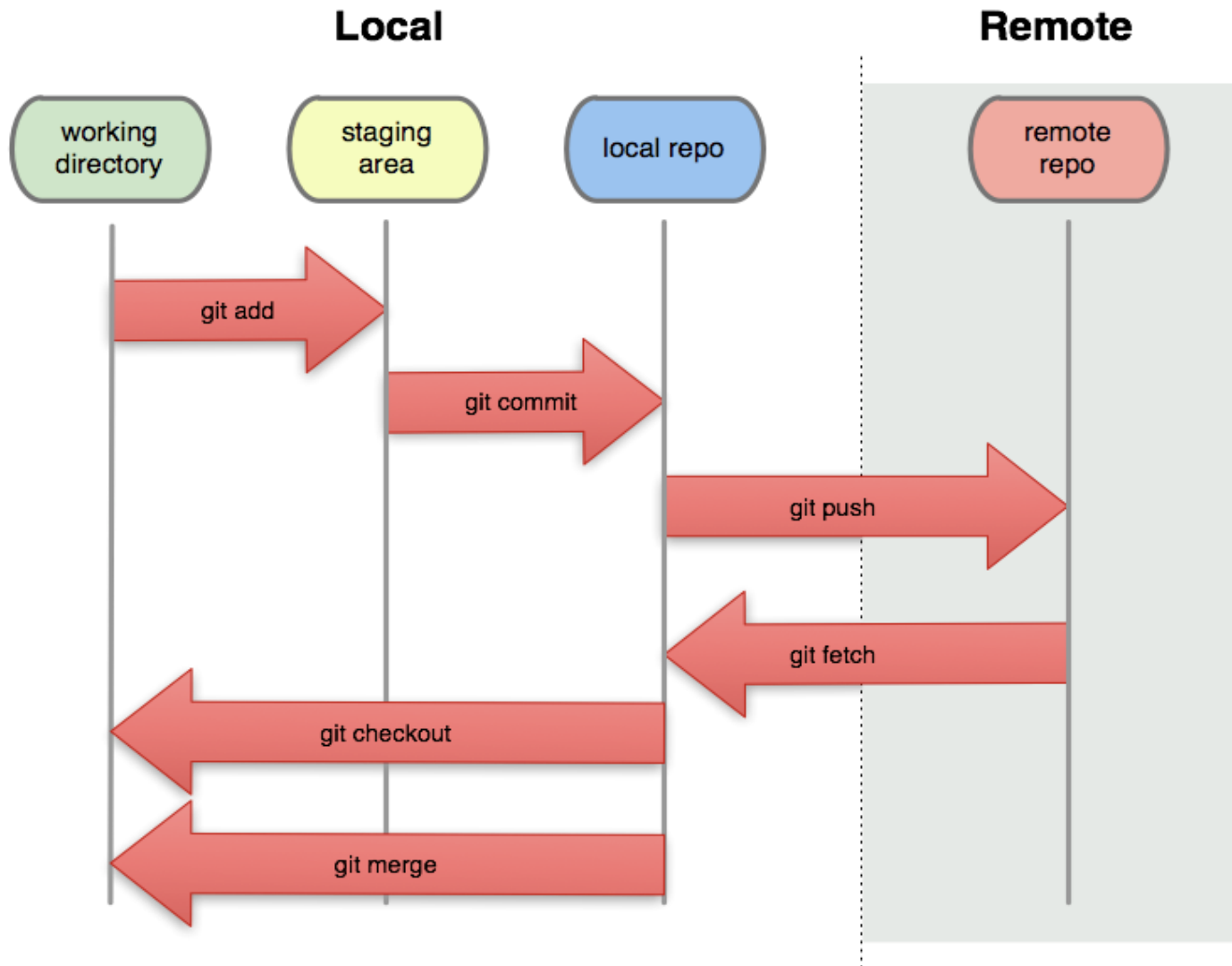
: Git 저장소 호스팅 서비스

- 공개 저장소는 무료로 제공
- 강력한 협업 기능 제공
- 오픈 소스 소프트웨어의 허브
- 다양한 서비스와의 연동
- 기업을 위한 엔터프라이즈 버전 제공



Git 시작하기

01 Git 필수 개념 - Git과 Github의 구조



Working Directory

현재 작업 디렉토리.

