

# Git / Github

Git/Github 을 이해하고  
Blockchain 프로젝트 만들어보기

- 2018.08.06 -

# 발표자 - 김종광



- ▶ 이름 : 김종광
- ▶ 소속
  - Manoshare - 대표
  - 이더리움 연구회 - Staff
  - 한국센차유저그룹 커뮤니티 - 운영자
  - (전) KossLab - 수석연구원
  - (전) 소프트웨어 마에스트로 - 멘토
- ▶ SNS
  - Company : <https://manoshare.com>
  - E-Mail : [kim@jongkwang.com](mailto:kim@jongkwang.com)
  - Facebook
    - > <https://www.facebook.com/kimjongkwang>

공개 SW

# Why? Open Source

- ▶ 공개된 Source Code
- ▶ 코드만 공개하면 공개SW 인가?
  - Android
  - Tizen
  - 공개SW 개발자 대회
- ▶ 공개된 Source Code 가 있으면 사용 할 수 있는가?
  - 대학교 과제를 Github 에 올려 놓으면 Open Source?

# Why? Open Source

› 모든것이 공개되어야 Open Source이다

- 기획 / 의도
- 의사결정
- 메뉴얼
- 소스코드
- 사후관리

# Why? Open Source

## ▶ Why? Open Source?

- 1 vs. 70억
  - 사용자 환경
  - Test
  - 문서화 / 번역
- 안할수가 없다
- 리눅스 개발자가 1,000만명이 넘습니다
  - Open Source 이외에는 답이 없음
- 직원이 10만명인 회사를 운영하는 가이드북이 있다면 구매 하겠다 없으니 소통으로 해법을 찾을 수 밖에 없다. - 코카콜라 회장

# Why? Open Source

## ▶ Open Source 를 하는 이유?

- 개인
  - 여러분?
  - Linus Benedict Torvalds
- 재단
- 삼성
- 구글

# Why? Open Source

## ▶ Google

- white paper 작성
  - MapReduce → Hadoop
  - BigTable → HDFS → HBase
  - Google Code Search → Sourcegraph
  - Borg → Docker
  - TensorFlow??

분산관리 시스템

Download

# Download

- ▶ Git
  - <http://git-scm.com/download/win>

- ▶ SourceTree
  - <https://www.sourcetreeapp.com/>

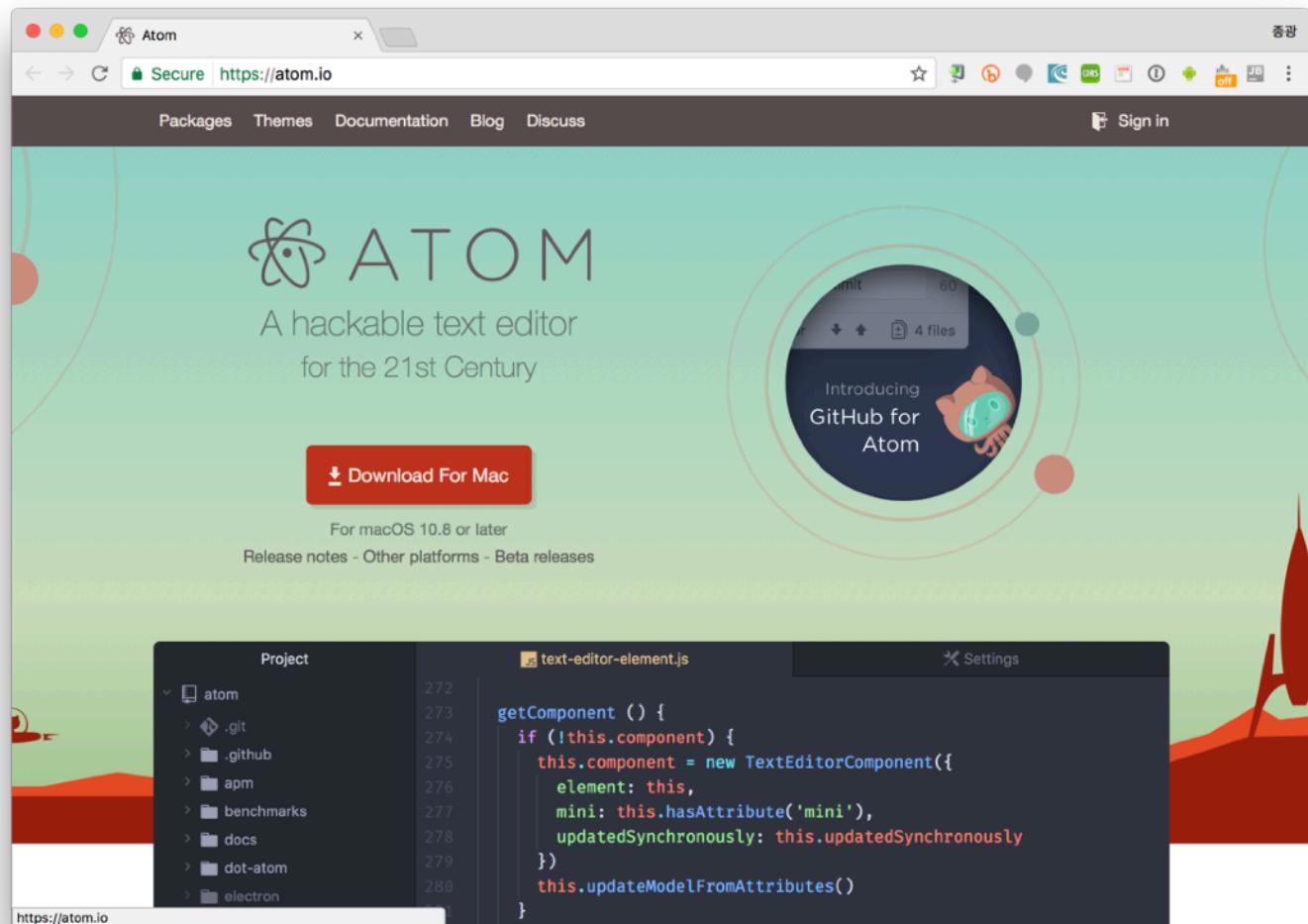
- ▶ ATOM
  - <https://atom.io/>

- ▶ Java
  - <http://www.oracle.com/technetwork/java/javase/downloads/index.html>
  - JDK 다운로드 (가입/로그인 필수)

- ▶ Eclipse
  - <https://www.eclipse.org/downloads/>

# ATOM Install

› <https://atom.io>



# SourceTree Install

› <https://www.sourcetreeapp.com>

The screenshot shows the SourceTree website with the Mac OS X version of the application open. On the left, there's a promotional text: "Simplicity and power in a beautiful Git GUI". Below it are two download buttons: "Download for Mac OS X" and "Also available for Windows". On the right, the SourceTree application window is displayed, showing a Git commit history for a repository named "sourcetree-website (Git)". The commits are listed in a table with columns for Commit ID, Author, Description, and Date. The commits are as follows:

Commit	Author	Description	Date
b7358c7	Rahul Chhab...	[r] master [r] origin/master [r] origin/HEAD Removing ol...	Mar 3, 2016, 11...
bdb8bef	Rahul Chhab...	Merged in update-google-verification (pull request #14)	Feb 18, 2016, 1:3...
dfe975d	Tyler Tadej...	[r] origin/update-google-verification Update google verificati...	Feb 11, 2016, 2:2...
3bc3290	Tyler Tadej...	Replace outdated Atlassian logo in footer with base-64 en...	Feb 11, 2016, 2:1...
dba4719	Tyler Tadej...	Add glistenore	Feb 11, 2016, 1:3...
ff67b45	Mike Minns...	Updated Mac min-spec to 10.10	Feb 15, 2016, 11:...
72d32a8	Michael Min...	Merged in hero_images (pull request #13)	Feb 15, 2016, 10:...
246c4ff	Joel Unger...	[r] origin/hero_images [r] hero_images Used TinyPng to c...	Feb 11, 2016, 3:3...
9d9438c	Joel Unger...	Replacing hero images with new version of SourceTree	Feb 9, 2016, 2:59...
ce75b63	Michael Min...	Merged in bug/date-https (pull request #12)	Feb 15, 2016, 10:...
85367bb	Patrick Tho...	[r] origin/bug/date-https fixed date and https errors	Jan 7, 2016, 12:2...
4f9b557	Joel Unger...	New Favicon	Feb 8, 2016, 3:55...
384e6d5	Rahul Chhab...	[r] origin/search-console-access search console google ver...	Feb 3, 2016, 2:09...
6fa47e9	Mike Minns...	updated to move supported version to OSX 10.9+	Dec 15, 2015, 2:0...
8dd87bb	Mike Minns...	remove extra , when a line is skipped due to empty server	Nov 23, 2015, 2:2...
faa195e	Mike Minns...	Skip records with empty server/user id as gas rejects them	Nov 23, 2015, 2:1...
0cdfe96	Mike Minns...	corrected paths after merge	Nov 23, 2015, 2:0...
051ab1b	Mike Minns...	corrected column counting	Nov 23, 2015, 1:5...
a723bc2	Mike Minns...	Merge branch 'au2gex'	Nov 23, 2015, 1:5...
65fd580	Mike Minns...	deal with invalid instanceids	Nov 23, 2015, 1:5...
500a892	Michael Min...	Merged in au2gex (pull request #11)	Nov 23, 2015, 1:0...

**A free Git client for Windows and Mac**

SourceTree simplifies how you interact with your Git repositories so you can focus on coding. Visualize and manage your repositories through SourceTree's simple Git GUI.

# Git Install

## ▶ Windows

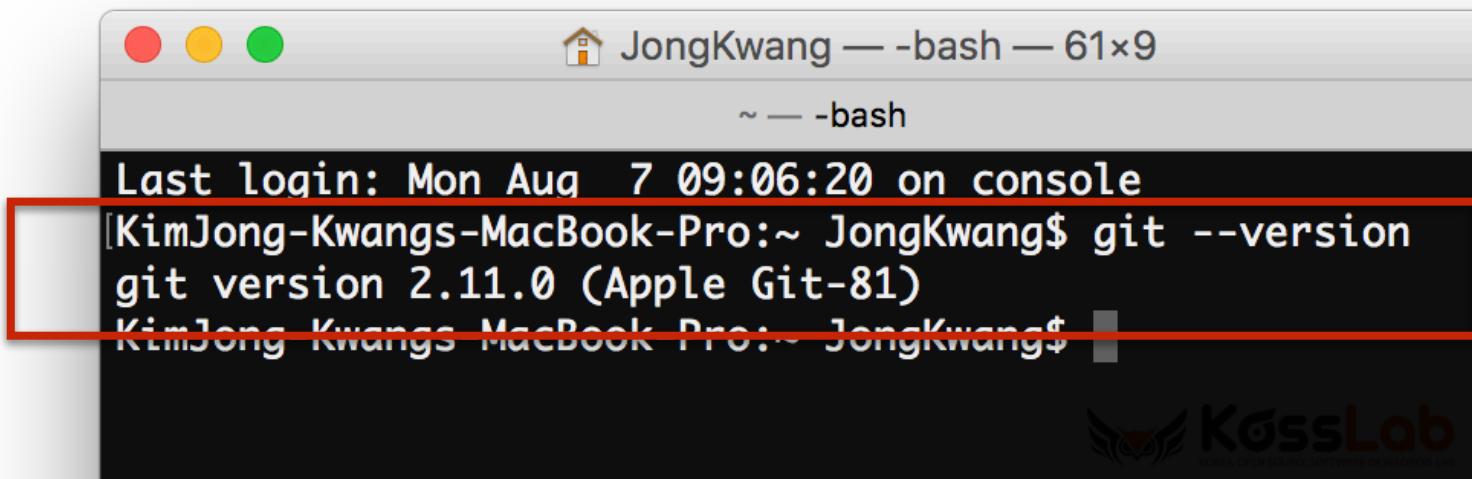
- <http://git-scm.com/download/win>

## ▶ macOS

- Terminal에서 “git” 실행
  - Mavericks(10.9) 부터 없으면 설치된다.
- 직접 설치 : <http://git-scm.com/download/mac>

## ▶ 설치 확인

- Terminal : git --version



The screenshot shows a terminal window titled "JongKwang — -bash — 61x9". The window contains the following text:

```
Last login: Mon Aug  7 09:06:20 on console
[KimJong-Kwangs-MacBook-Pro:~ JongKwang$ git --version
git version 2.11.0 (Apple Git-81)
KimJong Kwangs MacBook Pro:~ JongKwang$ ]
```

A red box highlights the command "git --version" and its output "git version 2.11.0 (Apple Git-81)".

분산관리 시스템

分散관리

시스템

# 버전관리 시스템

## ▶ 게임에서 Save 같은 존재

- RPG 게임에서 Save Point

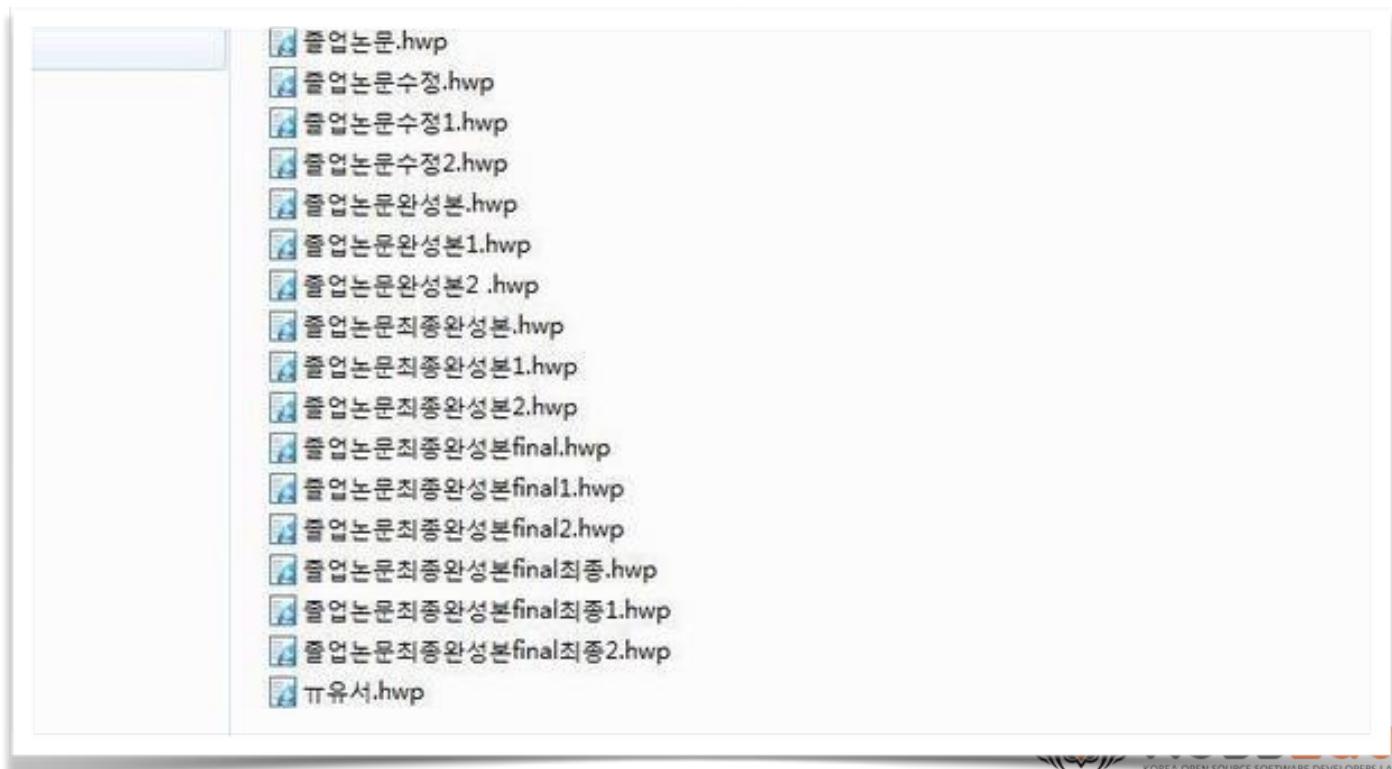
- 게임에서는 Save Point 를 만나야 Save 하지만
- 버전관리 시스템에서는 내가 원할 때 Save 가능



# 버전관리 시스템

## ▶ 우리는 이미 리포트 작성할 때 경험 했다

- 파일에 번호를 부여하여 History를 관리하는 방식은 상당히 훌륭하다
  - SVN에서도 같은 방식을 사용하며
  - SVN에서는 이 번호를 Revision Number 라고 부른다
  - 하지만, 파일이 1개 일 때만 가능하다



# 버전관리 시스템

- ▶ 하지만, 여러 파일이 변경되는 경우에는?
  - 파일 넘버로 관리 할 수 없다

클라우드 백업?



SVN Repository Structure:

- > celebq 389 [svn://svn.jongkwang.com/celebq: celebq]
  - JAX-WS Web Services
  - Java Resources
  - JavaScript Resources
  - build 3
  - docs 254
    - Certificate 254
      - Android 253
    - iOS 254
      - aps\_development.cer 253
      - aps.cer 253
      - Cert\_Development.p12 254
      - Cert\_Production.p12 254
      - CertificateSigningRequest.certSigningRequest 253
      - Key\_Development.p12 254
      - Key\_Production.p12 254
      - Readme.txt 253
      - SocialQ\_Development.mobileprovision 253
      - SocialQ\_Production.mobileprovision 253
    - OAuth 399
      - instagram.txt 399
  - WebContent 389
    - assets 388
      - css 388
      - img 376
      - js 13
        - animheader.js 4
        - bootstrap.js 6
        - contact.js 4
        - countto.js 6
        - flexslider.js 4
        - jquery.js 6
        - masonry.pkgd.min.js 13
        - parallax.js 4
        - plugins.js 4
        - portfolio.js 4
        - scripts.js 4
        - reservationform 6
      - css 388
        - style.css 388
      - inc 343
      - META-INF 100
        - index.html 6
        - MANIFEST.MF 1
        - Setting.jsp 100
      - WEB-INF 369
        - address.jsp 96
        - addressiOS.jsp 326
      - CardConfirm.jsp 377
      - CardDetail.jsp 380
      - CardList.jsp 334
      - Clause.jsp 204
      - Co\_CardCreat\_first.jsp 386

