

Git을 사용한 버전 관리

Git 이란?



```
/* elice */
```

커리큘럼

1

Git이란?

프로젝트 협업시 많이 사용되는 Git에 대해 알아보고,
Git을 사용하는 이유에 대해 알아보니다.

2

Git 시작하기

Git 명령어에 대해 배우고,
이를 활용하여 나만의 Git 저장소를 만들어 봅니다.

3

Git 가지 치기

동일한 소스코드를 함께 공유하고 다루게 도와주는
Git의 가지 치기에 대해 알아보고,
마주칠 수 있는 문제에 대해 알아보겠습니다.

커리큘럼

4

Git 원격 저장소

원격 저장소를 연결하여 이용하는 방법을 배워봅시다.
이를 이용해, 서버에 있는 저장소를 이용해 다른 사람과
협업하는 방법을 배워보겠습니다.

*

Git 파고들기 부록

Git에 대한 심층적 이해를 돕기 위한 부록입니다.

수강 대상



정보기술 직무 종사자



Git을 통한 **프로젝트 관리**를 하고자 하는 개발자



Github와 같은 Git 호스팅 서비스를 이용하여 **활용**하고자 하는 개발자

수강 목표

Git을 사용하는 **이유**를 설명할 수 있습니다.

Git 환경을 **구성**할 수 있습니다.

Git을 사용하여 프로젝트를 효율적으로 **관리**할 수 있습니다.

목차

1. Git을 사용하는 이유
2. Git의 특징
3. Git 설치와 초기 설정
4. Git 저장소 생성

효율적인 협업



효율적인 협업

철수

comment.js 오류 사항 수정 했어.

영희

어? 내가 수정한 부분은 다 어디갔지?