Staging Area

커밋 시 반영되는 파일이 보관되어 있는 곳
인덱스영역 이라고도 하며, 커밋하면
local repo(로컬 저장소)로 이동됨

Local repository

Git으로 관리되는 프로젝트의 정보가 저장되는 곳
Git 명령어를 통해 최초에 초기화 하거나
다른 저장소를 Clone 하면 생성된다.

Remote repository

Github에 있는 원격 저장소를 의미하며,
Git 명령어를 통해 local repo(로컬 저장소)와 소통함.

출처: <http://pismute.github.io/whygitisbetter/#any-workflow>

02 Git 필수 개념 - 파일의 세 가지 상태

Modified

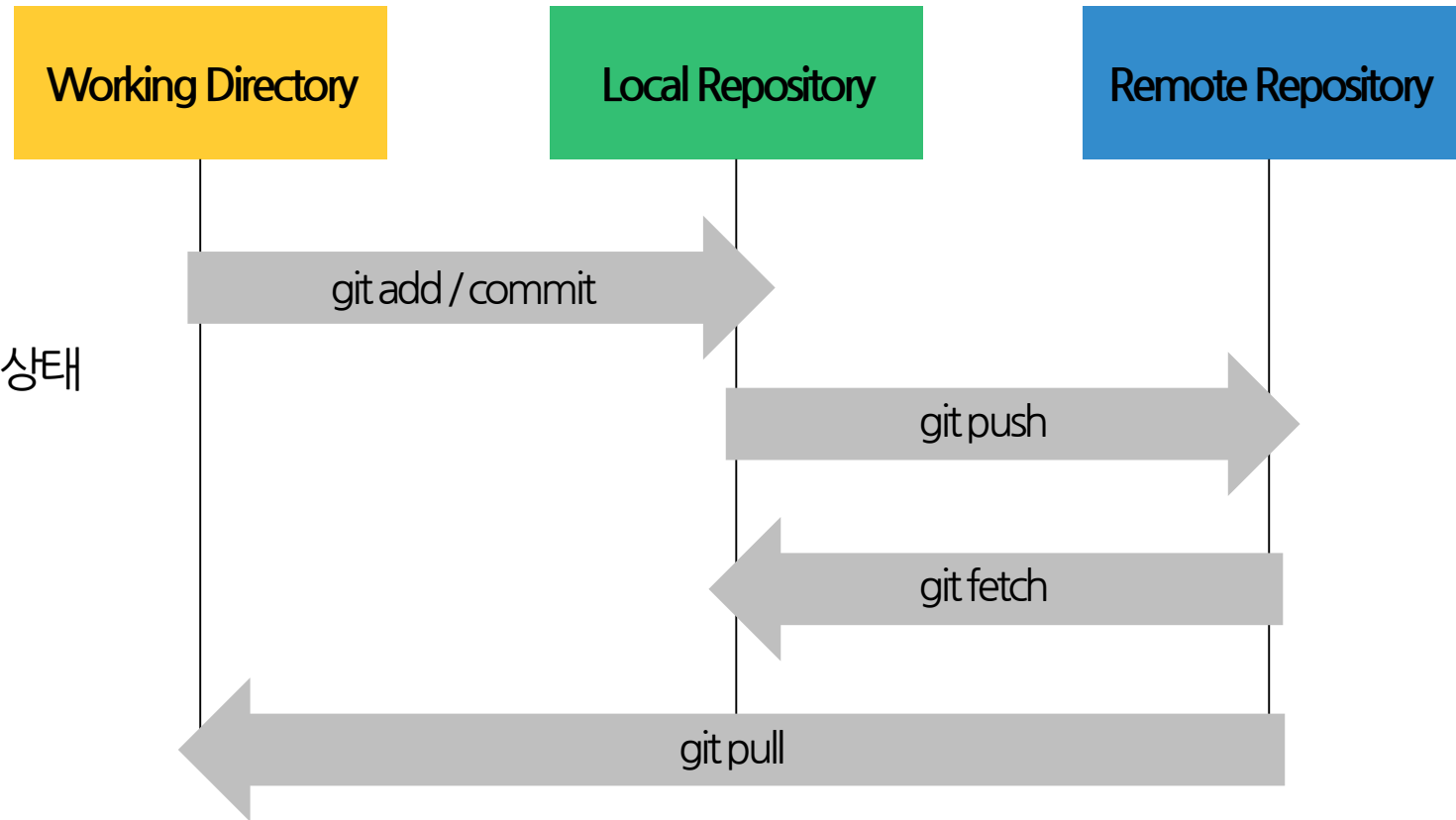
- 수정만 하고 Stage나 커밋하지 않은 상태
- 작업 디렉토리에 존재