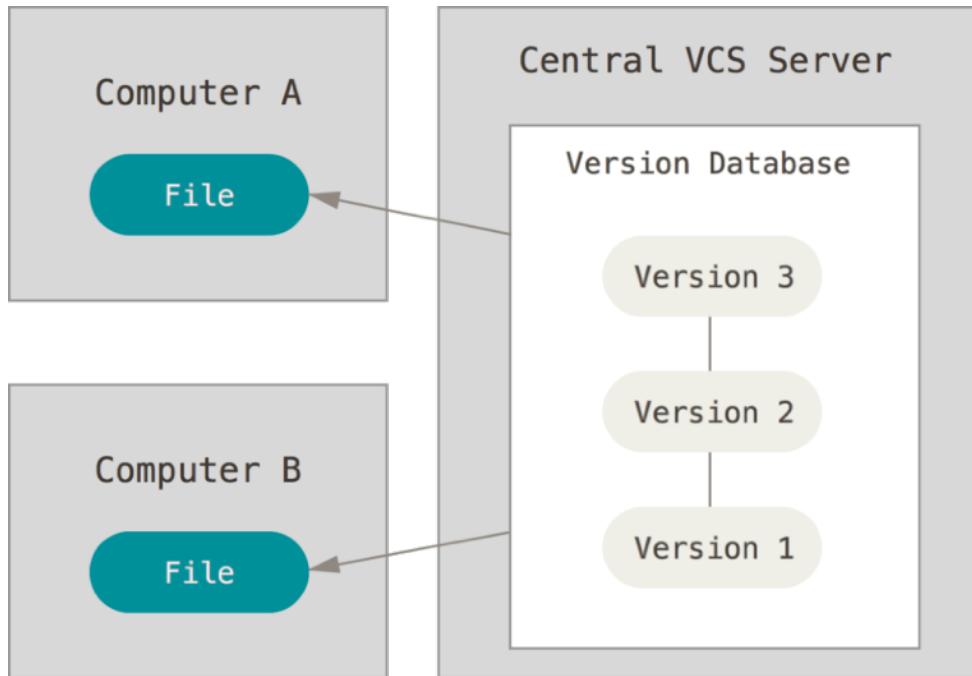
분산관리 시스템

分散관리 시스템  
도구로 이용하자

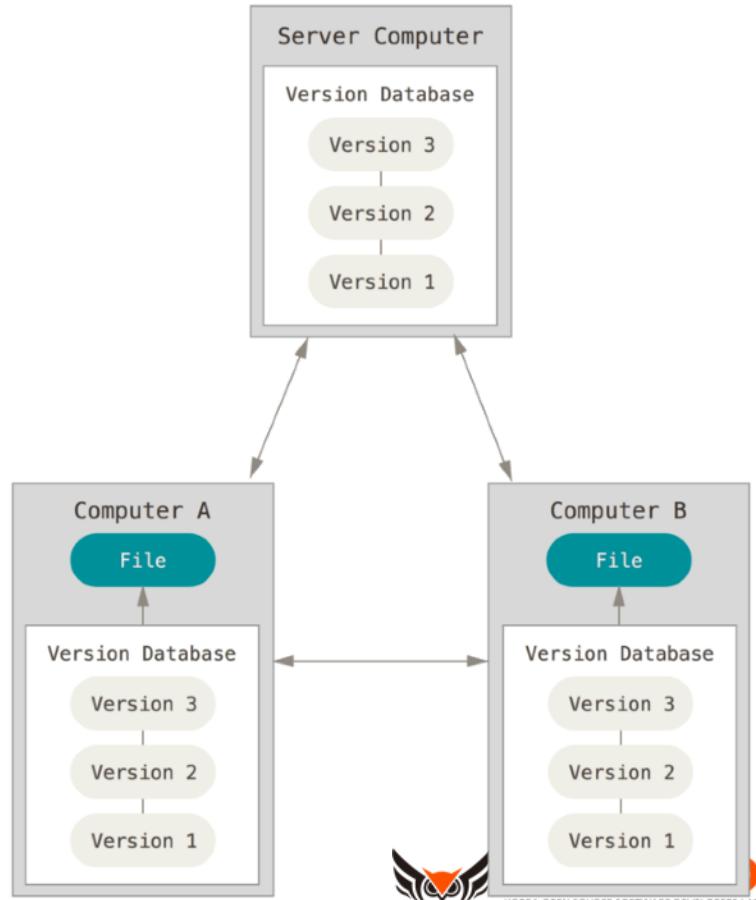
# 버전관리 시스템

- ▶ 버전관리 시스템의 2가지 분류

## 클라이언트-서버

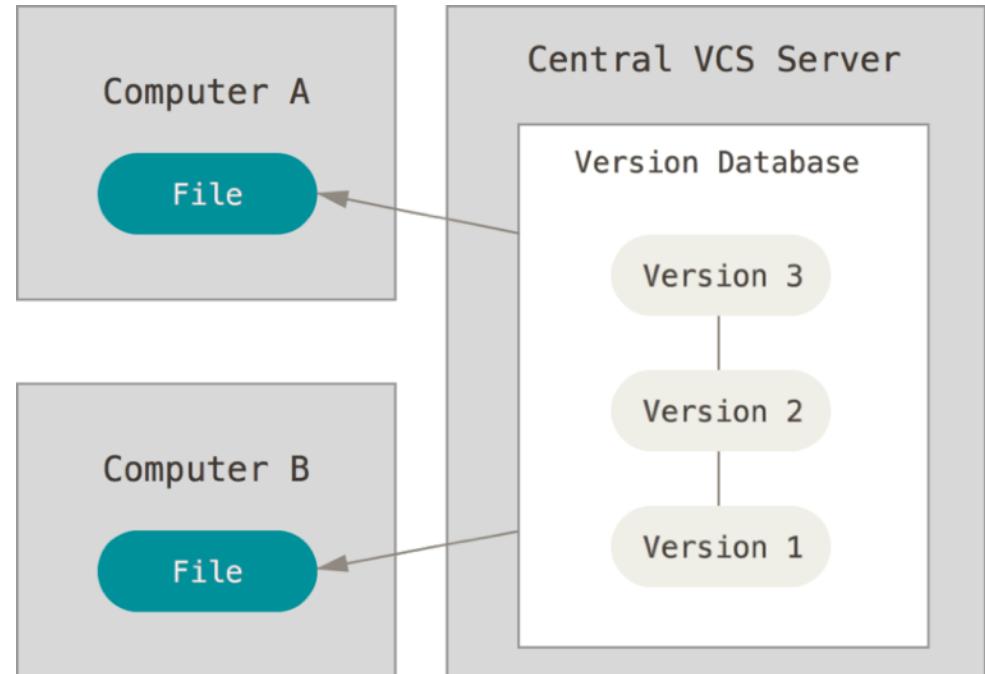


## 분산



# 클라이언트-서버

- › 서버와 클라이언트 구조
- › 서버가 최종 버전을 관리
- › SVN , CVS 등이 여기에 해당
- › 장점
  - 쉽다
- › 단점
  - 서버가 고장나면 끝이다
  - Online 상태가 아니면 작업불가
  - Branch 가 불편하다
  - 느리다 (Online)



# 분산

## 특징

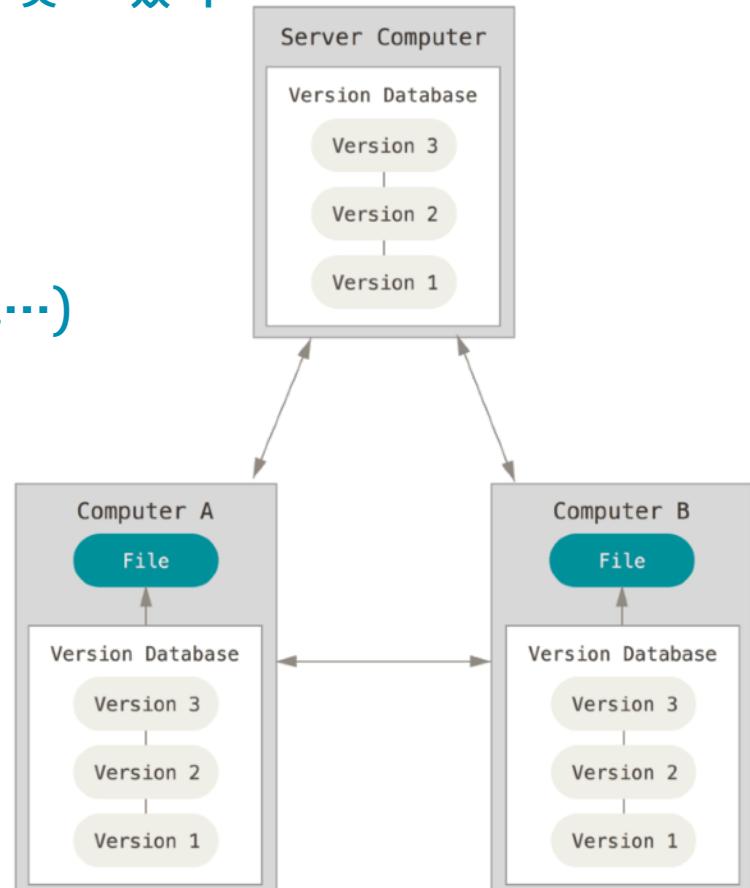
- 모든 클라이언트가(개발자)가 전체 저장소를 갖고 있다
- 전체 저장소의 사본을 모두 갖는다

## 장점

- Offline 작업 가능
- Commit 이 자유롭다 (중간버전 커밋 찍어도...)
- Branch 가 자유롭다
- 빠르다

## 단점

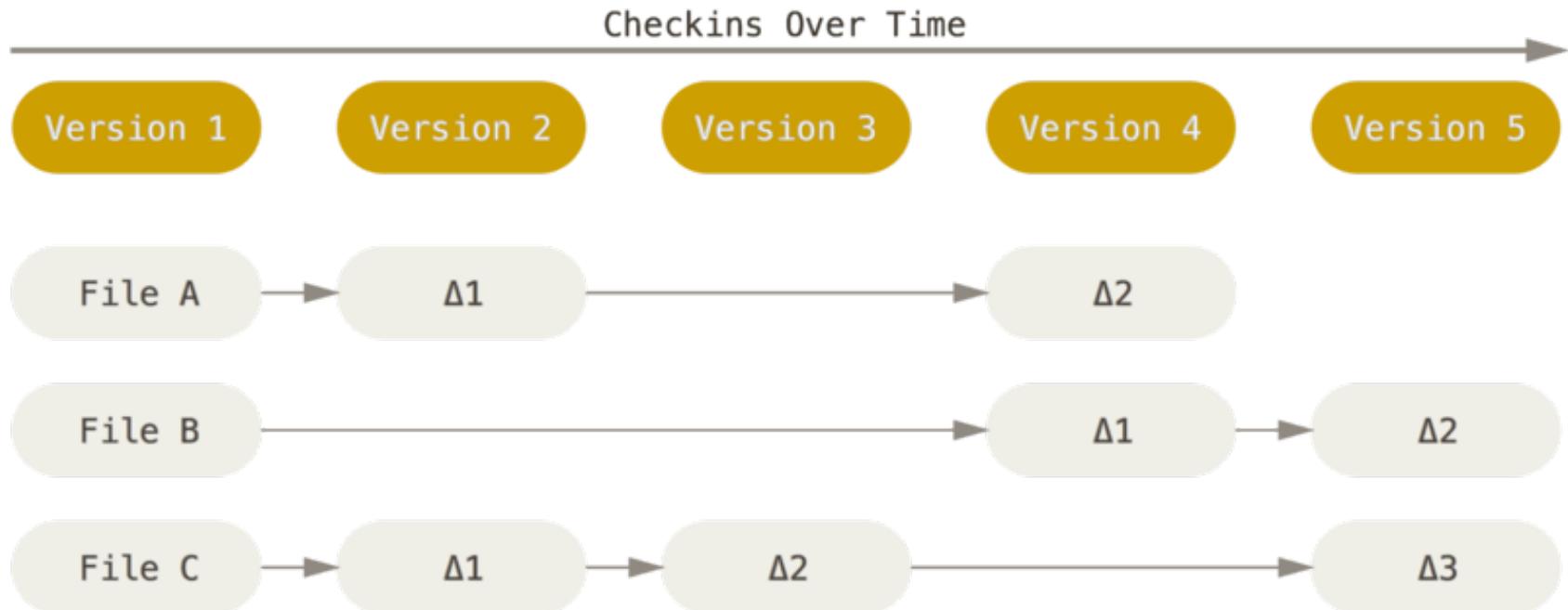
- 조금 어려운 정도
- Client에 저장소를 설치해야 한다



# 데이터를 다루는 차이

## ▶ SVN

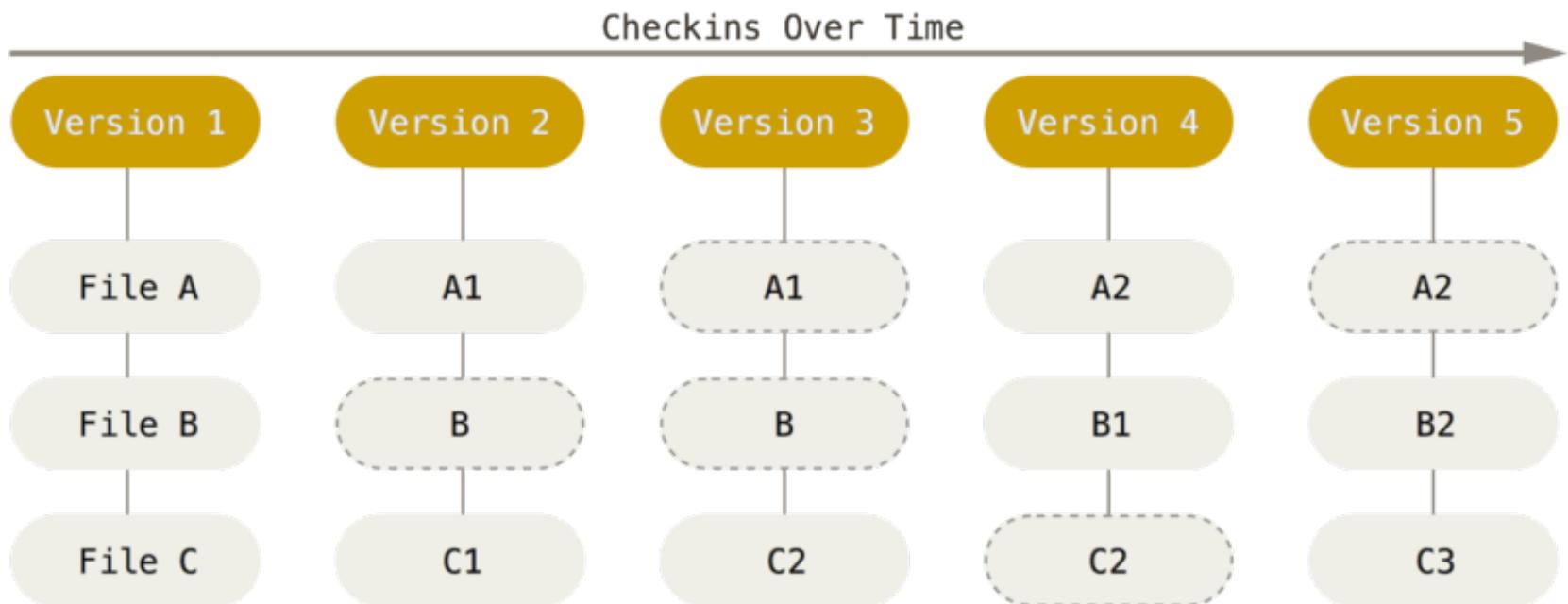
- 파일의 변화를 시간순으로 관리
- 파일들의 집합을 관리



# 데이터를 다루는 차이

## ▶ Git

- 순간의 Snapshot 관리
- 변경이 없으면 Link 만 저장



# 취업가기 - Git History

- ▶ Git 을 만든 사람은?

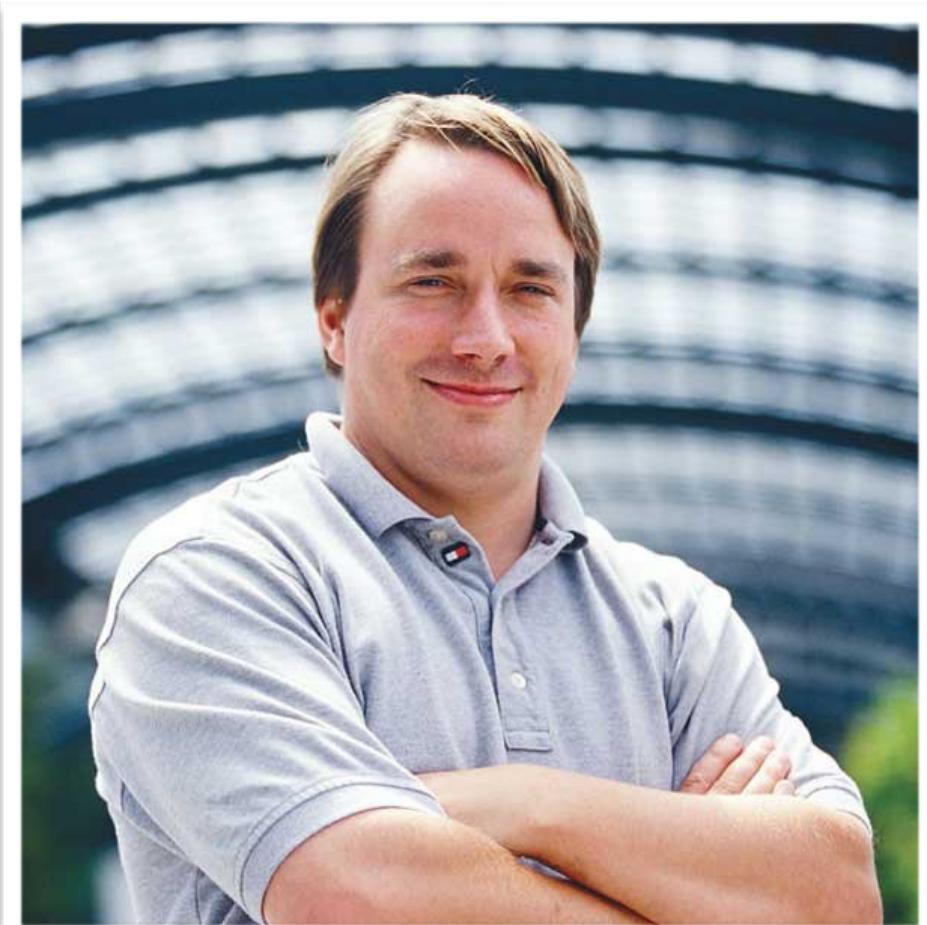
# 취업가기 - Git History

- ▶ Git 을 만든 사람은?



# 취어가기 – Git History

- › 2002년 Linux Kernel 개발을 위해 BitKeeper 사용
- › 2005년 BitKeeper에서 무료 사용에 난색
- › 이에 분노한 리누스 토발즈



# 취미가기 - Git History

## ▶ 2주만에 Git을 만듬

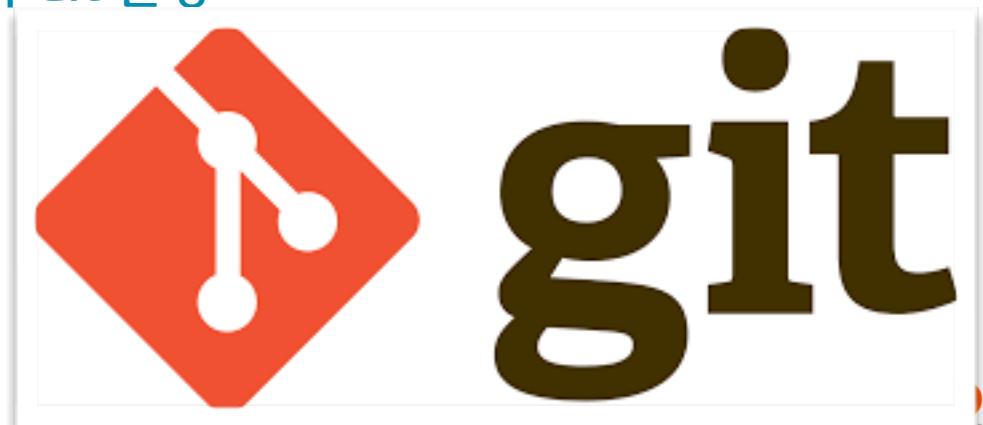
- 처음 용도 : Linux Kernel 버전관리

## ▶ 목표

- 완벽한 분산 환경
- 빠르고 단순하게 수백/수천 개의 Branch 작업을 목표
- 실제로 리눅스 커널 같은 대형 프로젝트에 사용됨

## ▶ 개발기간

- 3일만에 Git 의 버전관리를 Git으로
- 2주만에 여러 Branch 를 병합하며 Git 완성
  - 현재까지 큰 변화 없이 사용됨



# Git의 특징

## ▶ 특징

- 로컬 및 원격 저장소 생성
- 로컬 저장소에 파일 생성 및 추가
- 수정 내역을 로컬 저장소에 커밋
- 파일 수정 내역 비교
- 원격 저장소에 커밋된 수정 내역을 로컬 저장소에 적용
- 로컬 커밋 내용을 원격 저장소로 Push
- Branch 생성
- 브랜치 병합(Merge)
- 브랜치 병합시 충돌 확인

# 개발자만 Git 을 사용해야 하는가?

## ▶ 누구나 사용 가능

- 디자이너 : 이미지 파일
- 기획자 : 발표자료

## ▶ 사례

- 서울 정보소통광장 행정정보 공개
  - <https://github.com/seoul-opengov/opengov>
- 백악관 각종 정보
  - 예산안 : <https://github.com/WhiteHouse/budgetdata>
  - 그외 여러가지
- 고위공직자 재산 공개
  - <https://github.com/codenamu/official-assets-explorer-2017>