효율적인 협업



쉬운 버전관리



comment.js



comment.js.bak

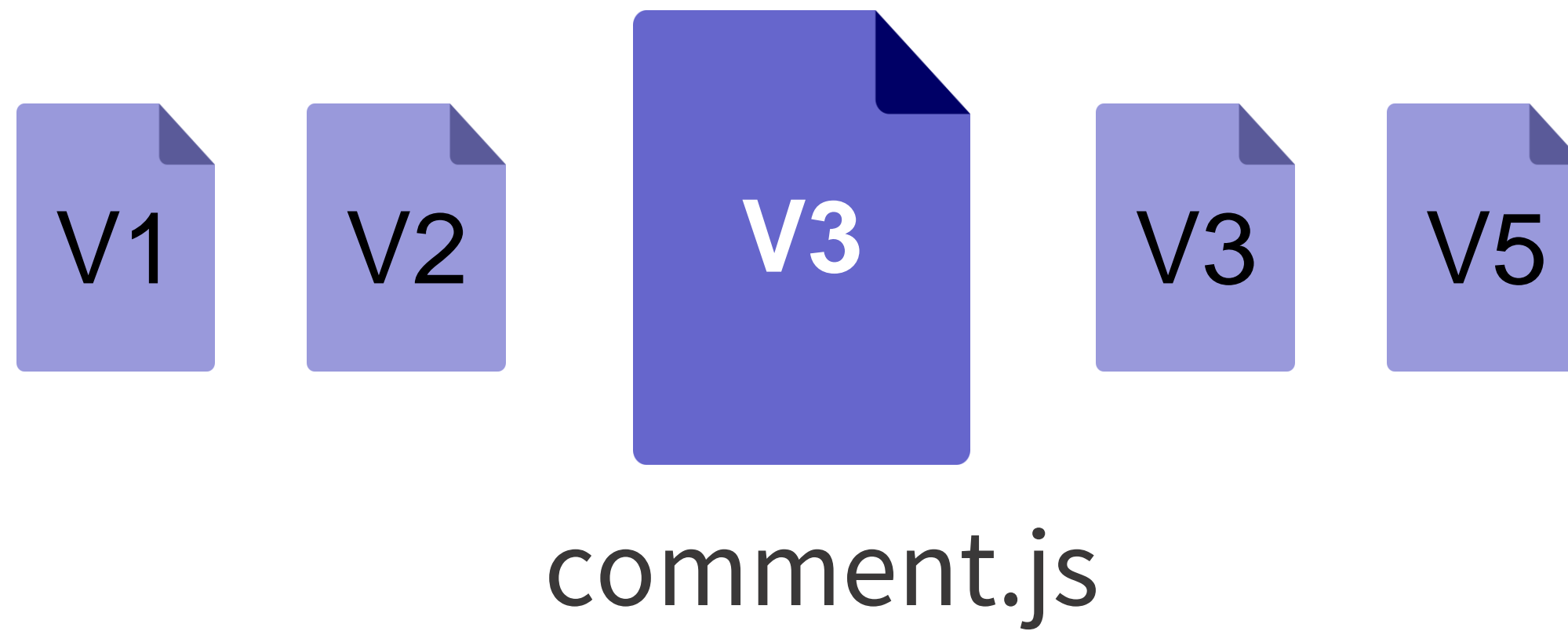


comment.js.bak2



comment.js.bak-180201