Staged

- 현재 수정한 파일을 곧 커밋할 것이라고 표시한 상태
- Staging area에 존재

Committed

- 파일이 로컬에 안전하게 저장된 상태
- 로컬 디렉토리에 존재



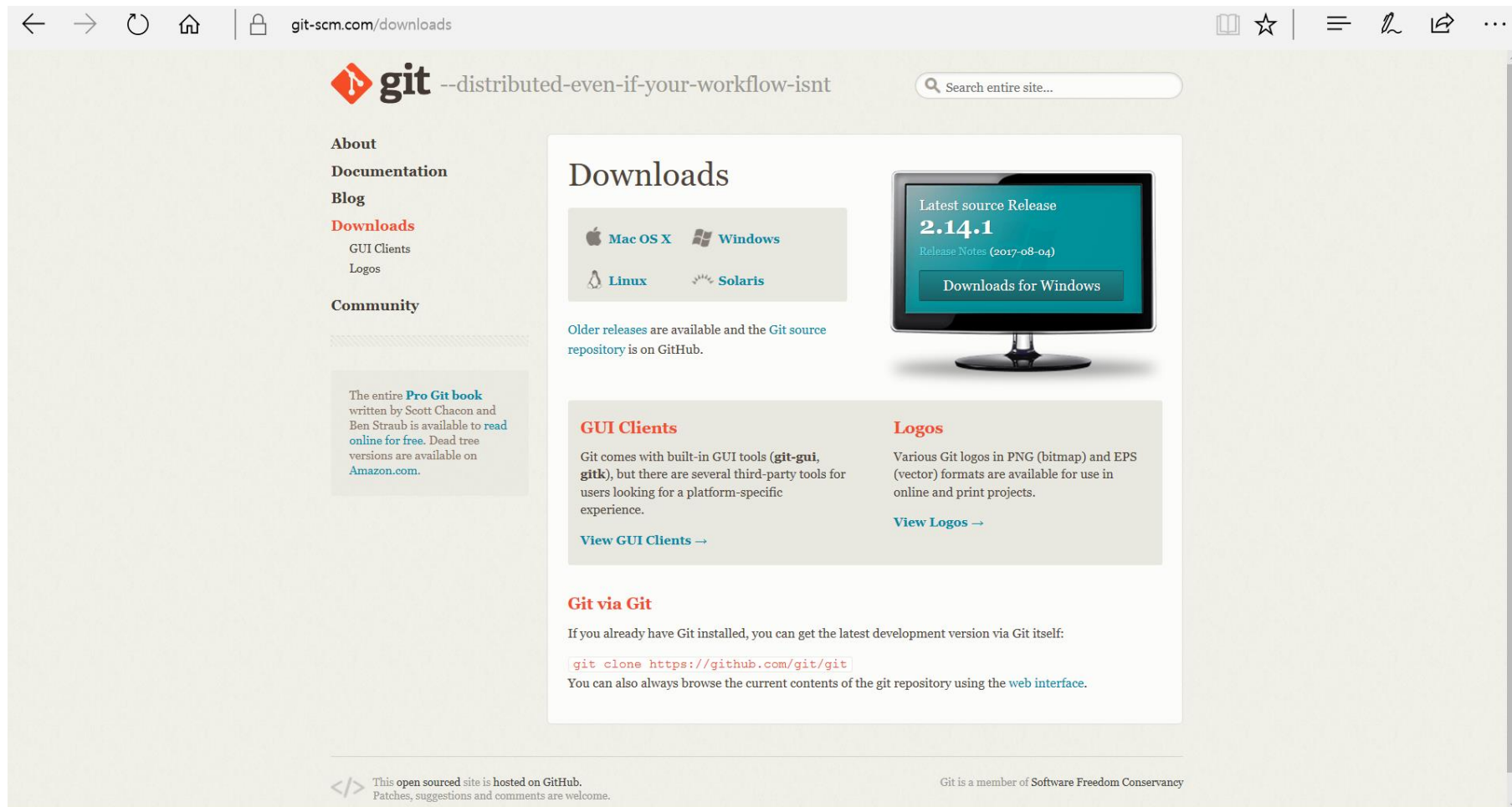


Github에 프로젝트 올려보기

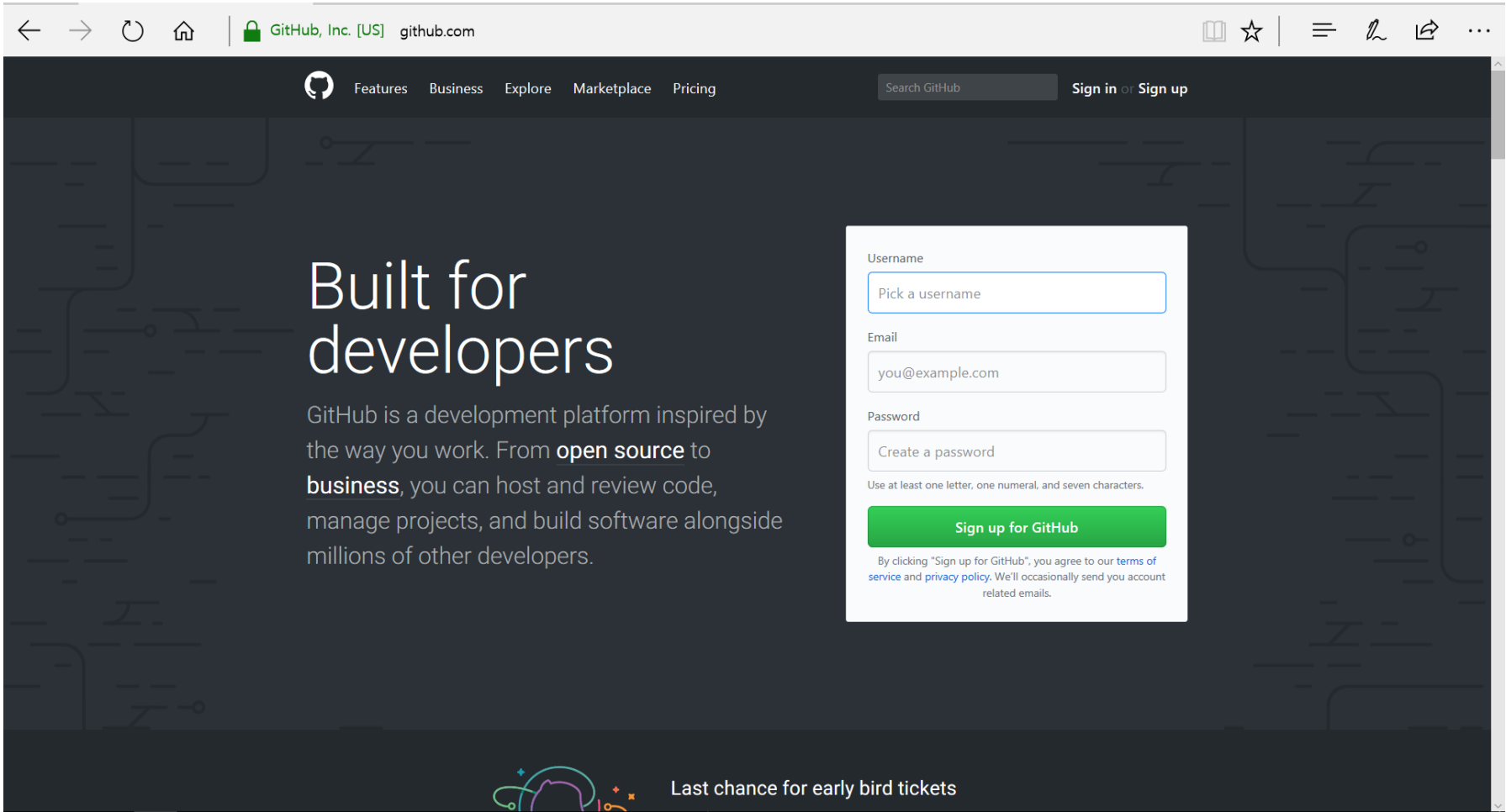
실습은 이렇게 진행할게요

1. Git 클라이언트 설치
2. Github 가입
3. Git에 개인 정보 설정하기
4. Github 온라인 저장소 만들기