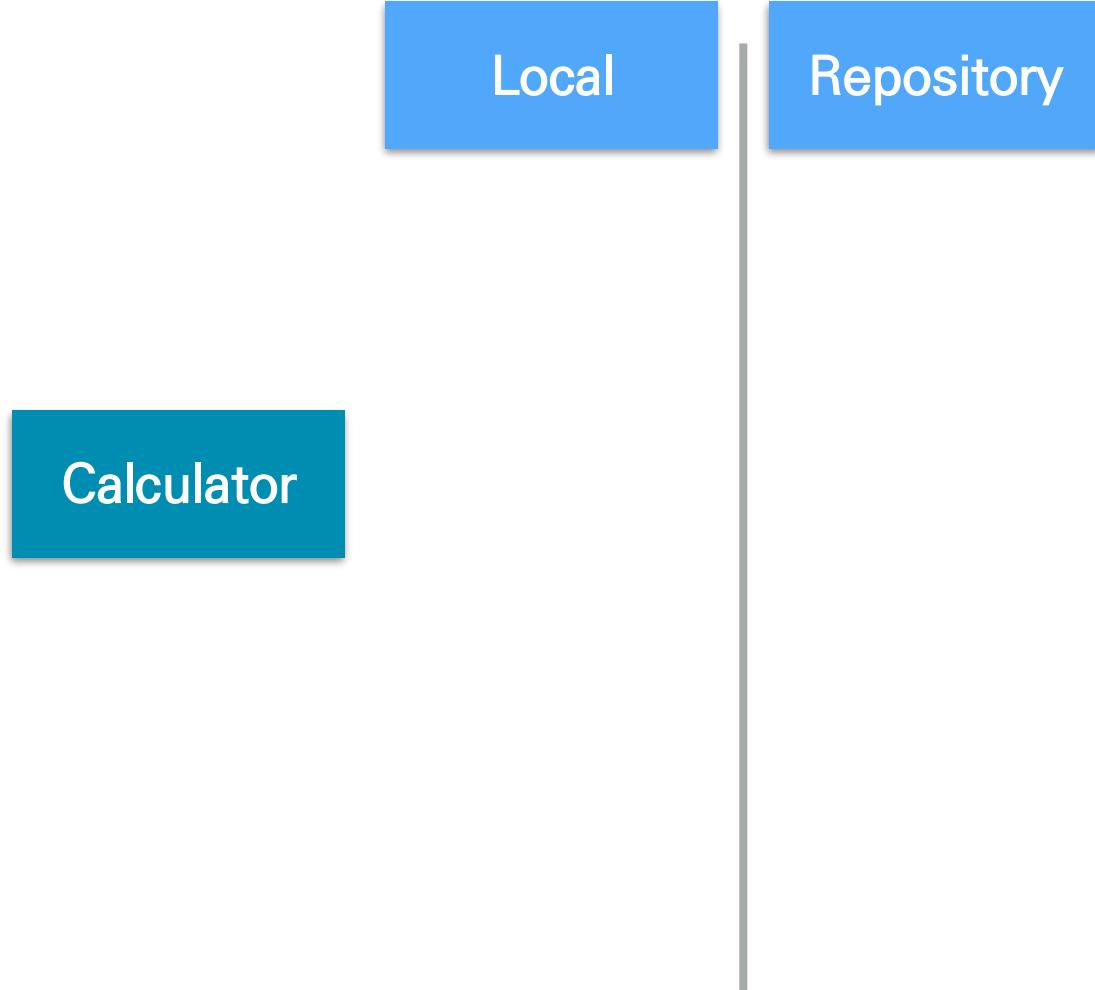
분산관리 시스템

Git

그림설명

# Git - 그림 설명

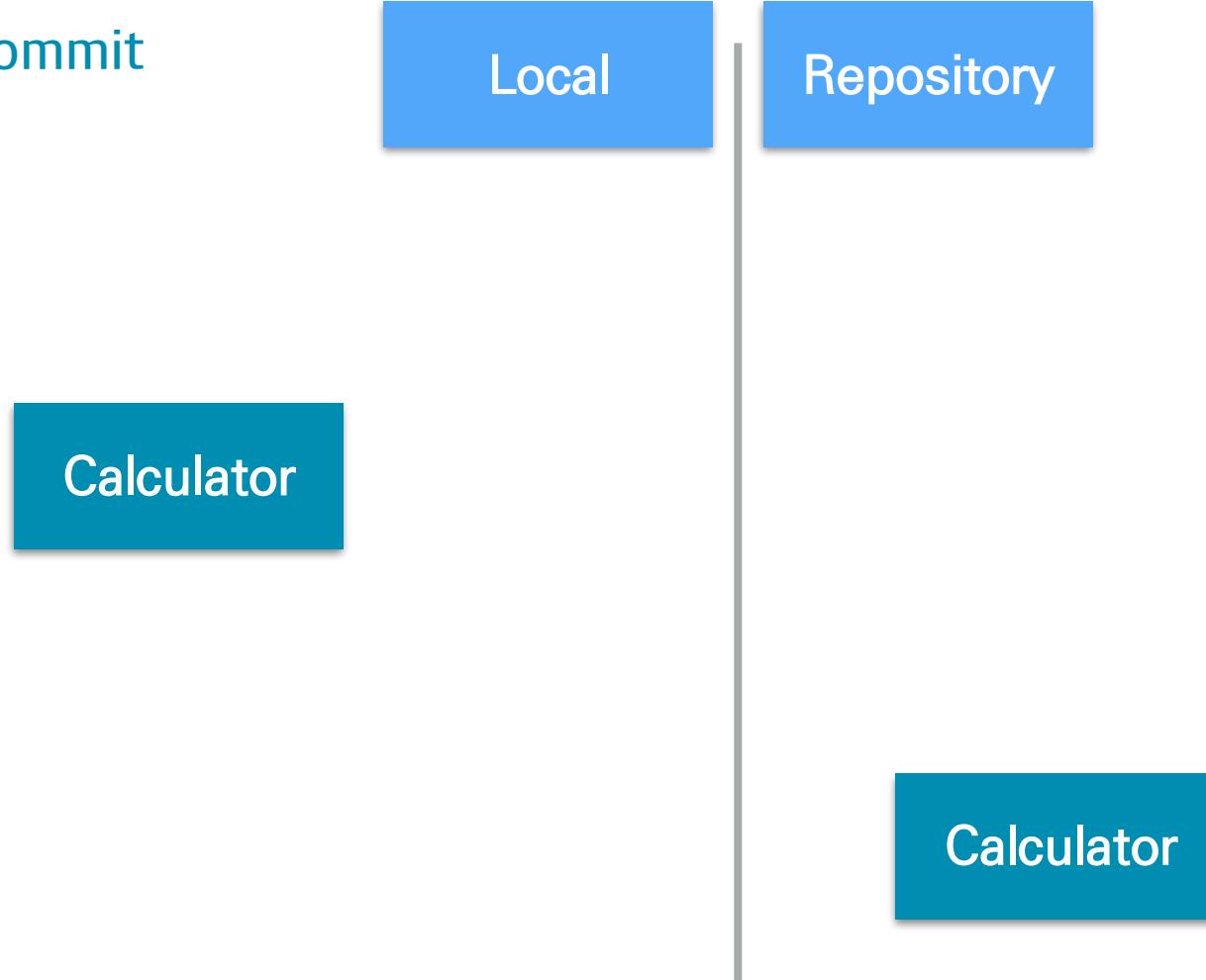
- › 계산기 클래스 생성



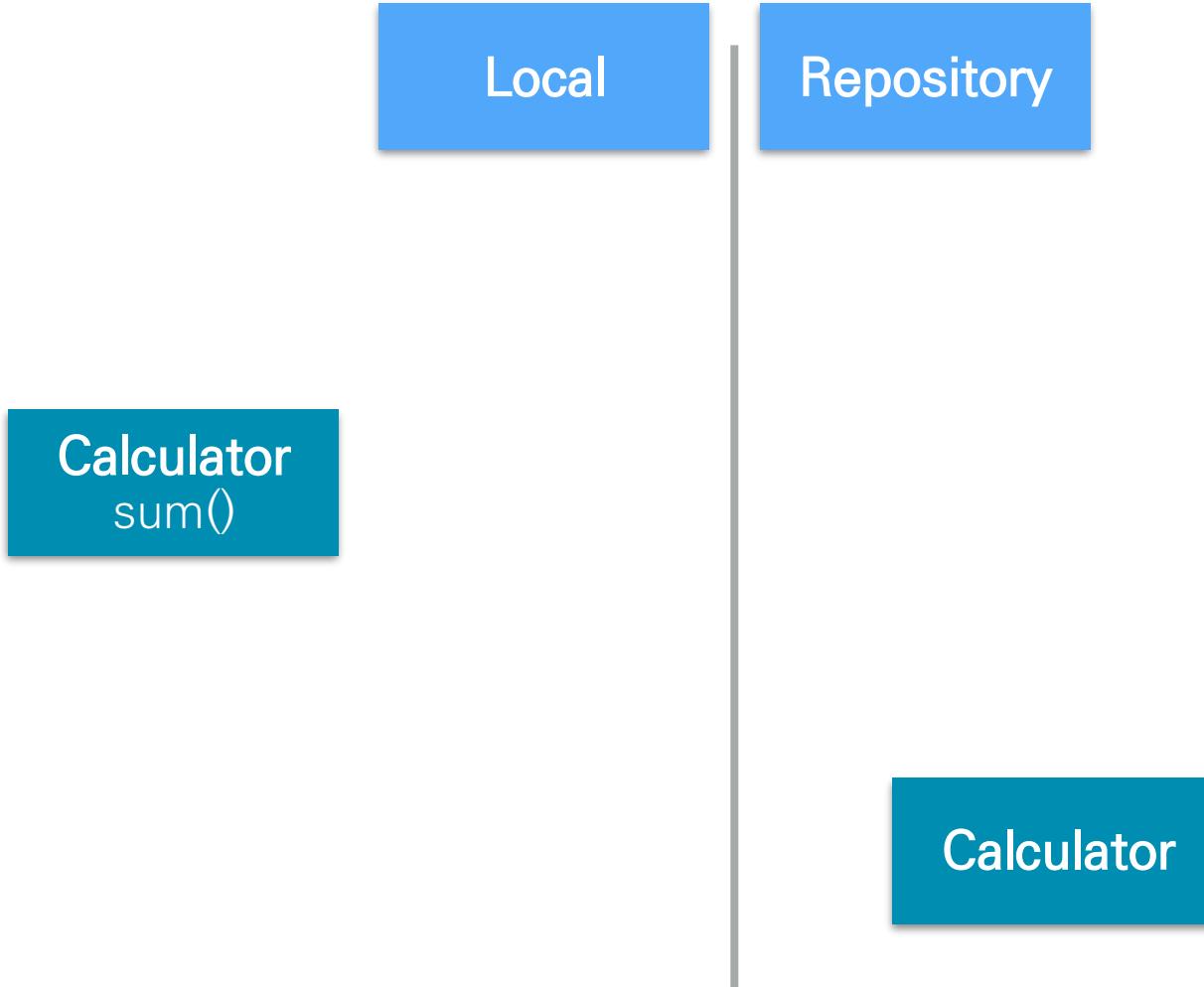
# Git – 그림설명

## ▶ Repository 에 저장

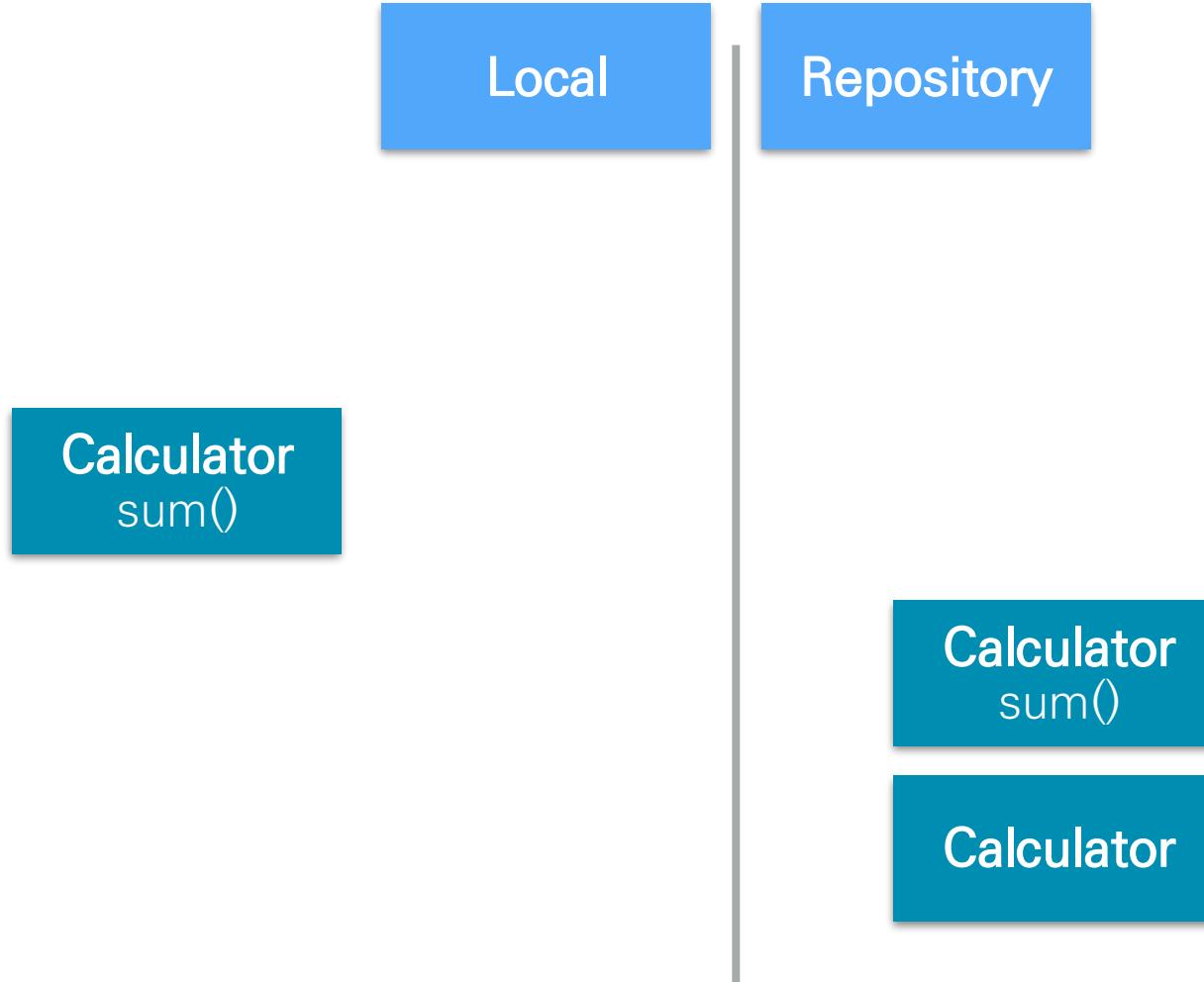
- Commit



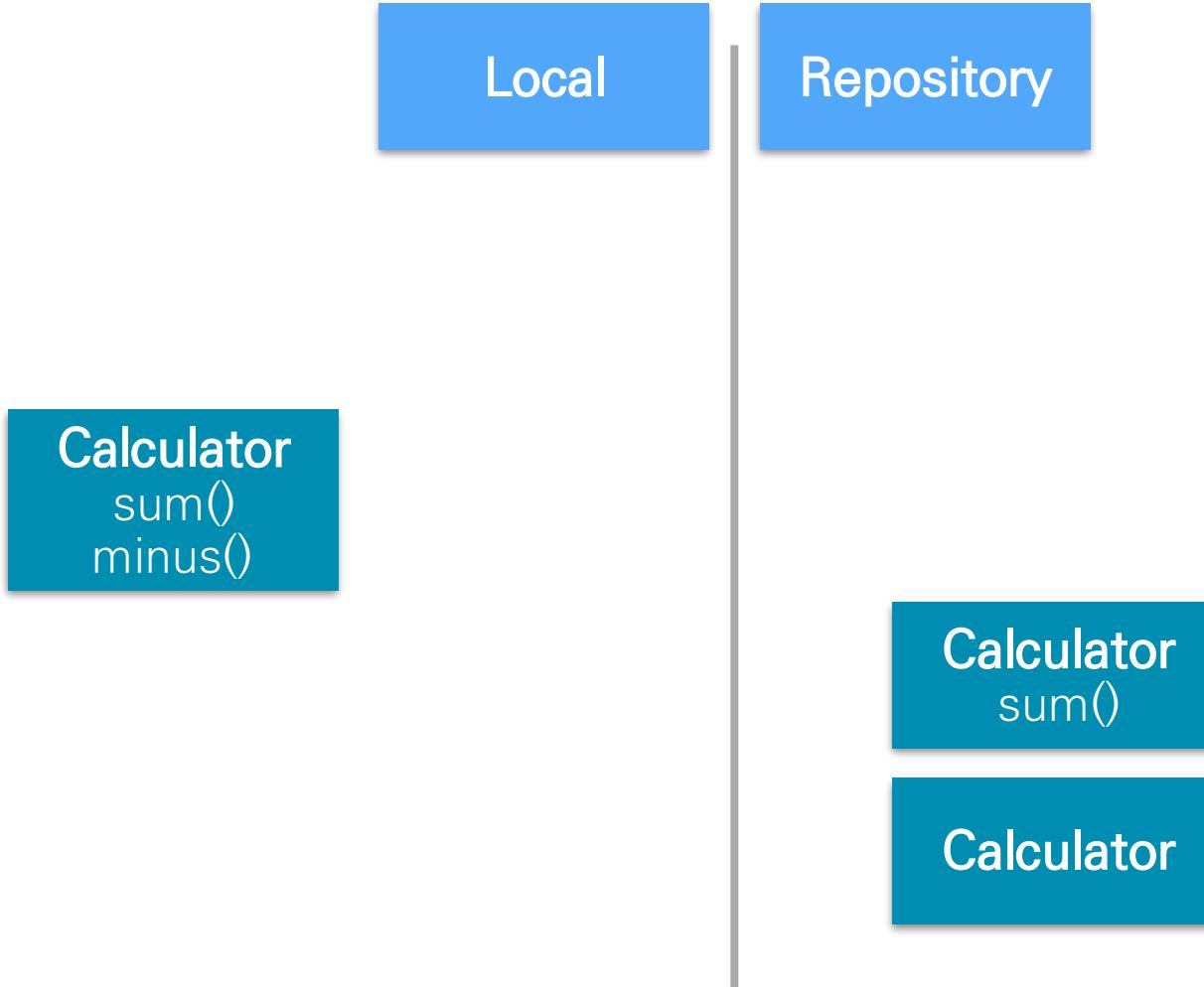
# Git - 그림 설명



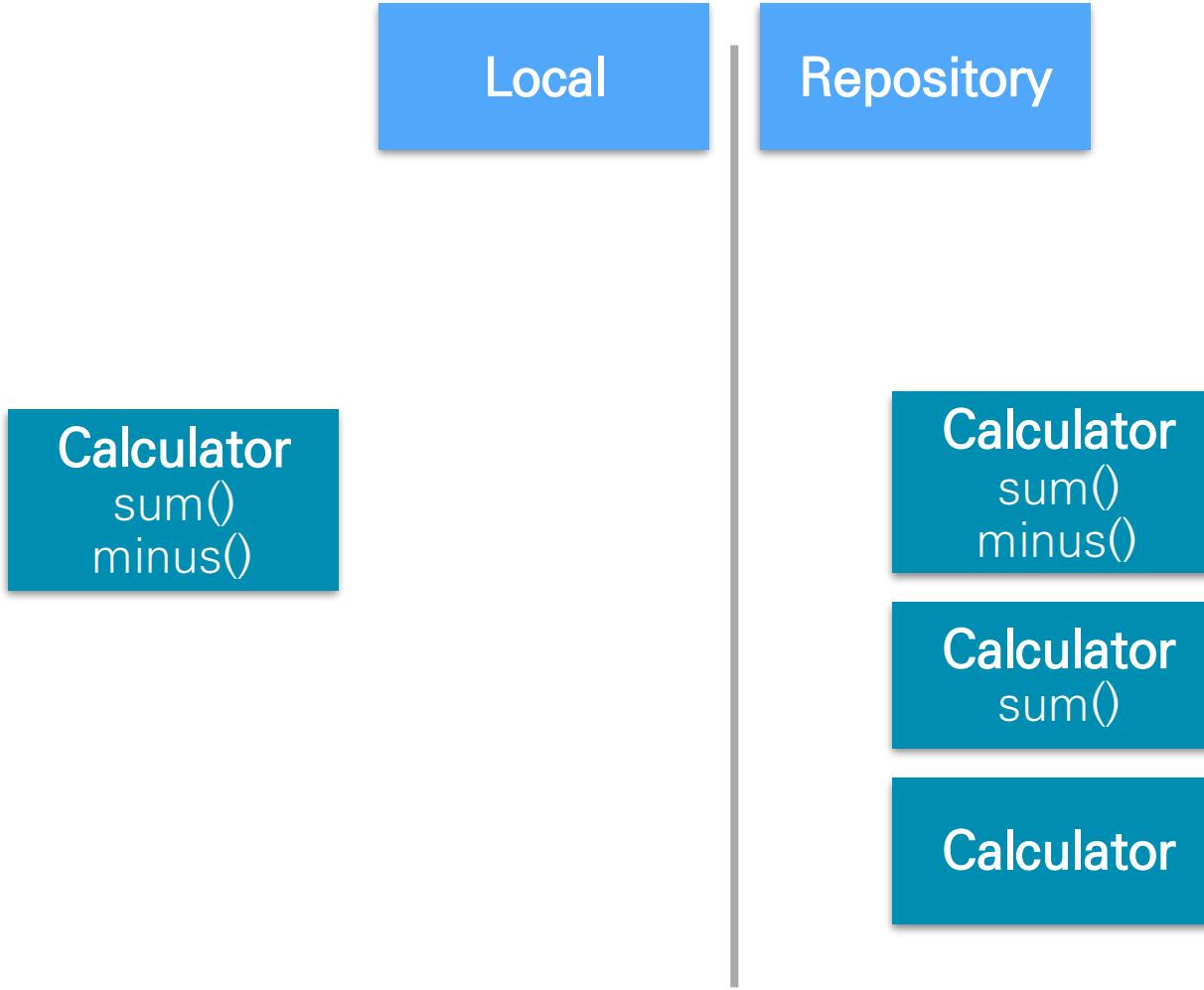
# Git - 그림 설명



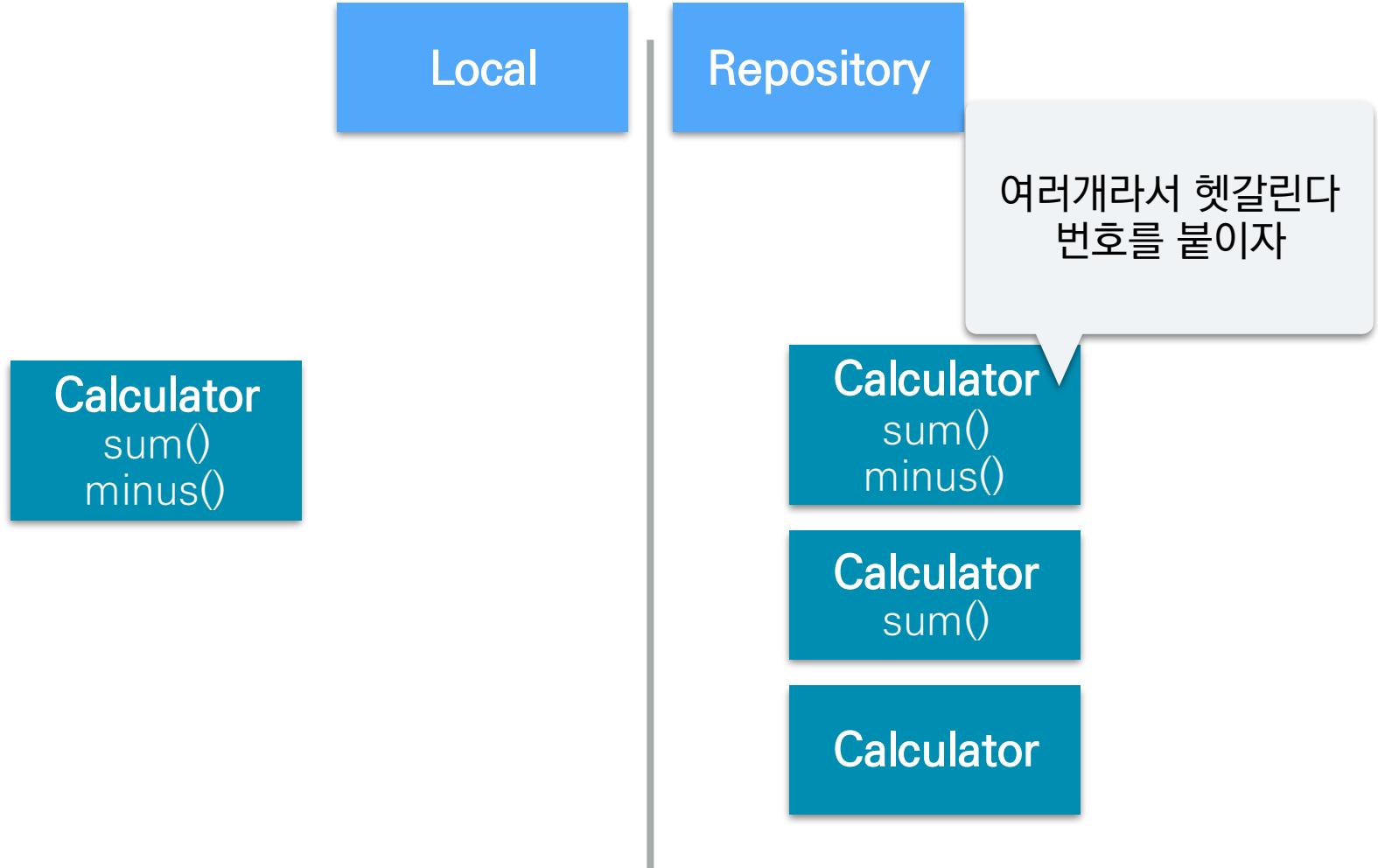
# Git – 그림 설명



# Git – 그림 설명



# Git – 그림 설명



# Git - 그림 설명

Local

Repo

..  
..

S U B V E R S I O N

Calculator

3

sum()  
minus()

Calculator

3

sum()  
minus()

Calculator

2

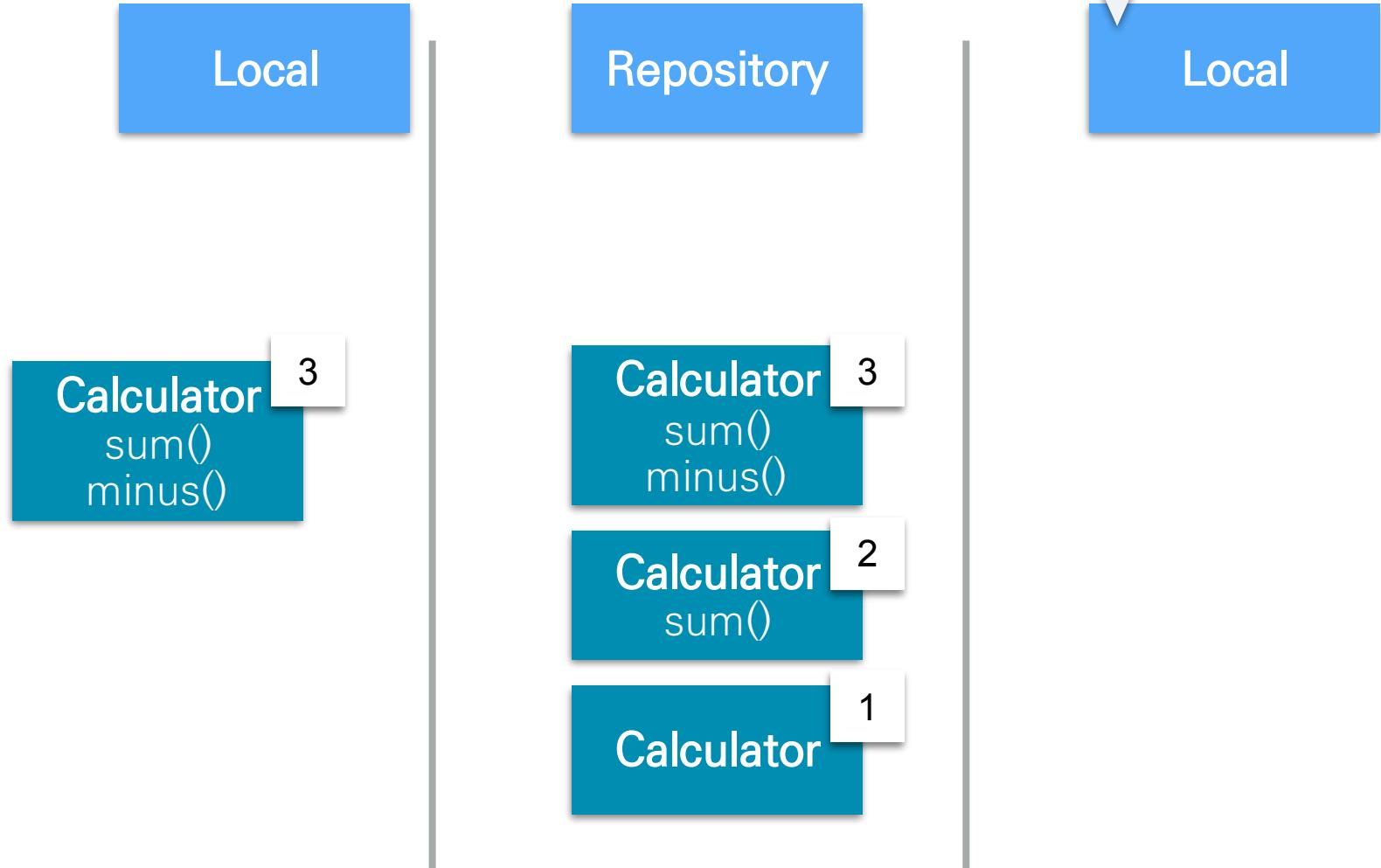
sum()