쉬운 버전관리



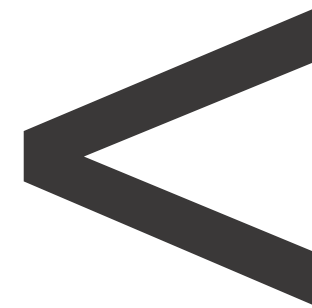
Git의 특징

1. 가지 치기와 병합



그림 git-scm.com

2. 가볍고 빠르다



git

3. 분산 작업

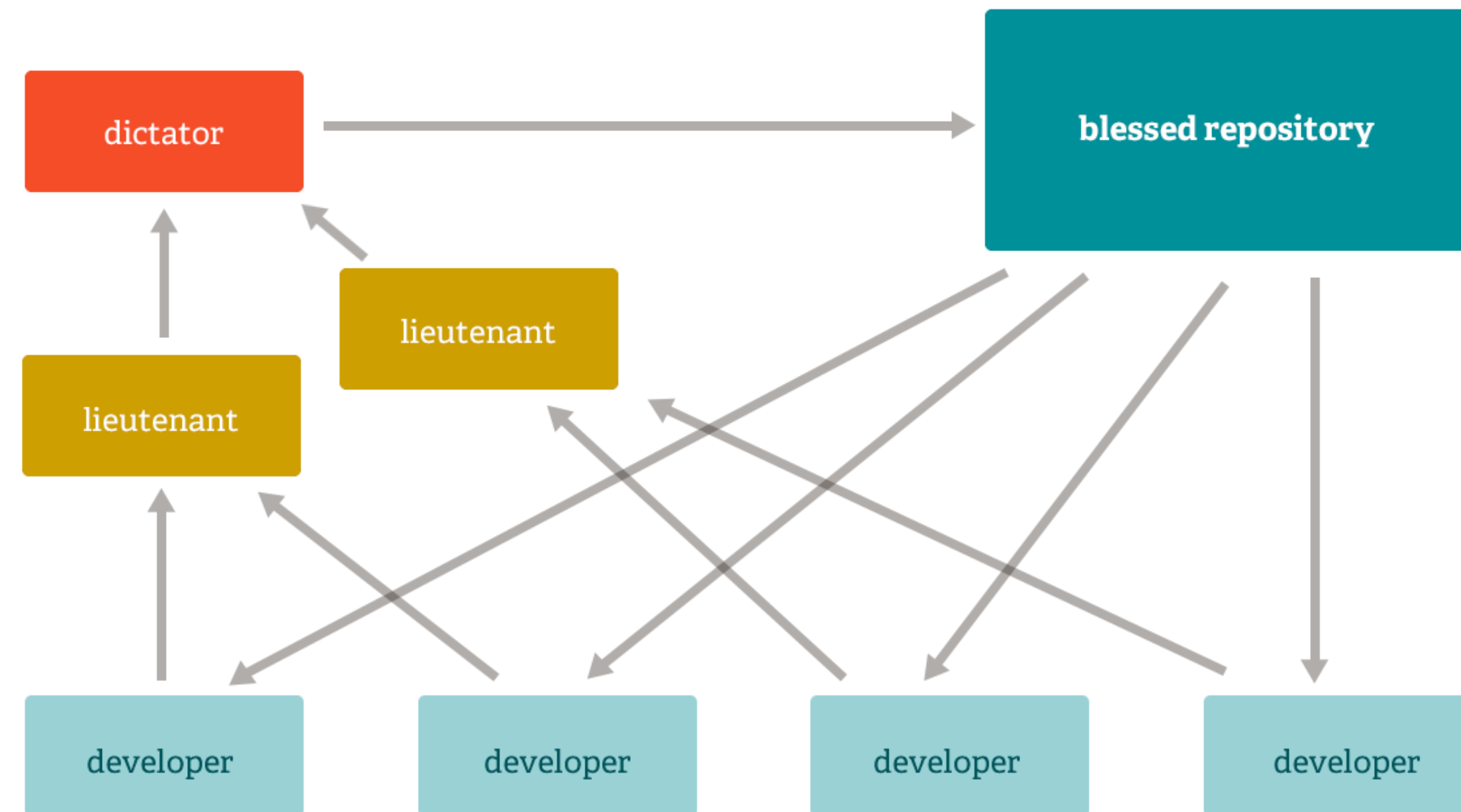


그림 git-scm.com

4. 데이터 보장

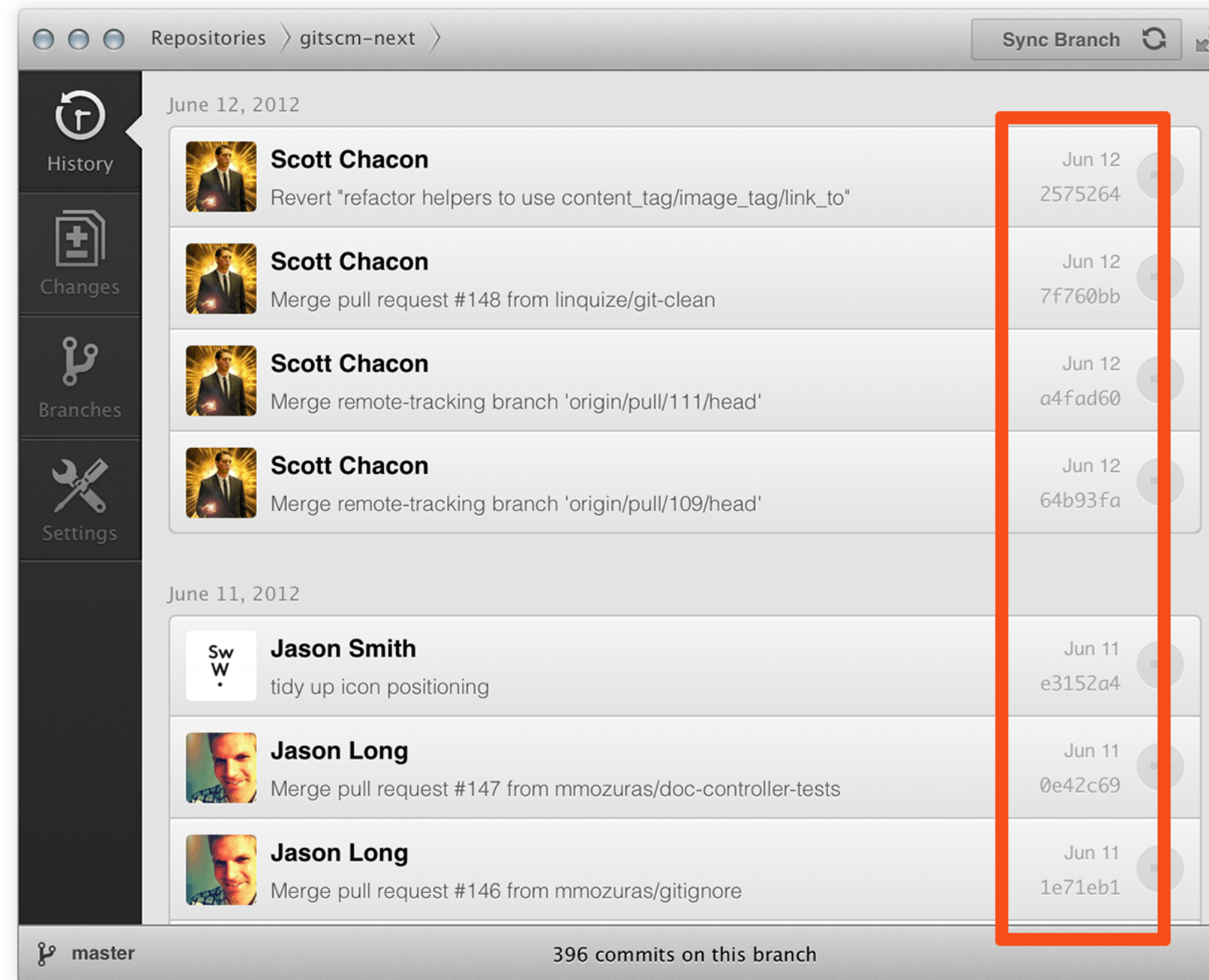


그림 git-scm.com

5. 준비 영역 (Staging area)