5. Git / Github 연동하기
6. Github에 프로젝트 파일 푸시하기

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



<https://github.com/join>



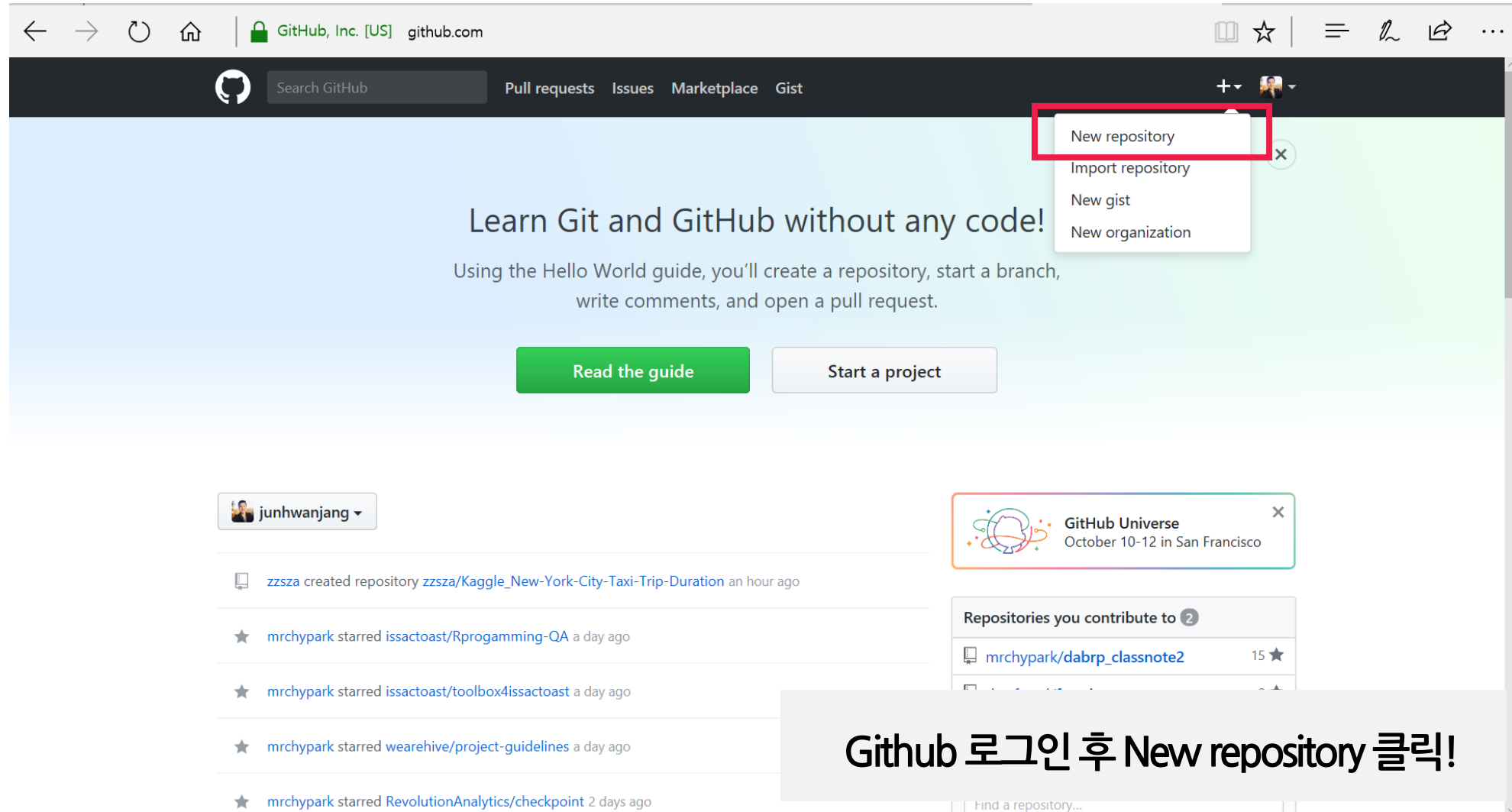
- 1) Git 클라이언트 설치
- 2) Github 가입
- 3) 개인 정보 설정하기

```
git config --global user.name "Your Name Here"  
# git에 username 설정하기
```

```
git config --global user.email "your_email@youremail.com"  
# git에 github ID(e-mail) 설정하기
```

```
C:\Users\Junhwan Jang>git config --global user.name "Junhwan Jang"
```

```
C:\Users\Junhwan Jang>git config --global user.email "junhwan16@gmail.com"
```