Calculator

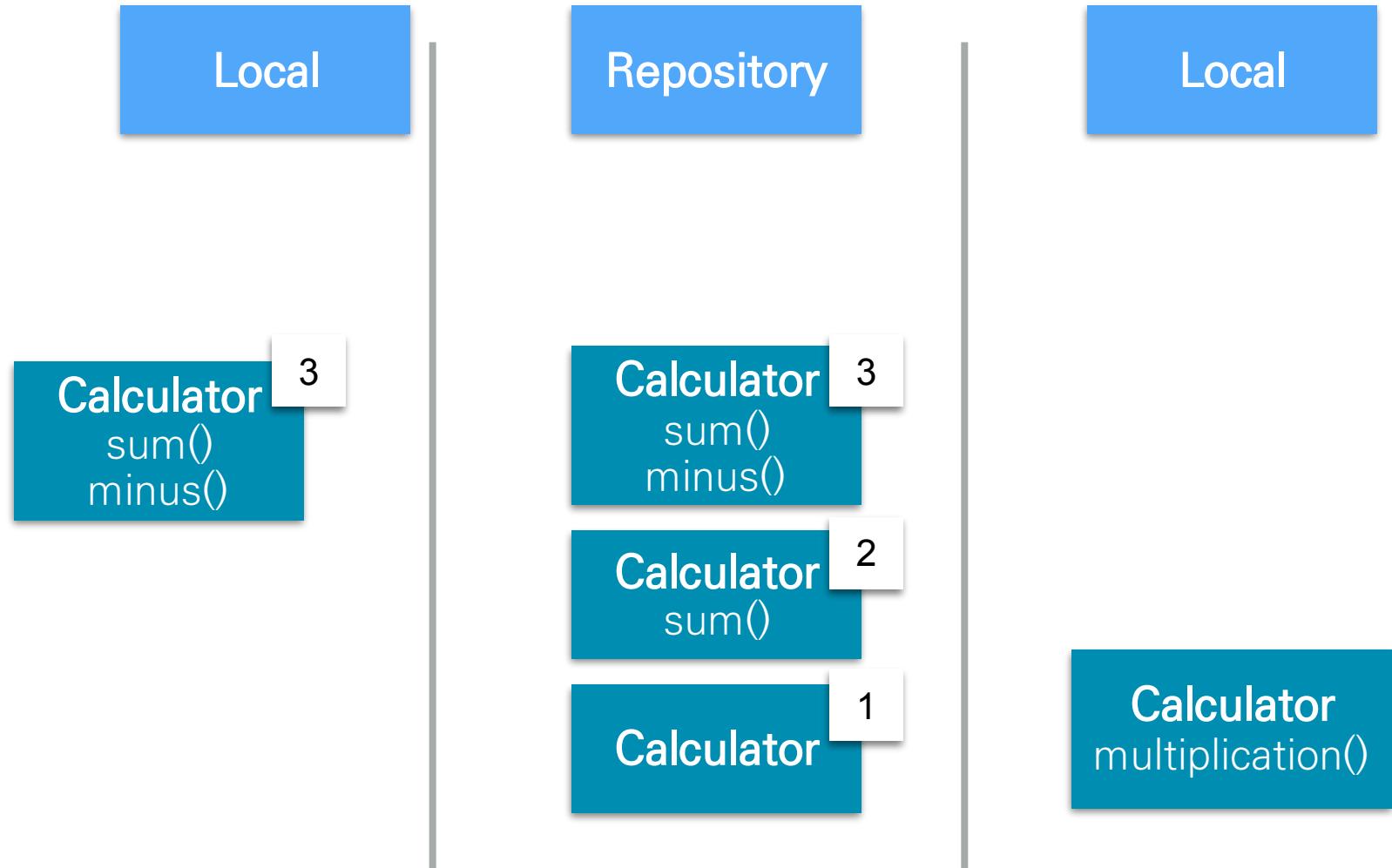
1

# Git – 그림 설명

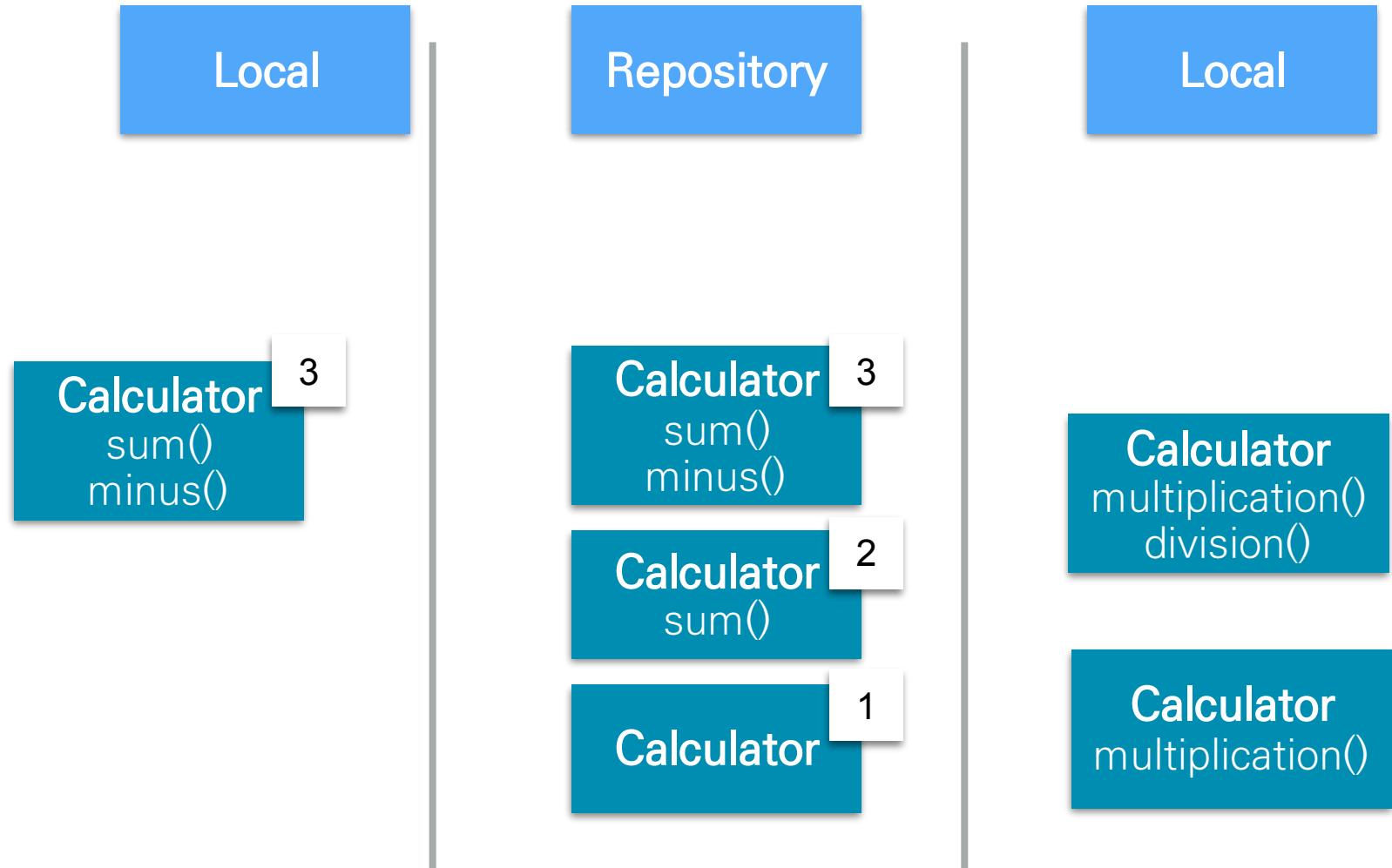
곱셈을 개발하는  
동료



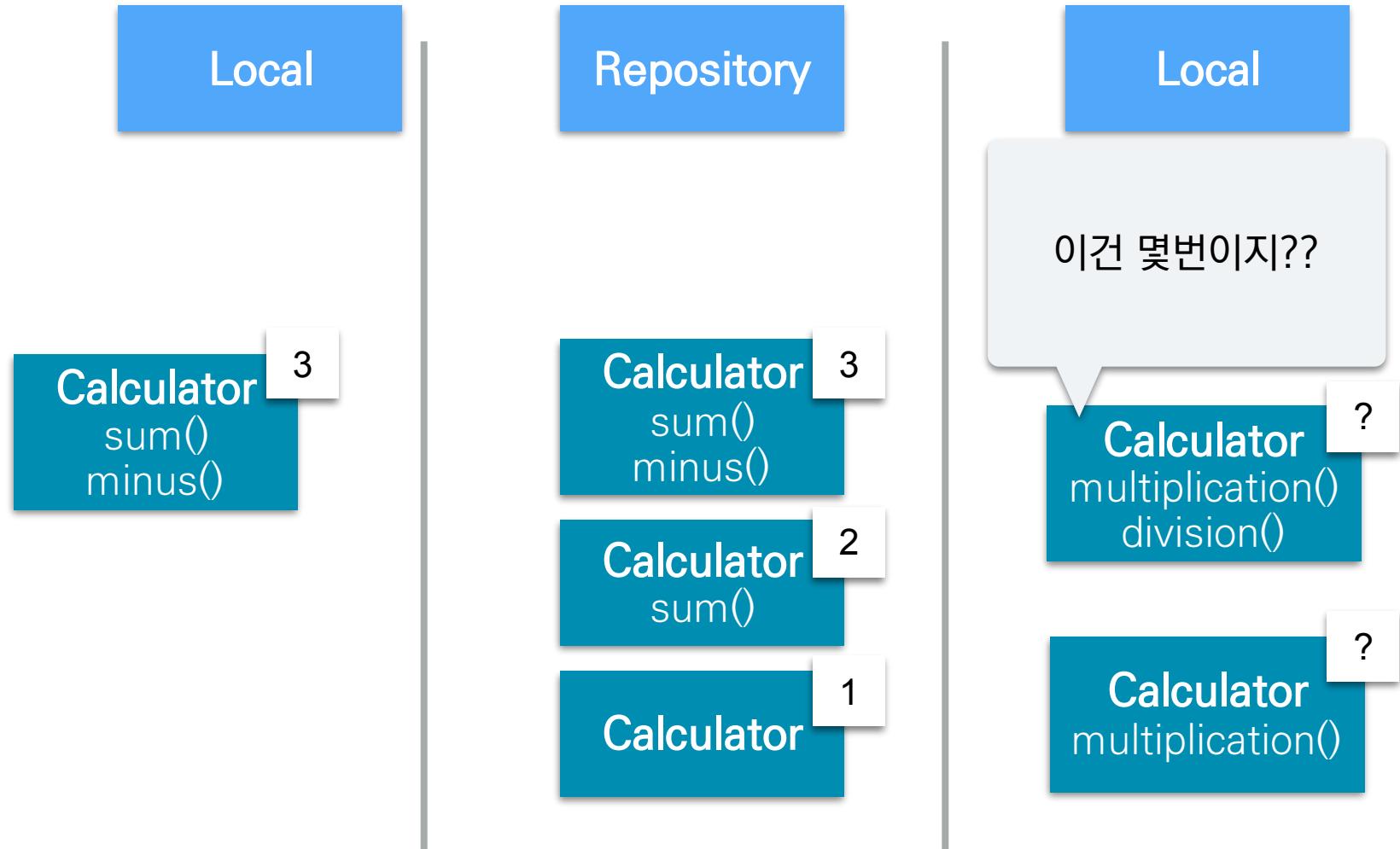
# Git – 그림 설명



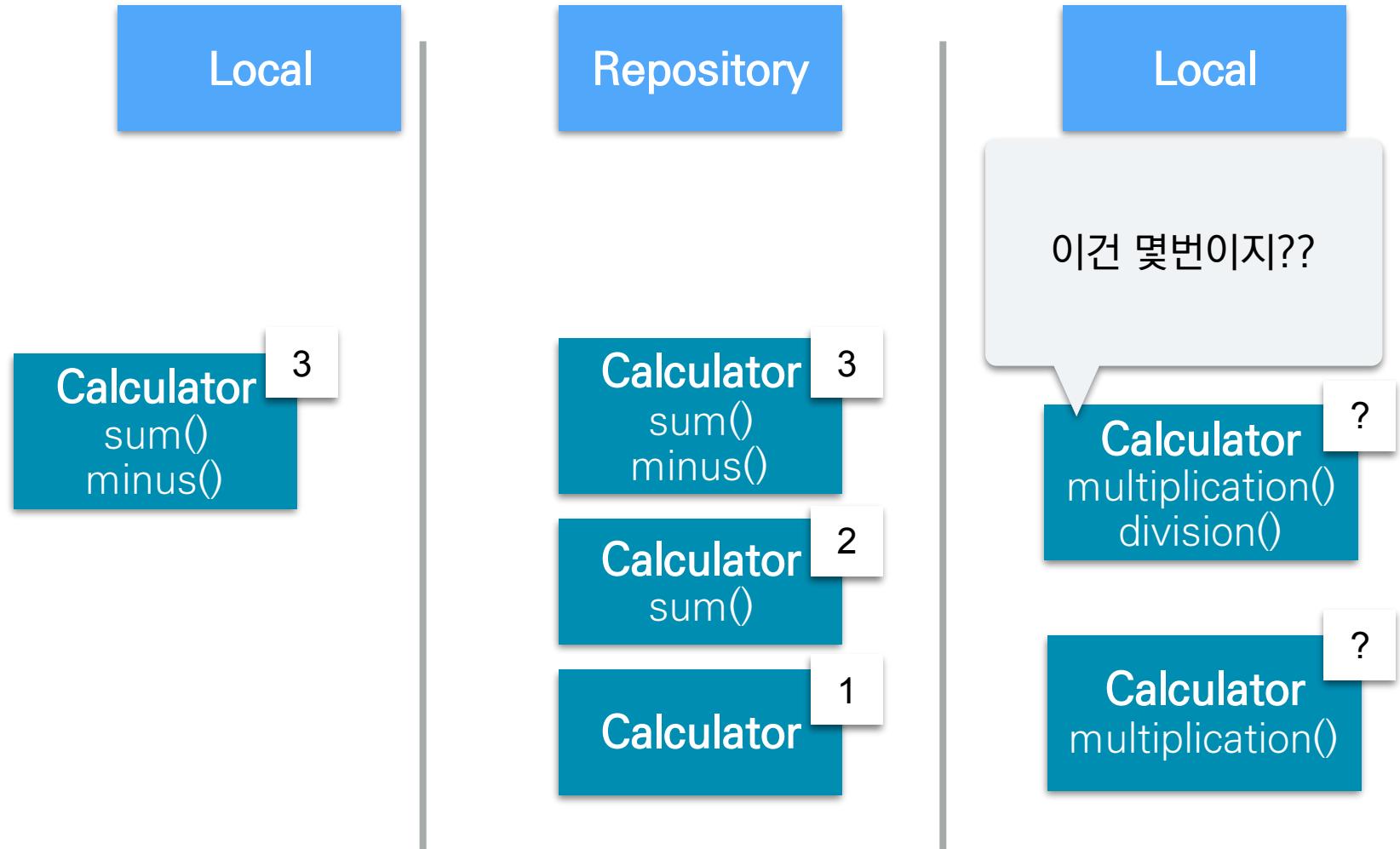
# Git – 그림 설명



# Git – 그림 설명



# Git – 그림 설명



# Git - 그림설명

변호로 암해!!  
Hash 씨자

# Git – Hash

## ▶ 모양

- “834a6d13967395216936419680ebd65ef8550566”
- “doca7b8294401dcfdaafa406c5ac039d8fb98c32”

# Git – Hash

- › 서버에 커밋 순번을 불일 수 없다
  - 나의 로컬 Repository에 커밋
  - 다른 개발자도 로컬에 커밋
- › 그냥 아무값이나 붙이자
  - Hash
  - SHA-1 값을 생성하여 붙인다 (랜덤값)
- › 아무값이나 붙이면 순서는?
  - 부모를 기록하자

# Git – Hash

## ▶ 랜덤값이면 중복은?

- 가능
- 확률 :  $1/2^{80}$ 
  - 1.2경 : 12,000,000,000,000,000
  - 지구에 존재하는 모래알 수의 1,200배
- 확률 예시
  - 지구상의 6.5억명의 개발자가 있다고 가정하고
  - 매초, 리눅스 커널 히스토리 전체(100만개)를 Push 할 때
  - 5년이 지나면, 중복 확률이 50% 정도 된다 (from Pro Git 서적)

# Git – Hash

## ▶ 모양

- “834a6d13967395216936419680ebd65ef8550566”
- “doca7b8294401dcfdaafa406c5ac039d8fb98c32”

The screenshot shows a GitHub repository dashboard for 'iot-labs / dashboard'. The 'Code' tab is selected. A commit from 'jongkwang' on Jun 29 is highlighted with a red box around the commit hash 'commit 9872e70f9fe5ff3dc387521fd74705c4201100c8'. The commit message is 'Add a license policy.' and it has 1 parent commit '6fab19b'. Below the commit, it says 'Showing 1 changed file with 35 additions and 0 deletions.' The file 'README.md' has 35 additions. The commit message in the diff view includes a link to documentation: '@@ -63,3 +63,38 @@ Please, visit [here](https://github.com/iot-labs/communication) for documentatio'.

# Git – Hash

## ▶ 모양

- “**834a6d13967395216936419680ebd65ef8550566**”
- “**doca7b8294401dcfdaafa406c5ac039d8fb98c32**”

The screenshot shows a GitHub repository dashboard for the 'iot-labs / dashboard' repository. The 'Code' tab is selected. At the top right, there are buttons for 'Unwatch' (3), 'Star' (6), 'Fork' (3), and 'Issues' (1). Below the navigation bar, it shows 'Branch: master' and the file 'dashboard / README.md'. A commit history is displayed, with the most recent commit by 'jongkwang' on June 29 highlighted with a red box. The commit message is 'Add a license policy.' and it has 1 contributor. At the bottom, it shows '101 lines (60 sloc) | 4.03 KB' and buttons for 'Raw', 'Blame', 'History', and file operations.

iot-labs / dashboard

Code Issues 1 Pull requests 0 Projects 0 Wiki Settings Insights

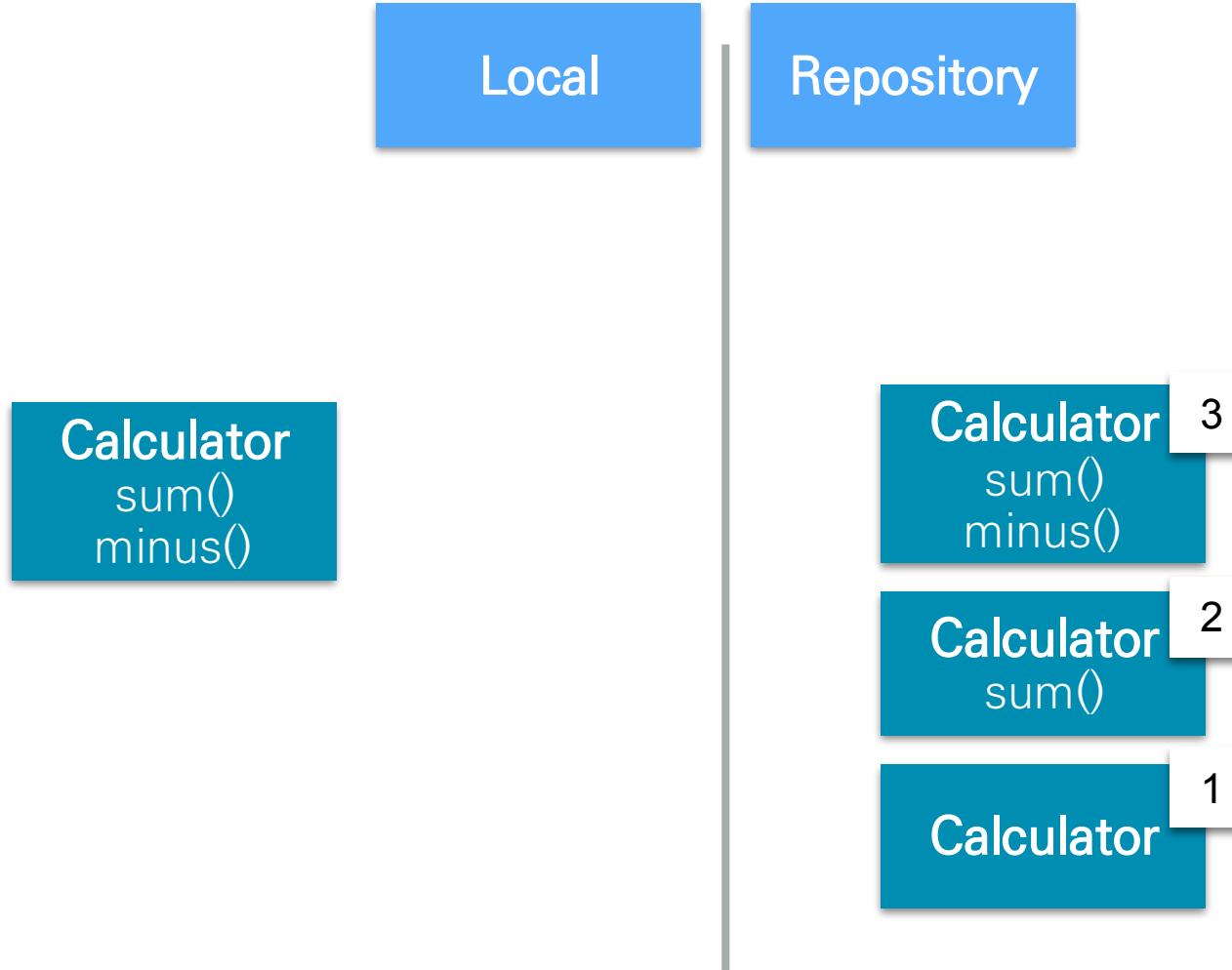
Branch: master dashboard / README.md Find file Copy path