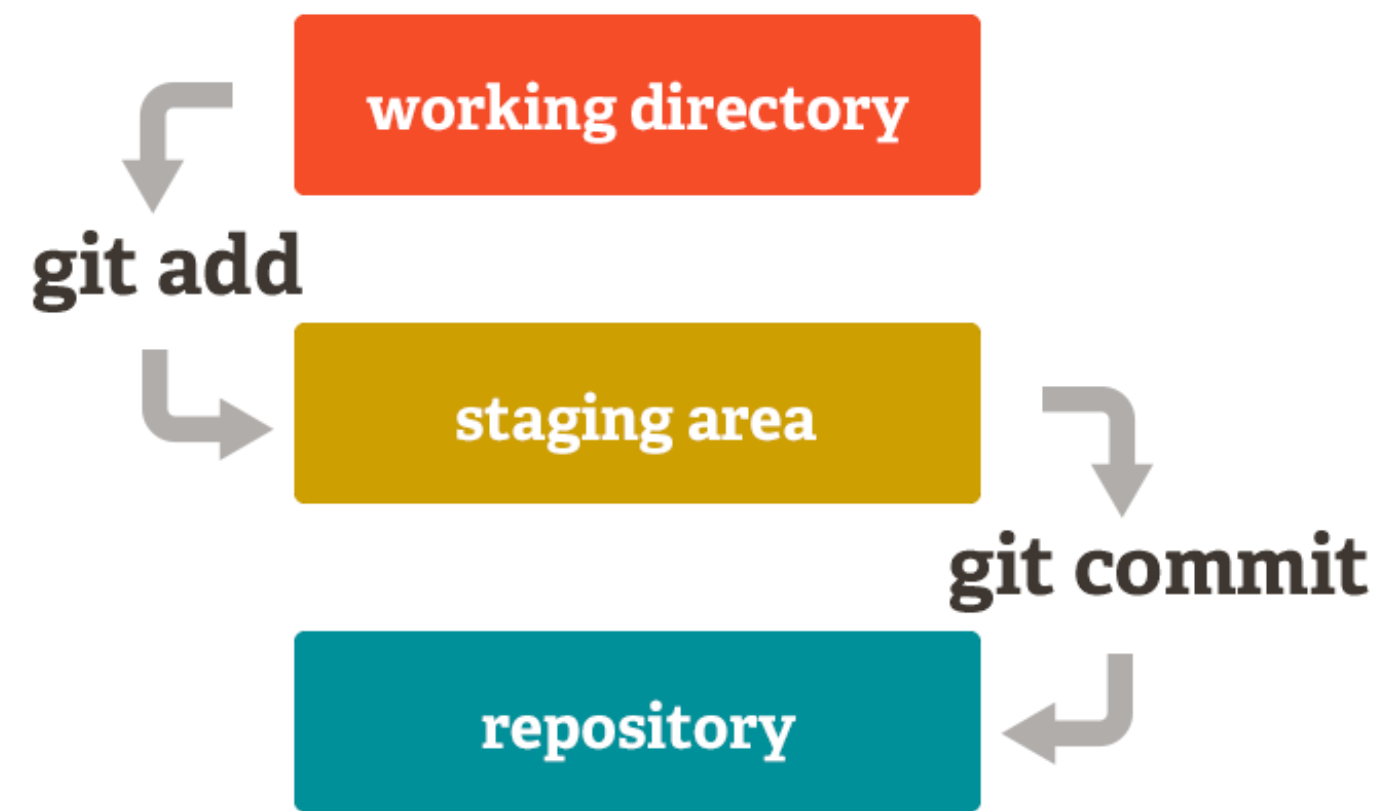


그림 git-scm.com

6. 오픈 소스



Git 호스팅 서비스



Git 설치와 초기 설정

Git 설치

1. Git 설치 여부 확인

- Linux 또는 macOS 환경에서는 대부분 이미 Git이 설치되어 있습니다.
- 다음 화면을 참고하여 Terminal 실행 후 Git 명령어를 실행해보세요.
- Windows 환경에서는 다음 단계로 넘어가주세요.

Git 설치

Terminal 실행, `git` 입력 후 엔터

```
$ git
```

```
usage: git [--version] [--help] [-C <path>] [-c <name>=<value>]
      [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
      [-p | --paginate | --no-pager] [--no-replace-objects] [--bare]
      [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
      <command> [<args>]
```

These are common Git commands used in various situations:

위와 같이 실행되지 않는다면 다음 단계에서
Git을 설치해주세요

Git 설치

2. Git 설치

- 아래의 사이트 접속 후 본인에게 맞는 설치 파일 다운로드

`https://git-scm.com/downloads`

- 다운로드 된 설치 파일 실행

Git 설치

3. Git 실행

- Linux 또는 macOS
Terminal 실행
- Windows
시작 메뉴 → Git → Git Bash 실행

Git 설치

4. Git 설치 확인

- Git이 정상적으로 설치되었나요?
아래의 명령어를 입력하여 Git 버전을 확인해보세요.
(정확한 버전은 아래와 달라도 상관 없습니다)

```
$ git --version
```

```
git version 2.17.2 (Apple Git-113)
```

Git 초기 설정

1. 사용자 정보 설정

- 저장소에 코드를 반영할 때 등록될 사용자 정보를 설정합니다.

```
$ git config --global user.name "elice"
```

```
$ git config --global user.email gitaccount@elice.com
```

Git 초기 설정

1. 사용자 정보 설정

- 프로젝트마다 다른 사용자 정보를 지정하고 싶으면
저장소 생성 후 `--global` 옵션을 빼고
실행해주세요.

```
$ git config user.name "elice"
```

```
$ git config user.email gitaccount@elice.com
```

Git 초기 설정

2. 설정 정보 확인

- 아래의 명령을 실행하여 앞에서 설정한 내용을 확인해보세요.

```
$ git config --list  
credential.helper=osxkeychain  
user.name=elice  
user.email=gitaccount@elice.com  
...
```

Git 저장소 생성

Git 저장소 생성

```
Git init
```

기존의 디렉토리를 git repository로 설정

기존 디렉토리 사용

Git을 사용할 프로젝트 폴더로 이동 후
아래의 명령어를 실행해주세요.

```
$ git init
```

```
Initialized empty Git repository in /Users/heeguchi/git_test/.git/
```

기존 디렉토리 사용

프로젝트 디렉토리에 `.git` 디렉토리가 생성되며
저장소 생성이 완료되었습니다. 간단하죠?

```
$ ls -al
drwxr-xr-x  3 heeguchi  staff   96 11 11 21:45 .
drwx-----+ 5 heeguchi  staff 160 11 11 21:45 ..
drwxr-xr-x 10 heeguchi  staff 320 11 11 21:45 .git
```




/*elice*/

contact@elice.io