Github 로그인 후 New repository 클릭!

The screenshot shows the GitHub 'Create a new repository' page. The browser address bar shows 'github.com/new'. The page title is 'Create a new repository' with a subtitle 'A repository contains all the files for your project, including the revision history.' The form includes fields for 'Owner' (set to 'junhwanjang'), 'Repository name' (set to 'house_sales_KC' with a green checkmark), and 'Description (optional)' (set to 'Kaggle group project'). There are two radio button options for visibility: 'Public' (selected) and 'Private'. A checkbox 'Initialize this repository with a README' is checked. At the bottom, there are dropdowns for '.gitignore' (set to 'None') and 'License' (set to 'MIT License'). A green 'Create repository' button is at the bottom left. Three gray boxes with Korean text are overlaid on the right: '프로젝트 명' (Project Name) next to the repository name field, '공개 여부' (Public/Private) next to the visibility options, and '라이선스 설정' (License Setting) next to the license dropdown.

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner: junhwanjang / Repository name: house_sales_KC

프로젝트 명

Description (optional): Kaggle group project

Public (selected) / Private

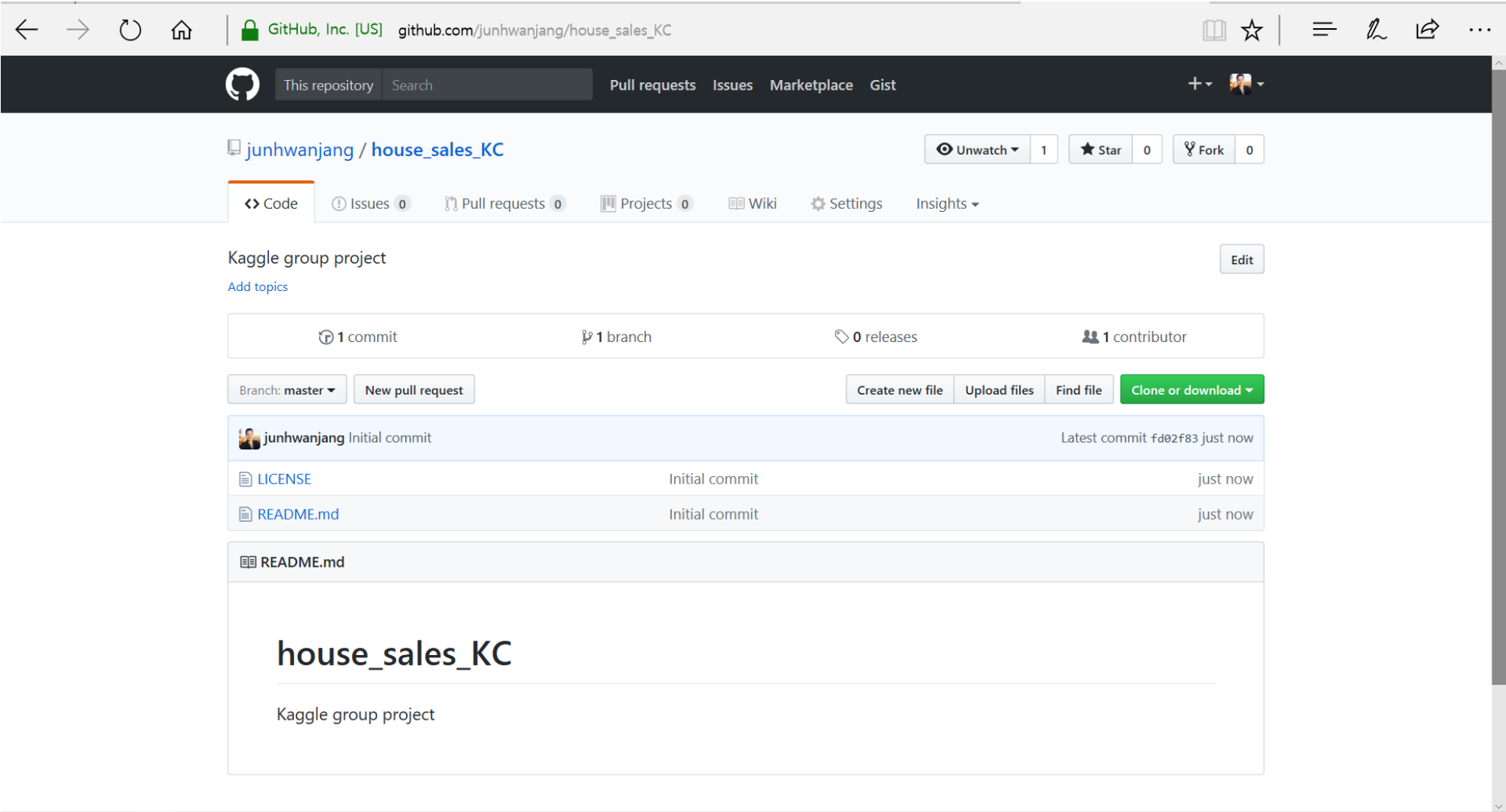
공개 여부