jongkwang Add a license policy. 9872e70 on Jun 29

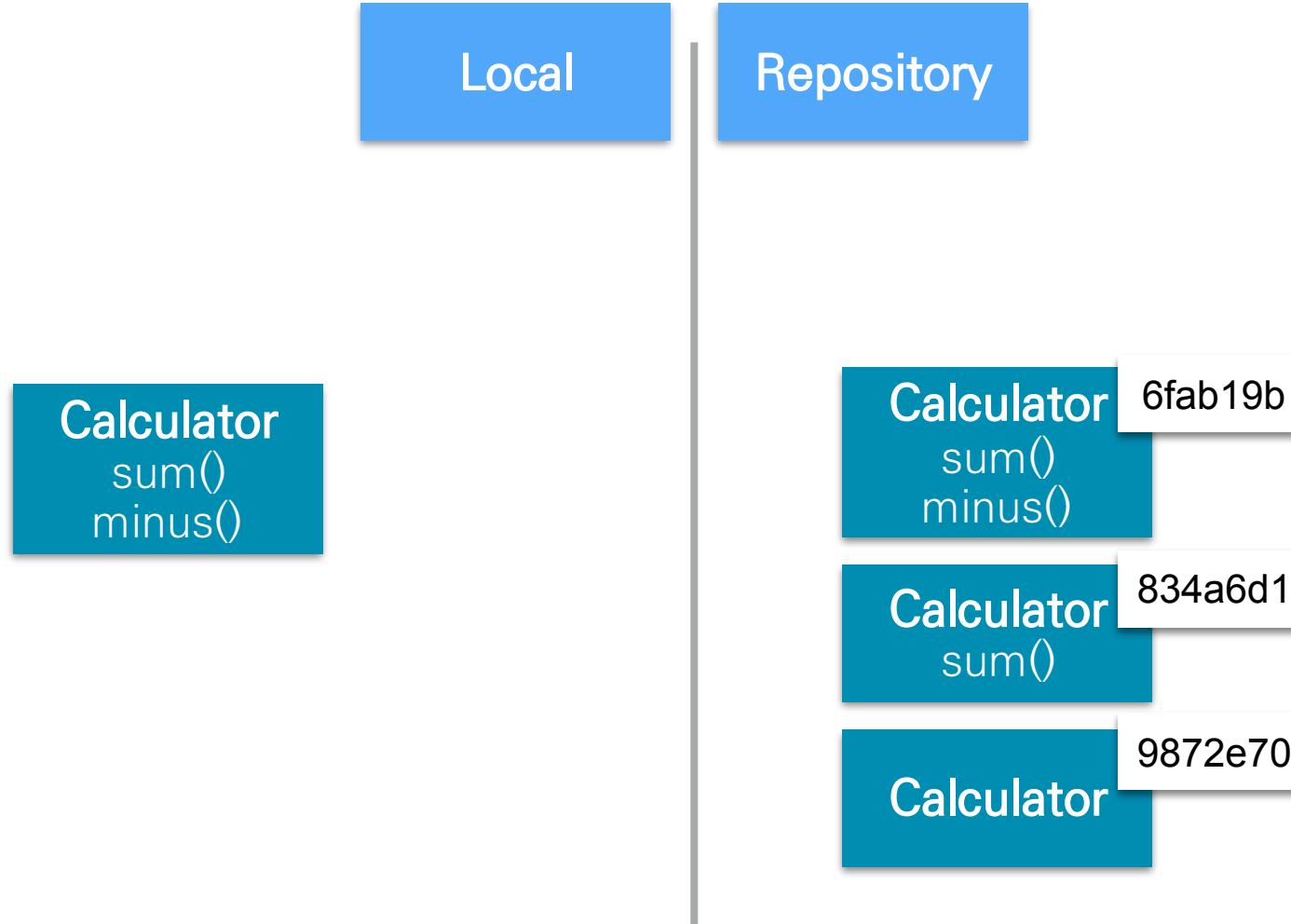
1 contributor

101 lines (60 sloc) | 4.03 KB Raw Blame History

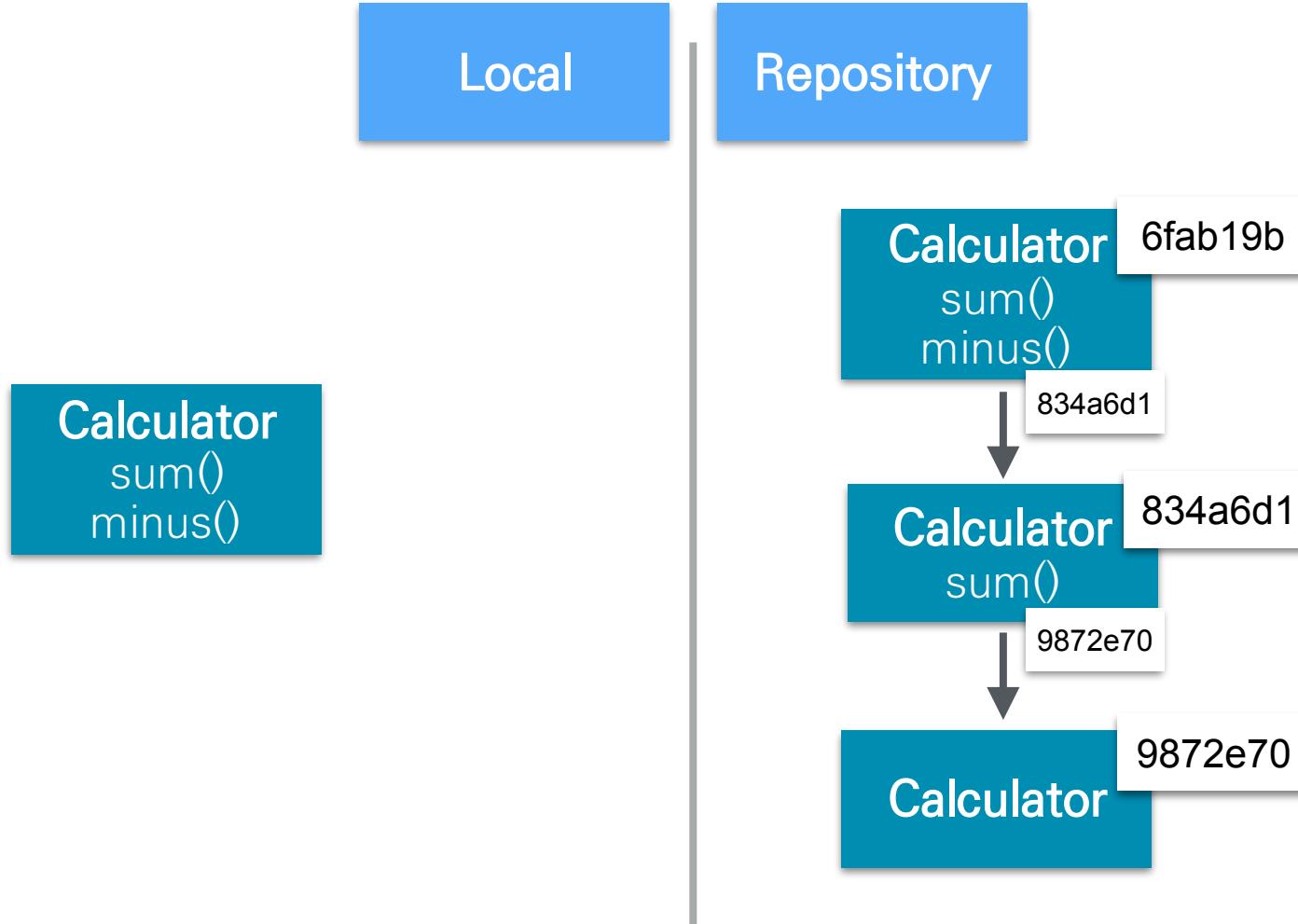
# Git – 그림 설명



# Git – 그림 설명



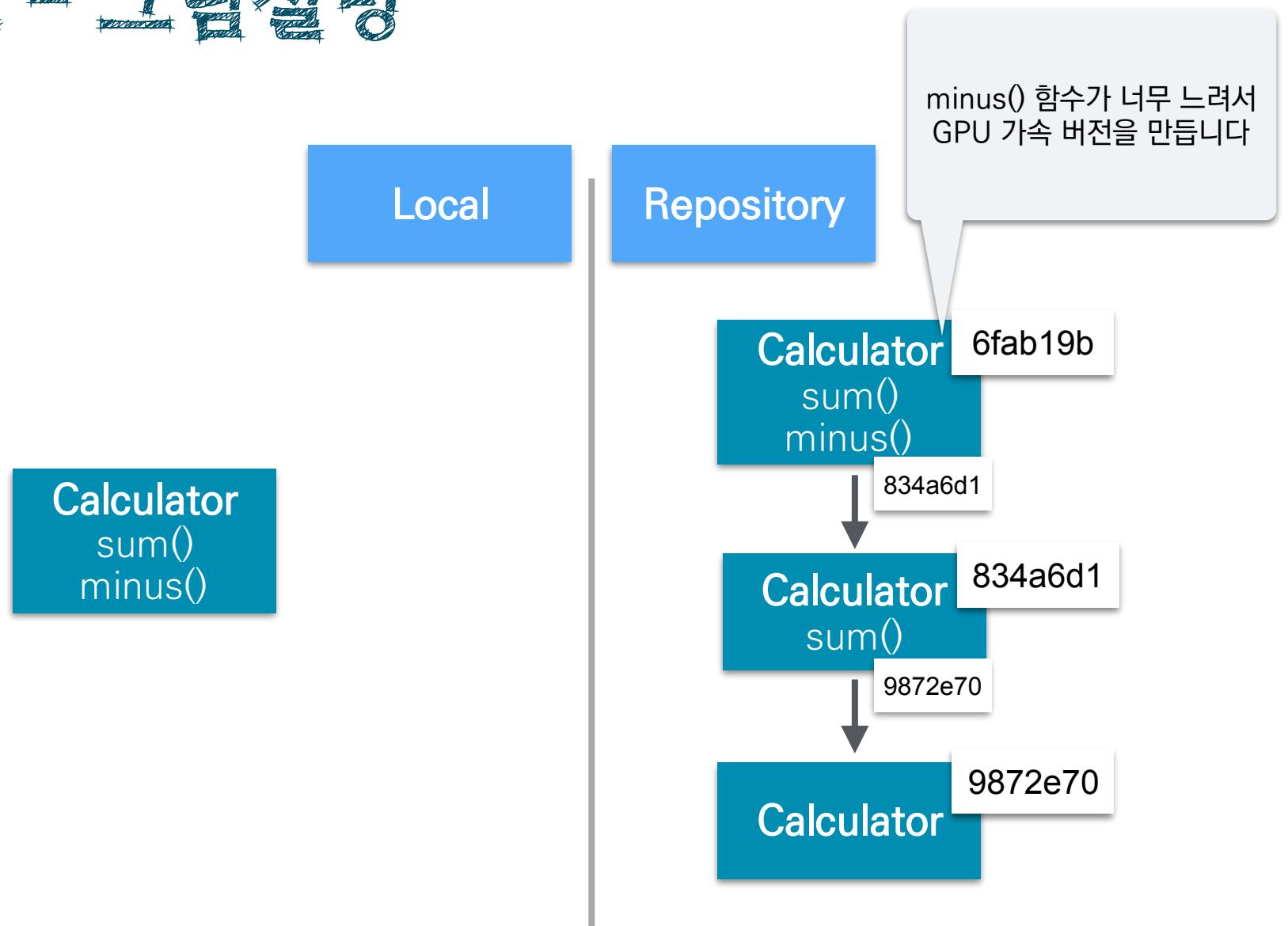
# Git – 그림 설명



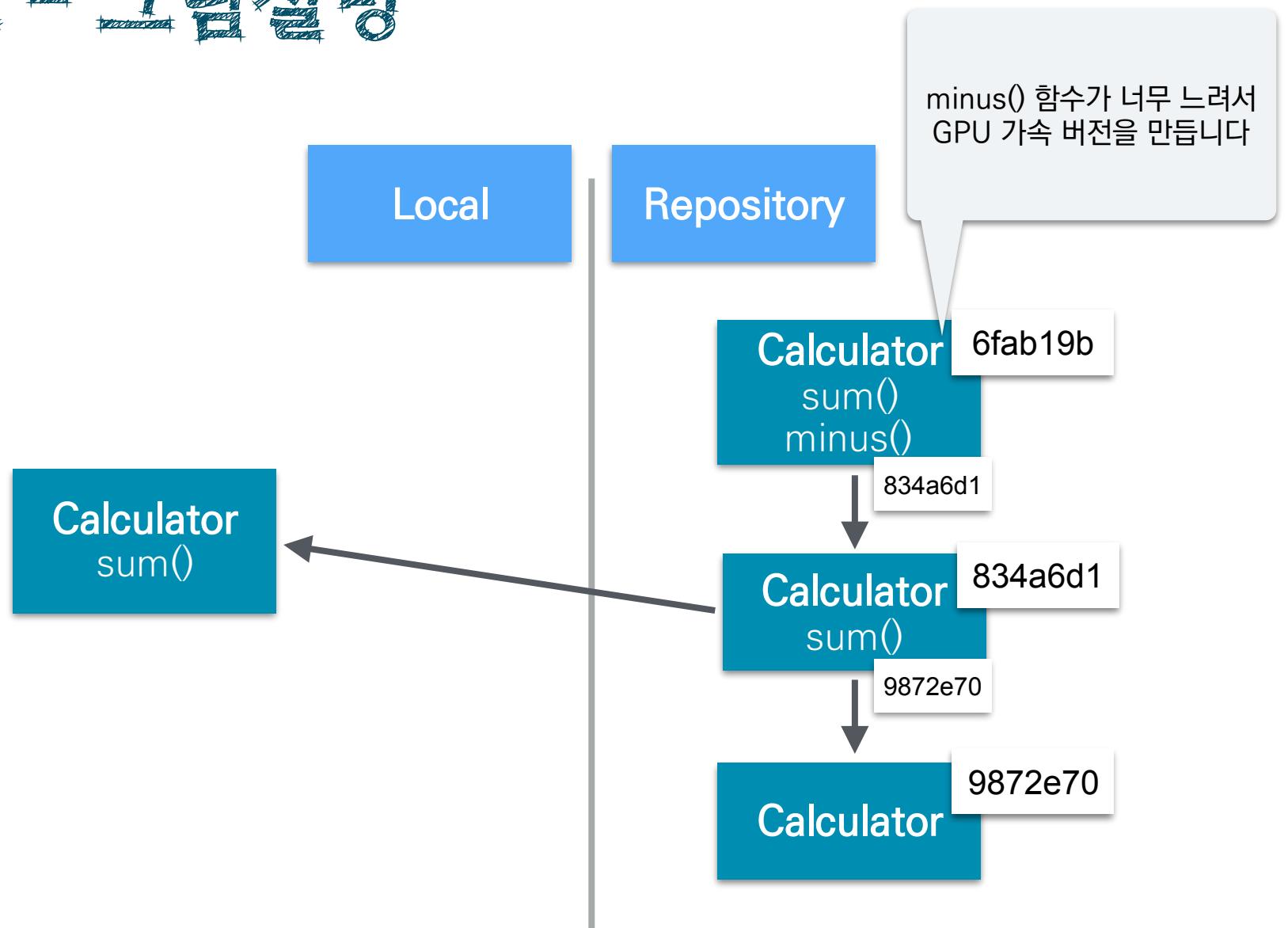
Git - 그림설명

minus0 재개발  
with GPU

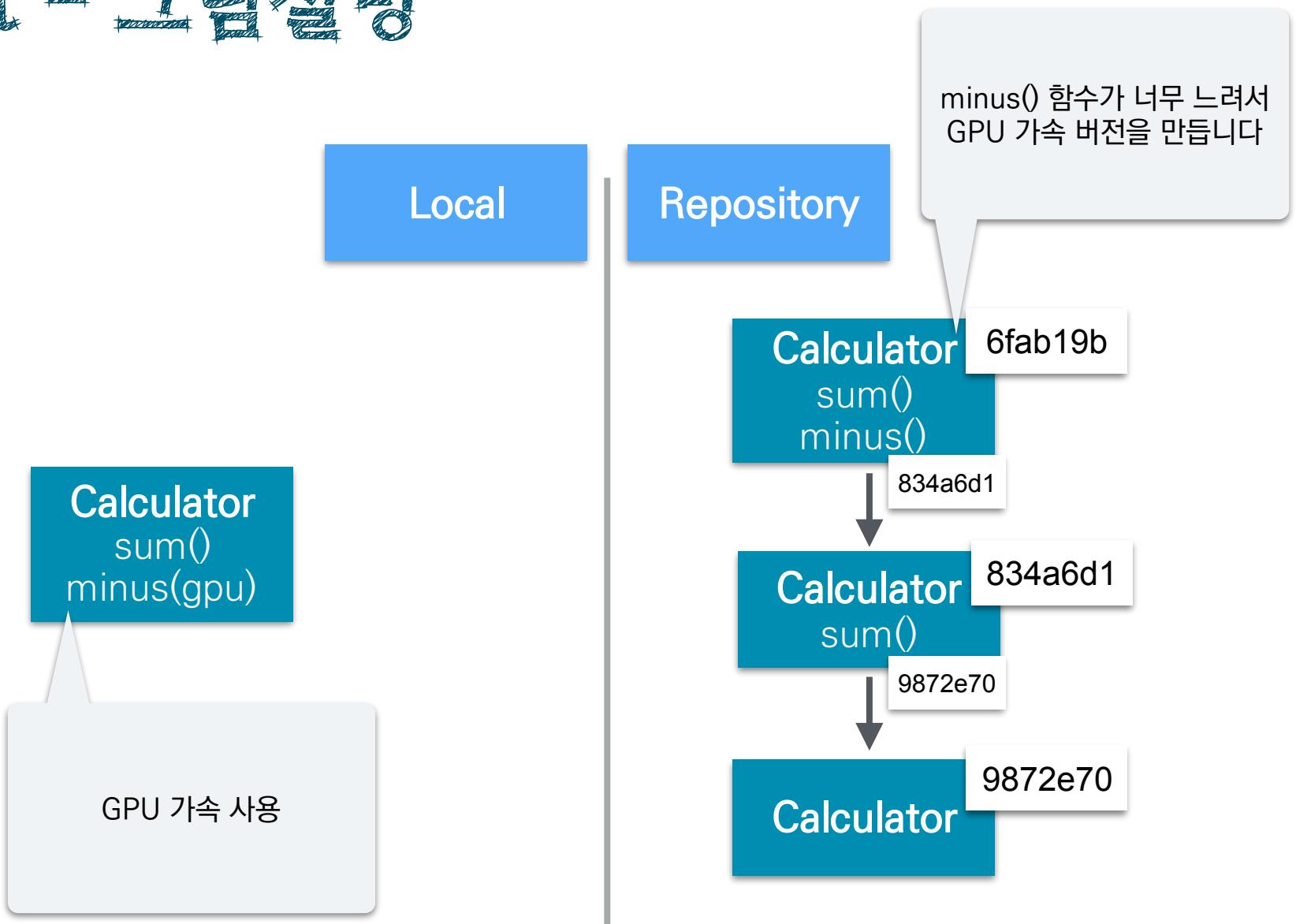
# Git – 그림 설명



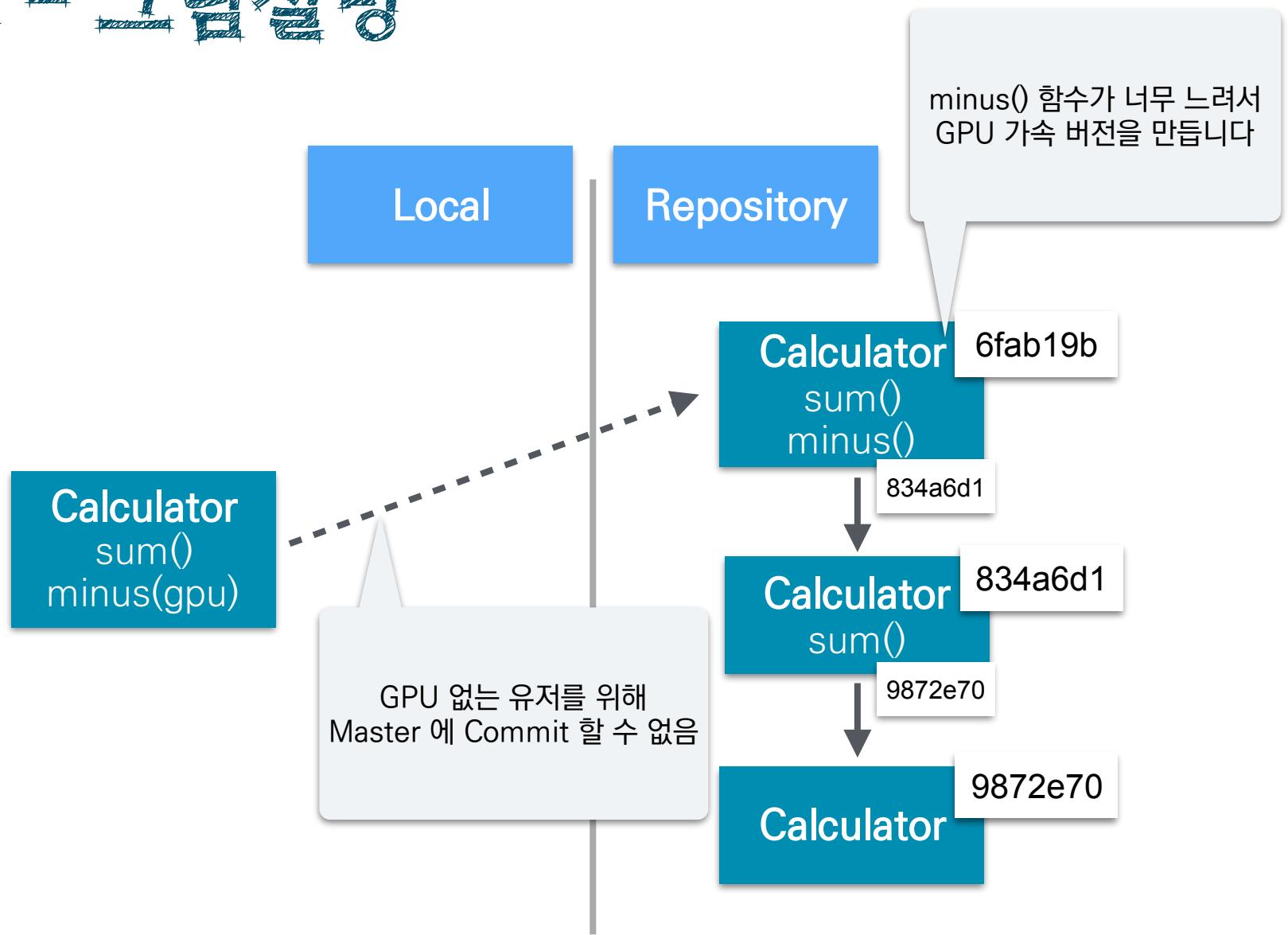
# Git – 그림 설명



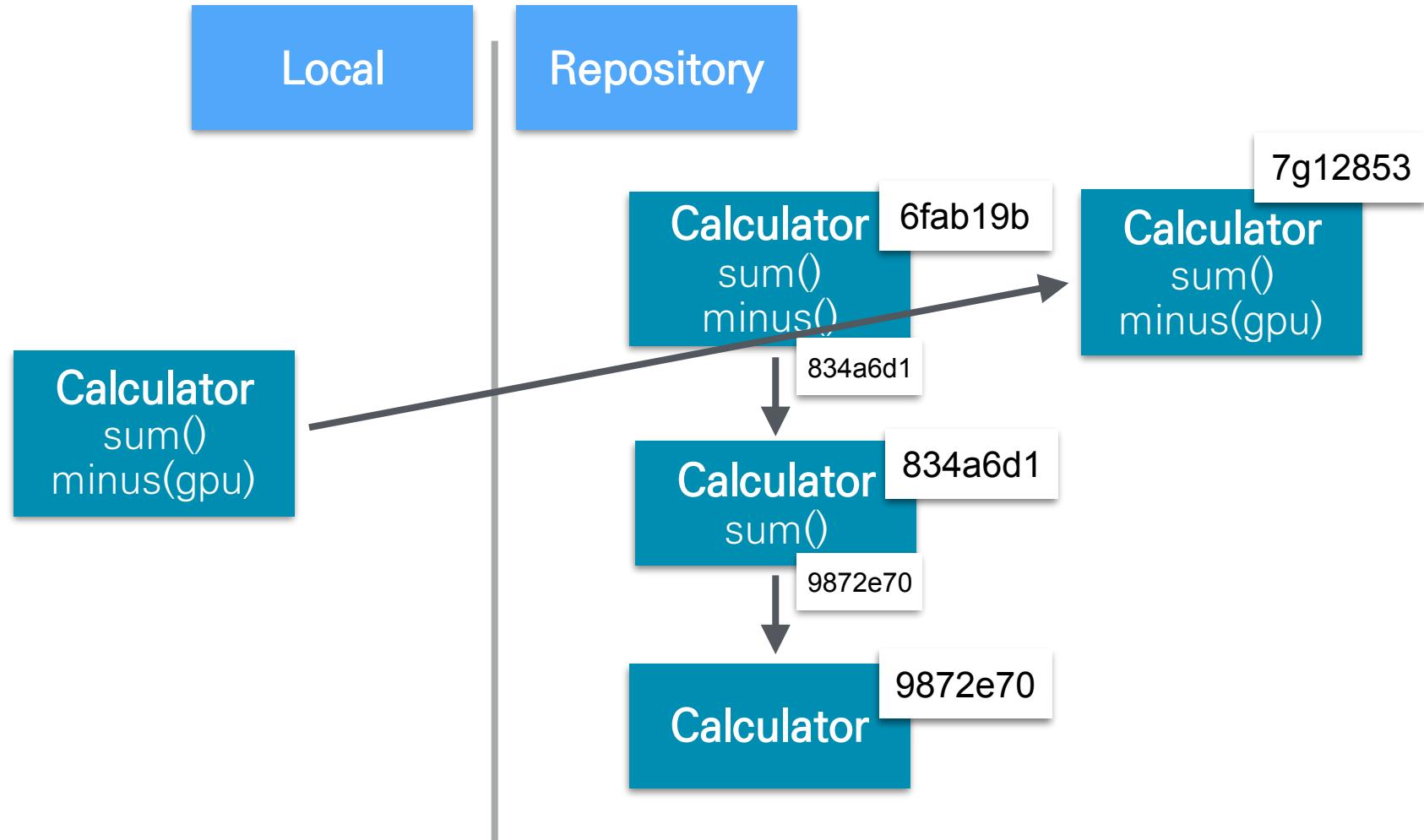
# Git – 그림 설명



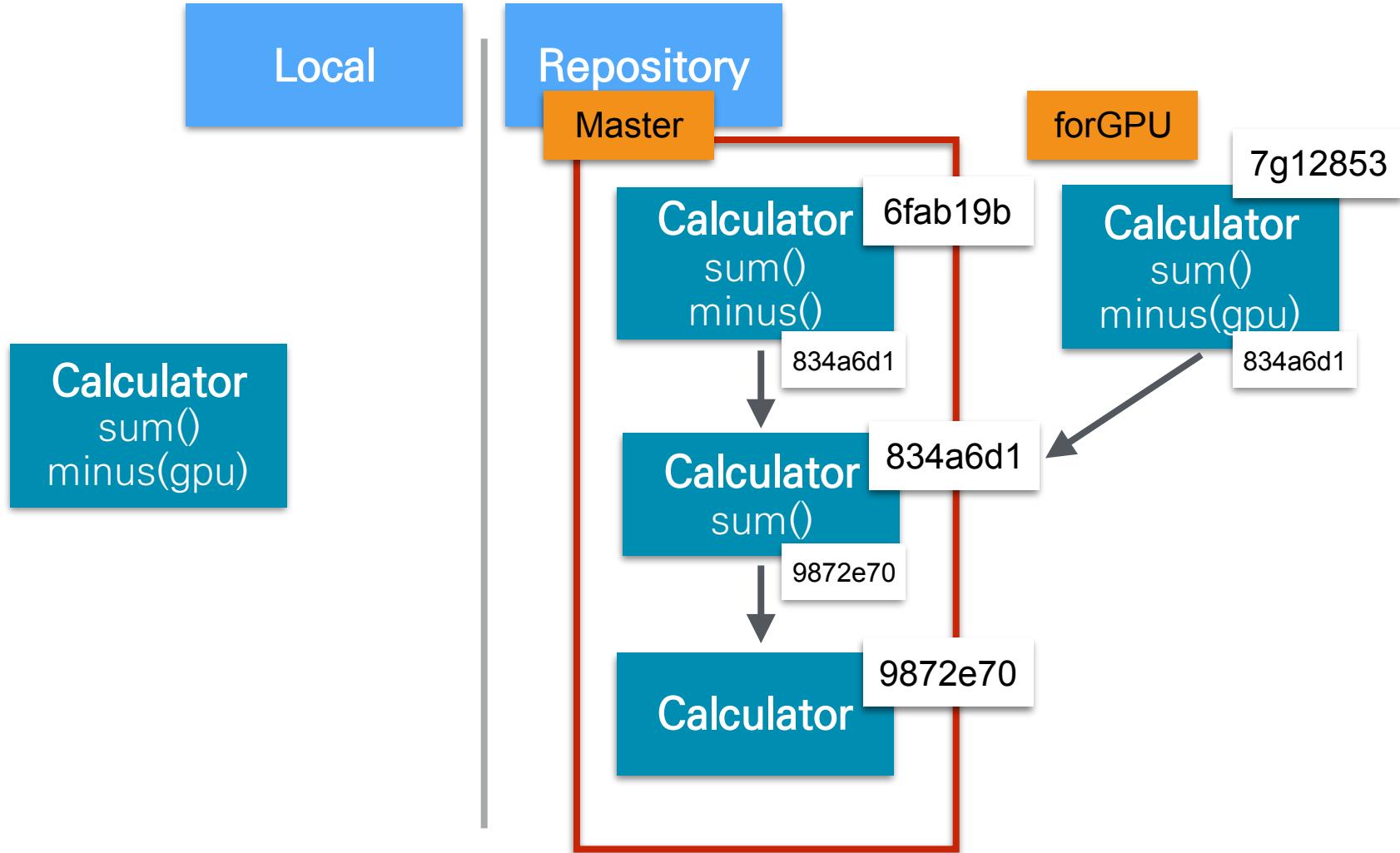
# Git – 그림 설명



# Git – 그림 설명



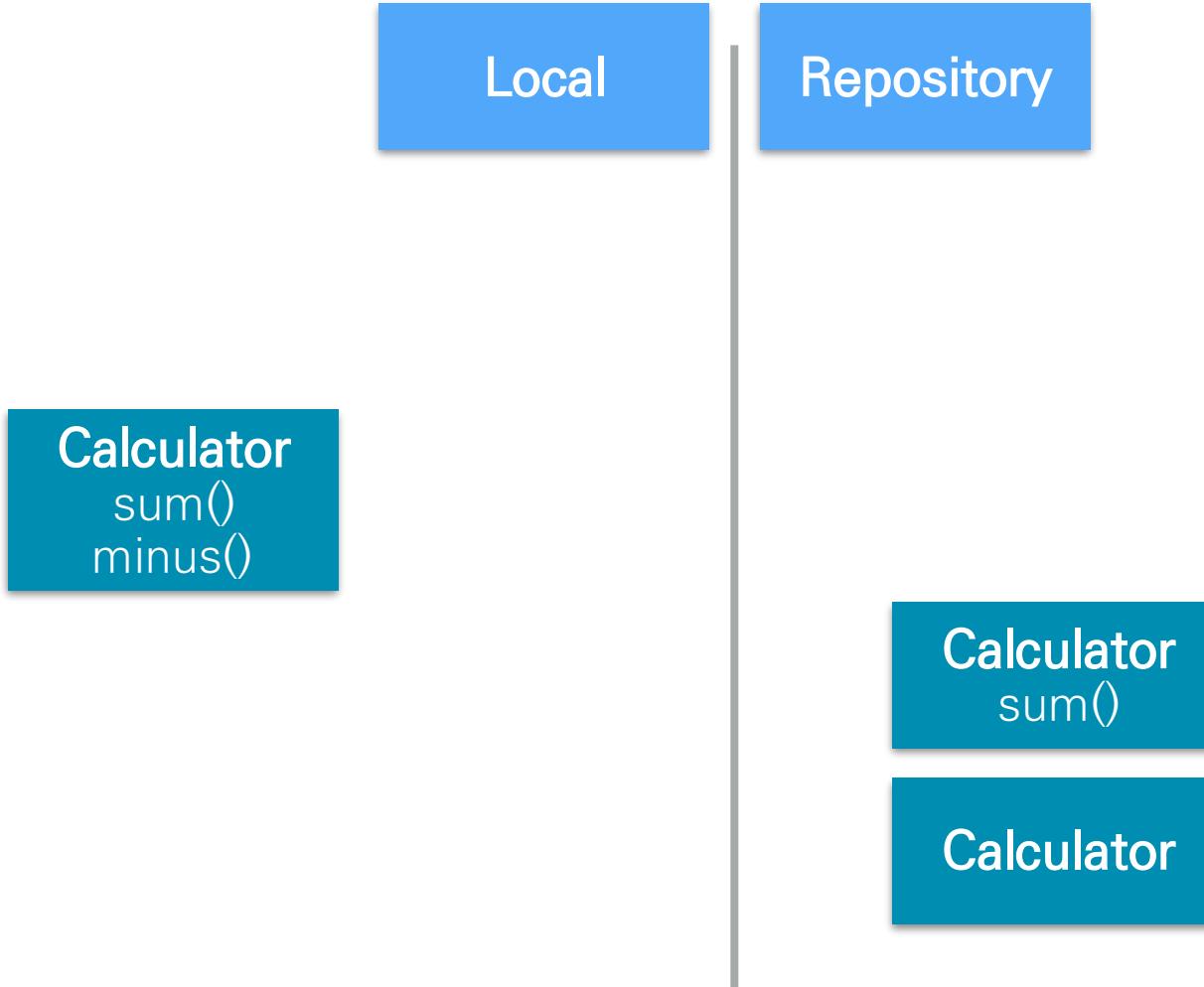
# Git – 그림 설명



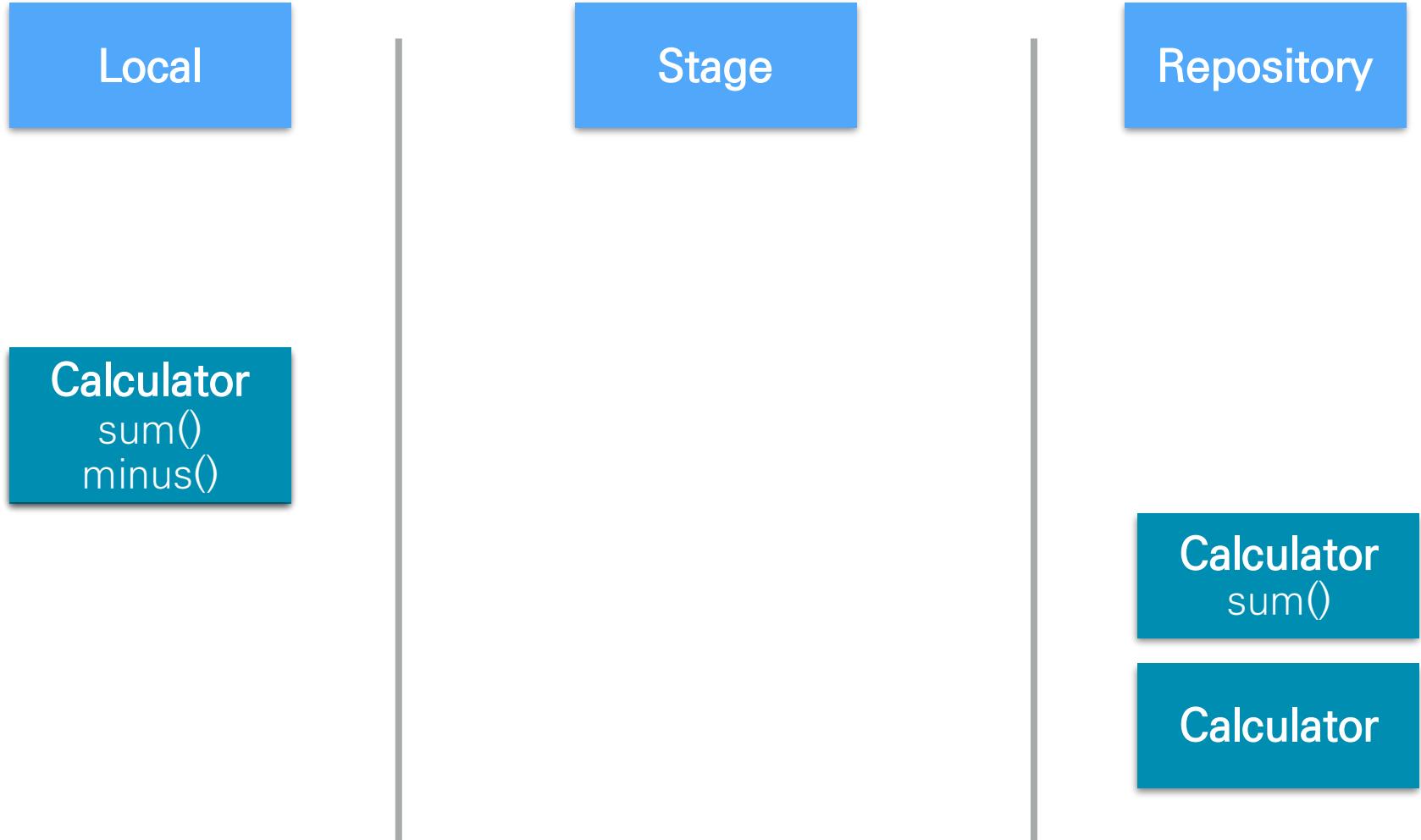
Git - 그림설명

Stage → 무대

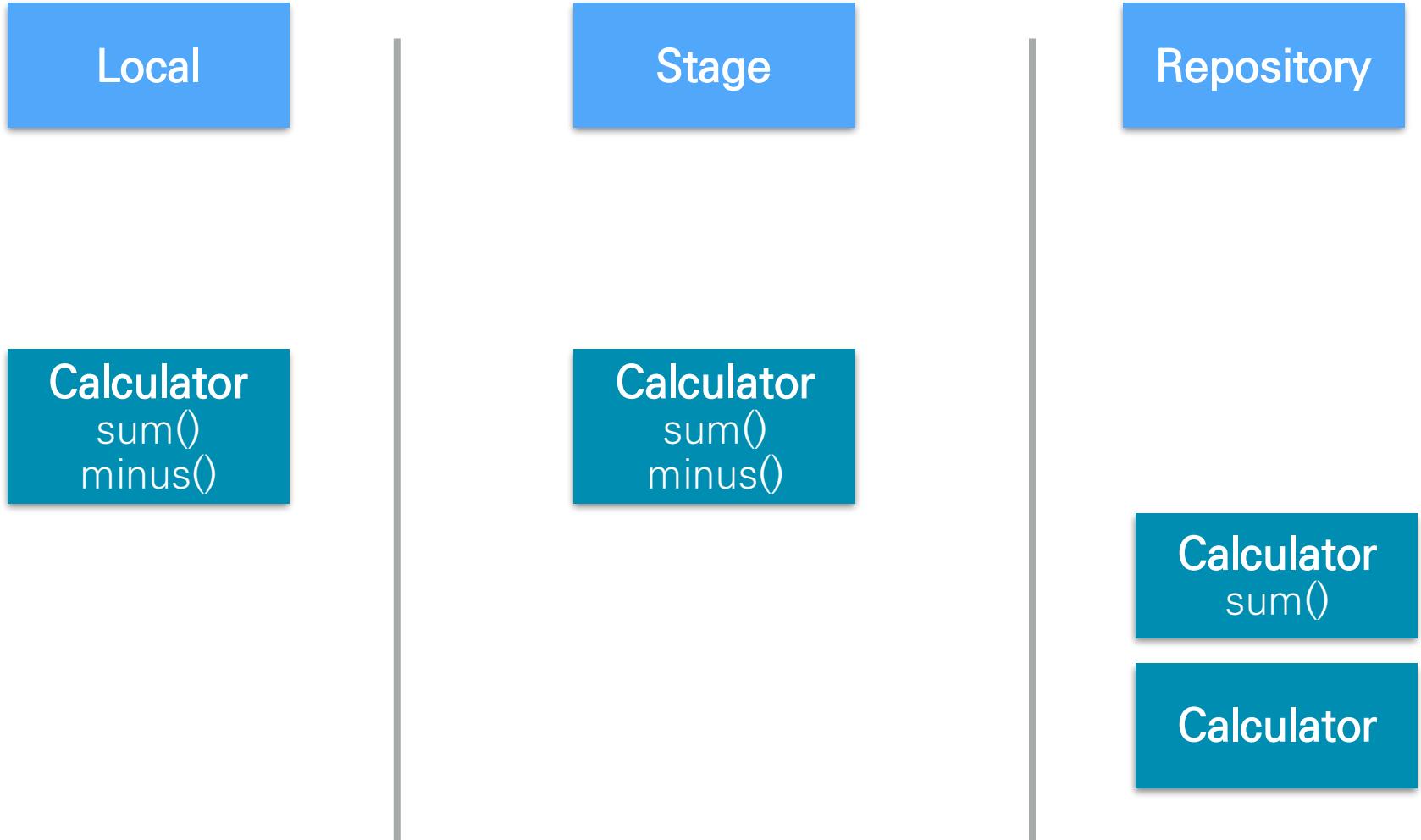
# Git – 그림 설명



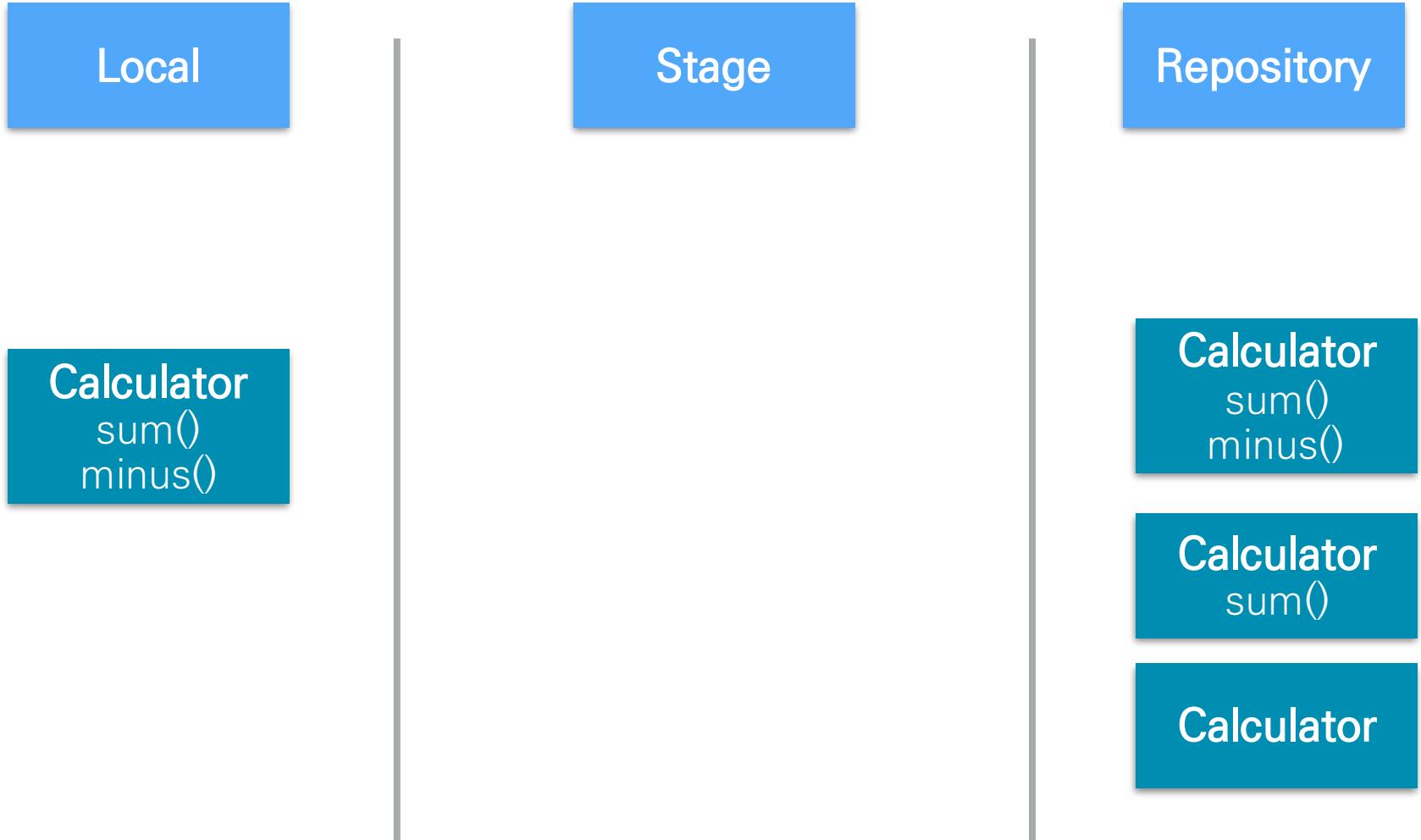
# Git - 그림 설명



# Git - 그림 설명



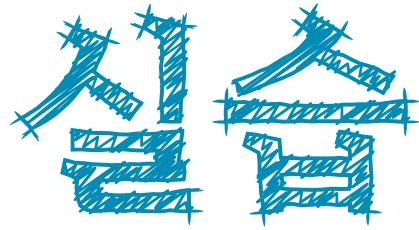
# Git - 그림 설명



# Git – 그림 설명



# Git - 그림설명



Git

Join Github

# Join Github

## ▶ Github

- <https://github.com>

The screenshot shows the GitHub 'Join GitHub' page. At the top, there's a navigation bar with links for Features, Business, Explore, Marketplace, Pricing, a search bar, and a 'Sign In' button. The main heading is 'Join GitHub' with the subtext 'The best way to design, build, and ship software.' Below this, there are three steps: 'Step 1: Create personal account', 'Step 2: Choose your plan', and 'Step 3: Tailor your experience'. The first step is highlighted. The 'Create your personal account' form includes fields for 'Username' (placeholder 'GitHub username'), 'Email Address' (placeholder 'Email address'), and 'Password' (placeholder 'Strong password'). Below the form, a note says 'Use at least one lowercase letter, one numeral, and seven characters.' A terms and conditions section at the bottom states: 'By clicking on "Create an account" below, you are agreeing to the [Terms of Service](#) and the [Privacy Policy](#)'. A large green 'Create an account' button is at the bottom.

# Join Github

- › Unlimited public repositories for free
  - 공개SW 무제한 무료

The screenshot shows the GitHub welcome screen after account creation. At the top, there's a navigation bar with the GitHub logo, a search bar, and links for Pull requests, Issues, Marketplace, and Gist. To the right are user profile icons and a '+' button.

The main heading is "Welcome to GitHub". Below it, a message says "You've taken your first step into a larger world, @jongkwangmail.".

Three steps are outlined:

- Completed**: Set up a personal account (green checkmark)
- Step 2:** Choose your plan (blue icon)
- Step 3:** Tailor your experience (gear icon)

The "Choose your personal plan" section contains two options:

- Unlimited public repositories for free.
- Unlimited private repositories for \$7/month.

A note below says "Don't worry, you can cancel or upgrade at any time."

Under "Both plans include:" is a list of features:

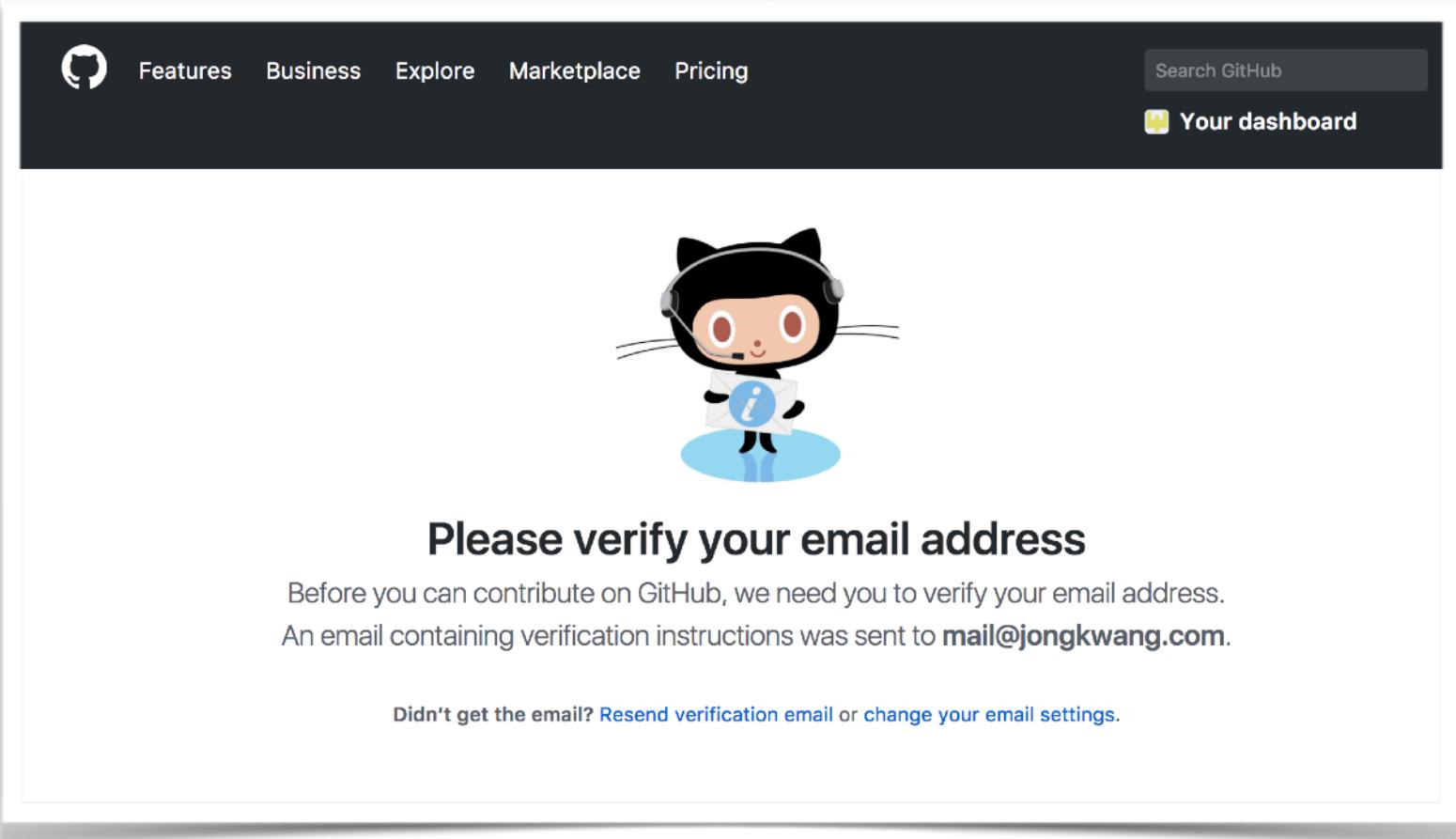
- ✓ Collaborative code review
- ✓ Issue tracking
- ✓ Open source community
- ✓ Unlimited public repositories
- ✓ Join any organization

At the bottom left is a "Continue" button.