Initialize this repository with a README (checked)

Add .gitignore: None / Add a license: MIT License

라이선스 설정

Create repository



The screenshot shows the GitHub interface for the repository 'junhwanjang/house_sales_KC'. The repository is a Kaggle group project with 1 commit, 1 branch, 0 releases, and 1 contributor. The 'Clone or download' button is highlighted with a red box. Below the button, the 'Clone with HTTPS' option is selected, showing the URL 'https://github.com/junhwanjang/house_sales_KC'. The 'Open in Desktop' and 'Download ZIP' buttons are also visible.

house_sales_KC
Kaggle group project

git clone 프로젝트 URL
github에 있는 파일 내 컴퓨터로 가져오기

```
선택 명령 프롬프트
Microsoft Windows [Version 10.0.15063]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\Junhwan Jang>cd Git

C:\Users\Junhwan Jang\git>git clone https://github.com/junhwanjang/house_sales_KC.git
Cloning into 'house_sales_KC'...
remote: Counting objects: 4, done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (4/4), done.
Checking connectivity... done.

C:\Users\Junhwan Jang\git>cd house_sales_KC

C:\Users\Junhwan Jang\git\house_sales_KC>git remote -v
origin https://github.com/junhwanjang/house_sales_KC.git (fetch)
origin https://github.com/junhwanjang/house_sales_KC.git (push)

C:\Users\Junhwan Jang\git\house_sales_KC>
```

git remote -v
원격저장소 URL이 제대로 설정 되어 있는지 확인

명령 프롬프트

```
C:\Users\Junhwan Jang\git\house_sales_KC>git status
On branch master
Your branch is up-to-date with 'origin/master'.
Untracked files:
  (use "git add <file>..." to include in what will be committed)

    kaggle_house_sales_in_KC_project.R

nothing added to commit but untracked files present (use "git add" to track)
C:\Users\Junhwan Jang\git\house_sales_KC>git add kaggle_house_sales_in_KC_project.R
C:\Users\Junhwan Jang\git\house_sales_KC>git commit -m "my first commit"
[master 095b237] my first commit
 1 file changed, 98 insertions(+)
 create mode 100644 kaggle_house_sales_in_KC_project.R
C:\Users\Junhwan Jang\git\house_sales_KC>git push
git: 'credential-cache' is not a git command. See 'git --help'.
Counting objects: 3, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 1.29 KiB | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/junhwanjang/house_sales_KC.git
 fd02f83..095b237  master -> master
C:\Users\Junhwan Jang\git\house_sales_KC>_
```

git status

git add 파일명

git commit -m “메세지명”

git commit -am “메세지명”

git push

```
명령 프롬프트
C:\Users\Junhwan Jang\git\house_sales_KC>git status
On branch master
Your branch is up-to-date with 'origin/master'.
Untracked files:
  (use "git add <file>..." to include in what will be committed)

    kaggle_house_sales_in_KC_project.R

nothing added to commit but untracked files present (use "git add" to track)
C:\Users\Junhwan Jang\git\house_sales_KC>git add kaggle_house_sales_in_KC_project.R
C:\Users\Junhwan Jang\git\house_sales_KC>git commit -m "my first commit"
[master 095b237] my first commit
1 file changed, 98 insertions(+)
create mode 100644 kaggle_house_sales_in_KC_project.R
C:\Users\Junhwan Jang\git\house_sales_KC>git push
git: 'credential-cache' is not a git command. See 'git --help'.
Counting objects: 3, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 1.29 KiB | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/junhwanjang/house_sales_KC.git
fd02f83..095b237 master -> master
C:\Users\Junhwan Jang\git\house_sales_KC>
```

git status
working directory 내 파일상태 확인

git add 파일명
working directory → staging area 로 이동

git commit -m “메세지명”
staging area → local repository 로 이동
commit은 반드시 수정된 단계별로 진행

git commit -am “메세지명”
add와 commit을 동시에 할 수 있다.

git push
local repository → remote repository 로 이동

← → ↺ 🏠 | GitHub, Inc. [US] github.com/junhwanjang/house_sales_KC

This repository Search Pull requests Issues Marketplace Gist

junhwanjang / house_sales_KC

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Settings Insights

Kaggle group project Edit

Add topics

2 commits 1 branch 0 releases 1 contributor MIT

Branch: master New pull request Create new file Upload files Find file Clone or download

junhwanjang my first commit Latest commit 095b237 4 minutes ago

LICENSE	Initial commit	an hour ago
README.md	Initial commit	an hour ago
kaggle_house_sales_in_KC_project.R	my first commit	4 minutes ago

README.md

house_sales_KC

Kaggle group project

프로젝트 파일 푸시 끝!

불만한 자료들

[튜토리얼]

Git 간편안내서 - 어렵지않아요

<http://rogerdudler.github.io/git-guide/index.ko.html>

완전초보를 위한 깃허브

<https://nolboo.kim/blog/2013/10/06/github-for-beginner/>

Pro-Git 공식 매뉴얼

<https://git-scm.com/book/ko/v2>

[한글 슬라이드]

버전관리를 들어본적 없는 사람들을 위한 DVCS - Git

<https://www.slideshare.net/ibare/dvcs-git>

A successful git branching model (번역본)

<http://dogfeet.github.io/articles/2011/a-successful-git-branching-model.html>