# Join Github

## ▶ Verify email address

- 확인 전까지 Repository 생성이 안된다



The screenshot shows the GitHub homepage with a dark header. The GitHub logo is on the left, followed by links for Features, Business, Explore, Marketplace, and Pricing. A search bar is on the right, along with a link to 'Your dashboard'. The main content area features the GitHub cat wearing a headset and holding a blue envelope with an 'i' on it. Below the cat, the text reads 'Please verify your email address'. It explains that verification is required to contribute and provides the recipient's email as 'mail@jongkwang.com'. At the bottom, there's a link for resend instructions.

Search GitHub

Your dashboard

Please verify your email address

Before you can contribute on GitHub, we need you to verify your email address.  
An email containing verification instructions was sent to [mail@jongkwang.com](mailto:mail@jongkwang.com).

Didn't get the email? [Resend verification email](#) or [change your email settings](#).

Git

Profile

# Profile

## ▶ Profile

- Repository
- Contributions
- Activity

The screenshot shows a GitHub profile page for the user 'jongkwang'. At the top, there's a header with a search bar, navigation links for 'Pull requests', 'Issues', 'Marketplace', and 'Gist', and account stats: 11 repositories, 22 stars, 28 followers, and 18 following. Below the header is a large profile picture of a smiling man with glasses. The main content area is divided into sections: 'Popular repositories' (listing 'Korea-Sencha-User-Group', 'MyGitHubTest', '12', 'tensorflow', 'cf-sample\_spring-music', and 'test-travis1'), 'Contributions in the last year' (a heatmap showing activity levels by month and day), and 'Contribution activity' (a timeline for August 2017). On the left side, there are sections for 'Follow' (button), 'Block or report user', and 'Organizations' (listing two organizations).

Popular repositories

- Korea-Sencha-User-Group  
Forked from Korea Sencha User Group  
★ 1
- MyGitHubTest  
HTML
- 12  
Forked from 1step6thswmaestro/12  
김종광  
Visual Basic
- tensorflow  
Forked from tensorflow/tensorflow  
Computation using data flow graphs for scalable machine learning  
C++ 1
- cf-sample\_spring-music  
Forked from blex/cf-sample\_spring-music  
A sample application for using database services on Cloud Foundry with Spring Framework.  
Java
- test-travis1  
Java

1,196 contributions in the last year

Aug Sep Oct Nov Dec Jan Feb Mar Apr May Jun Jul

Mon Wed Fri

Learn how we count contributions.

Less More

Contribution activity

August 2017

Jump to ▾ 2017

KOREA OPEN SOURCE SOFTWARE DEVELOPERS LAB

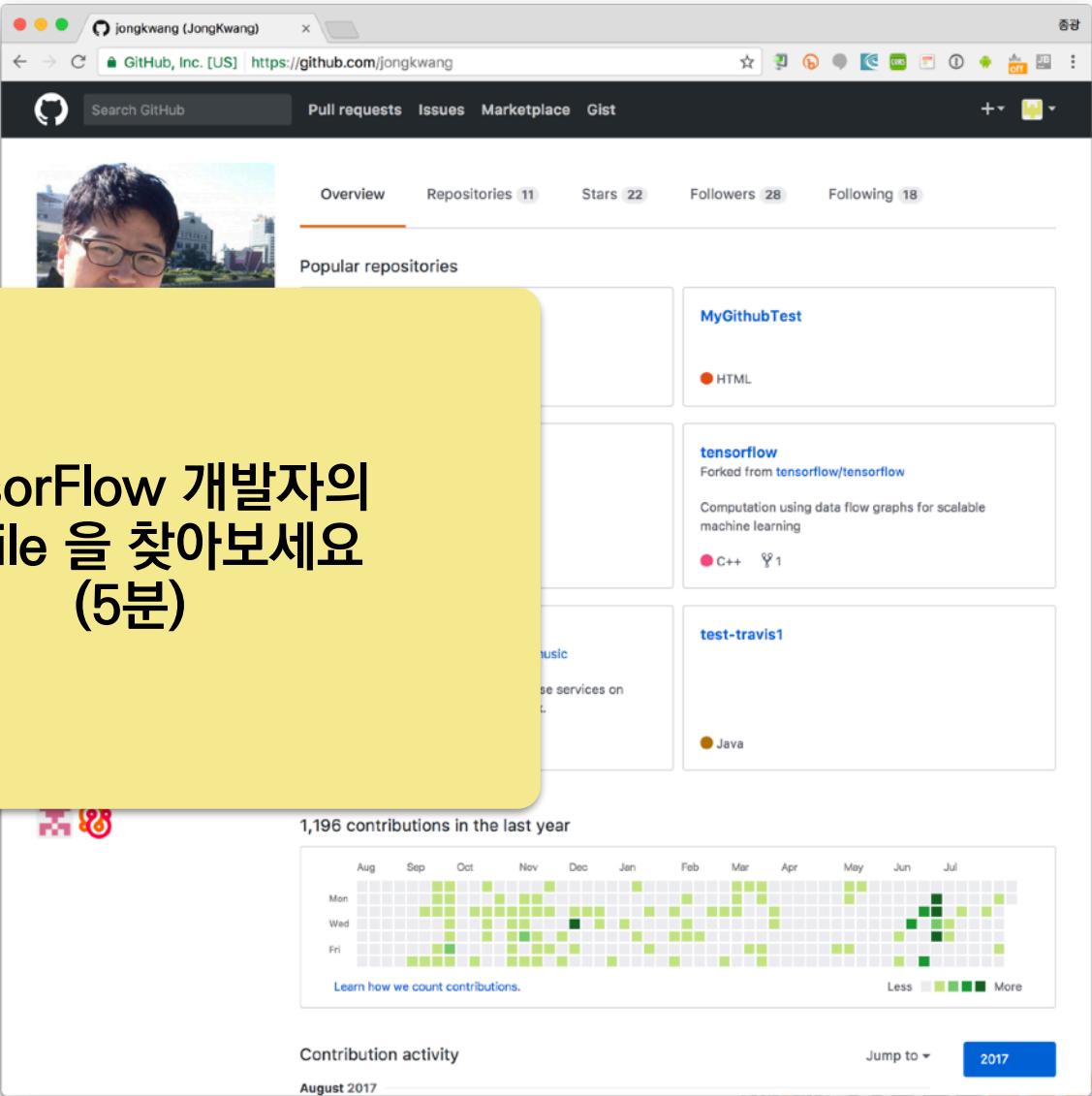
# Profile

## ▶ Profile

- Repository
- Contributions
- Activity



TensorFlow 개발자의  
Profile 을 찾아보세요  
(5분)



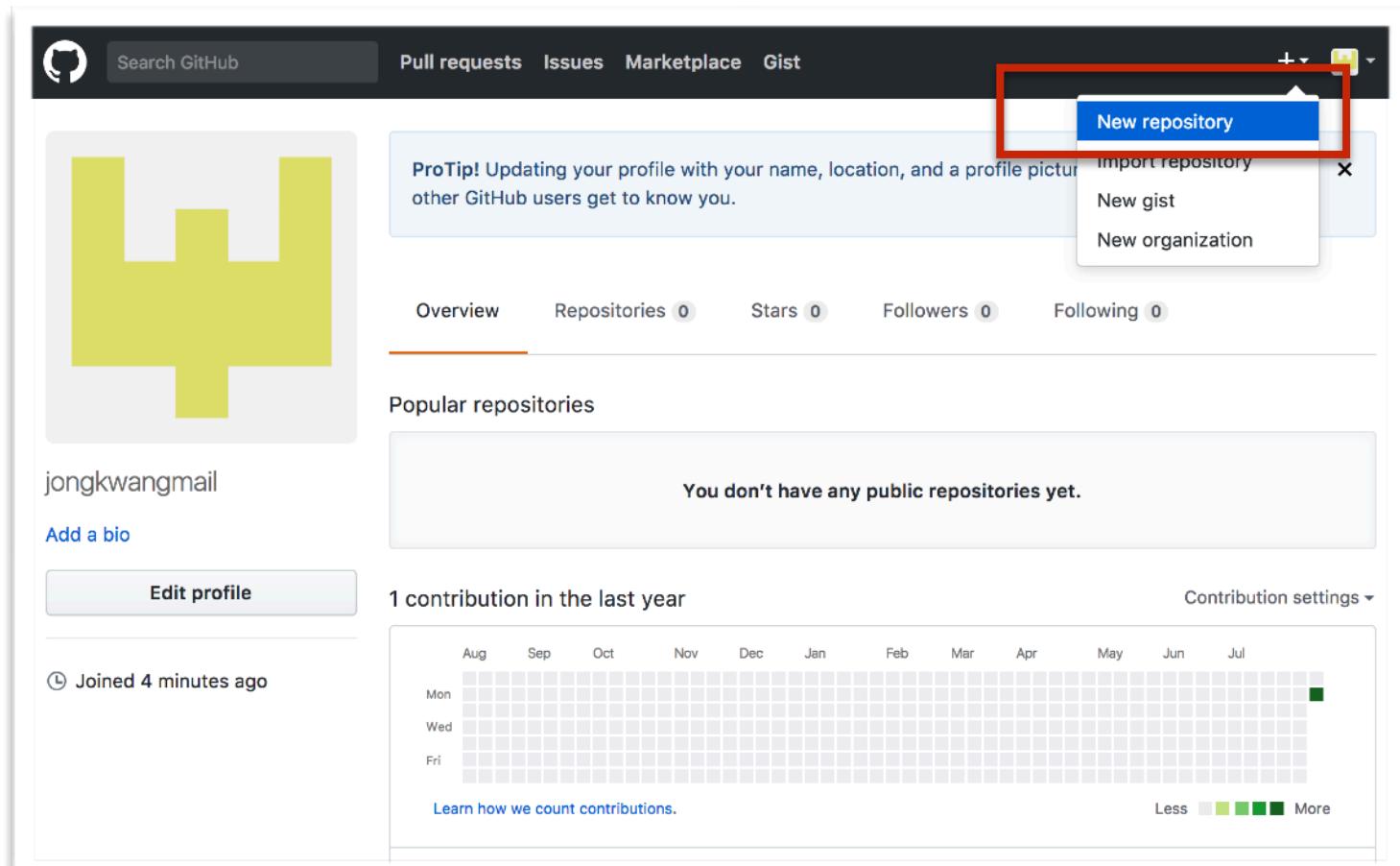
The screenshot shows a GitHub profile for the user 'jongkwang'. The profile includes a photo of a person with glasses, a bio section, and tabs for Overview, Repositories (11), Stars (22), Followers (28), and Following (18). Below the tabs, there's a section for 'Popular repositories' featuring 'MyGitHubTest' (HTML), 'tensorflow' (C++), 'music' (Java), and 'test-travis1'. At the bottom, there's a heatmap visualization of contribution activity for the last year, with a legend indicating 'Less' (light green) and 'More' (dark green).

Git

New Repository

# New Repository

- ▶ 새로운 Repository 생성
  - 확인 전까지 Repository 생성이 안된다



# New Repository

Create a new repository

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

Owner: jongkwang / Repository name: test-github

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

Description (optional):

Public: Anyone can see this repository. You choose who can commit.

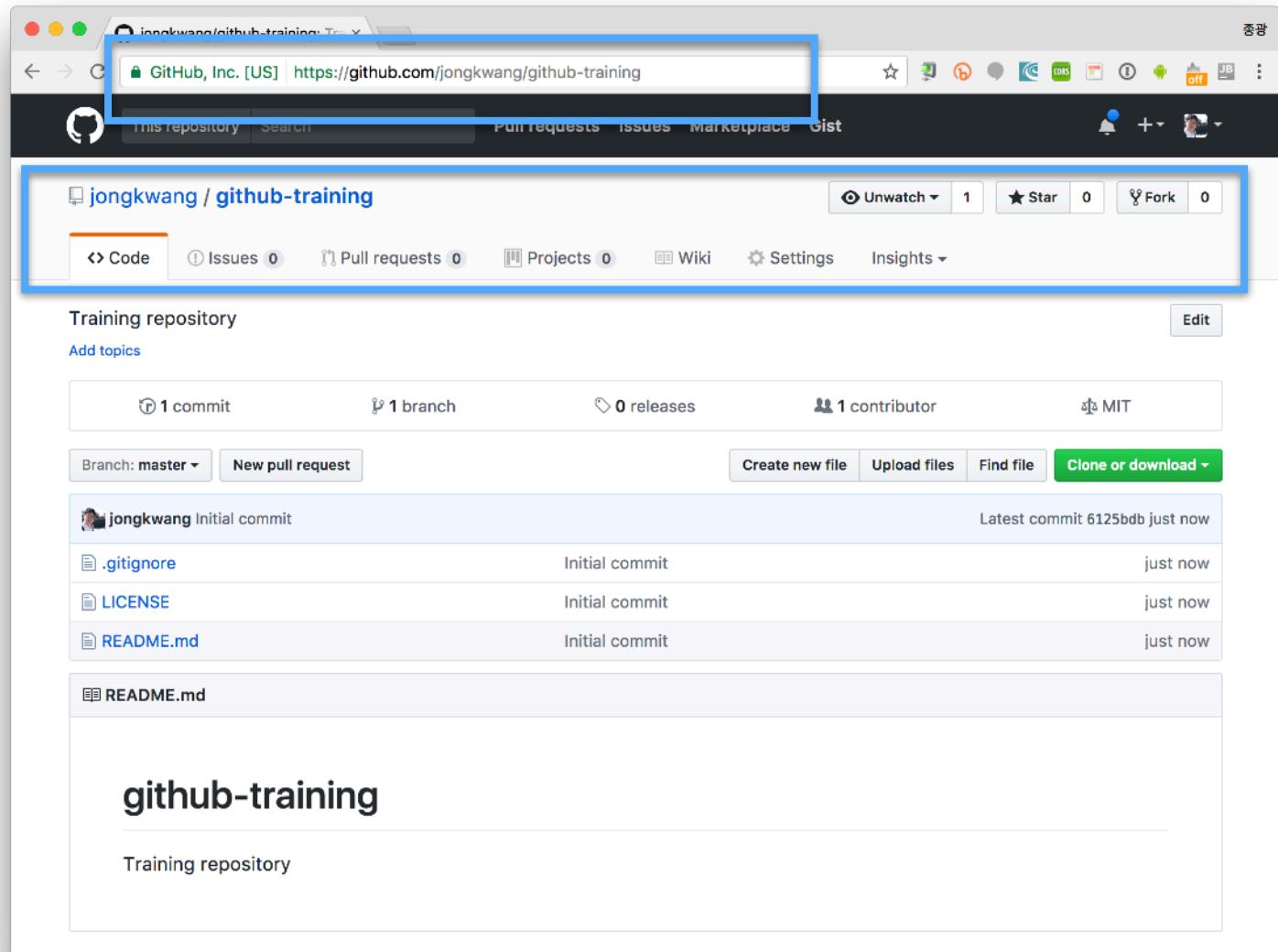
Private: You choose who can see and commit to this repository.

Initialize this repository with a README: This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: None | Add a license: None | ⓘ

**Create repository**

# New Repository



The screenshot shows a GitHub repository page for 'jongkwang / github-training'. The URL 'https://github.com/jongkwang/github-training' is highlighted in a blue box at the top. Below it, the repository name 'jongkwang / github-training' and the main navigation bar are also highlighted in a blue box. The repository is described as a 'Training repository'. It contains 1 commit, 1 branch, 0 releases, 1 contributor, and is licensed under MIT. The latest commit was made just now by 'jongkwang'. The repository includes files like '.gitignore', 'LICENSE', and 'README.md'. The 'README.md' file is shown expanded, containing the text 'github-training' and 'Training repository'.

GitHub, Inc. [US] | <https://github.com/jongkwang/github-training>

jongkwang / [github-training](#)

Code Issues 0 Pull requests 0 Projects 0 Wiki Settings Insights

1 commit 1 branch 0 releases 1 contributor MIT

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

jongkwang Initial commit Latest commit 6125bdb just now

.gitignore Initial commit just now

LICENSE Initial commit just now

README.md Initial commit just now

github-training

Training repository

Git - 그림설명

Git Install

# Git Install

## ▶ Windows

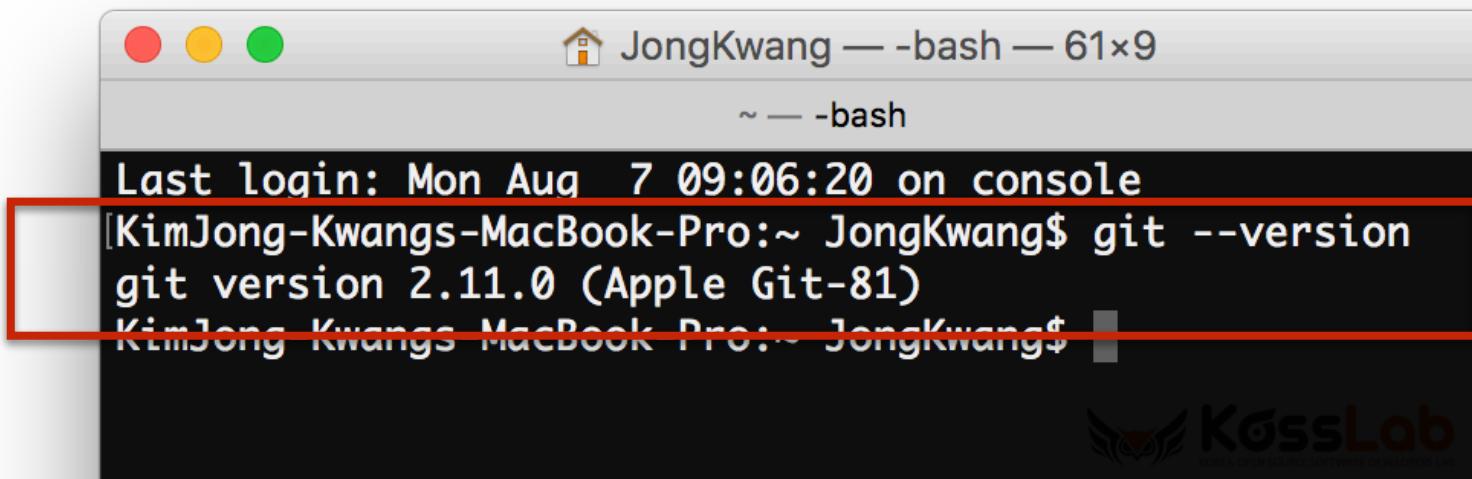
- <http://git-scm.com/download/win>

## ▶ macOS

- Terminal에서 “git” 실행
  - Mavericks(10.9) 부터 없으면 설치된다.
- 직접 설치 : <http://git-scm.com/download/mac>

## ▶ 설치 확인

- Terminal : git --version



The screenshot shows a terminal window titled "JongKwang — -bash — 61x9". The window contains the following text:

```
Last login: Mon Aug  7 09:06:20 on console
[KimJong-Kwangs-MacBook-Pro:~ JongKwang$ git --version
git version 2.11.0 (Apple Git-81)
KimJong Kwangs MacBook Pro:~ JongKwang$ ]
```

A red rectangular box highlights the command "git --version" and its output "git version 2.11.0 (Apple Git-81)".

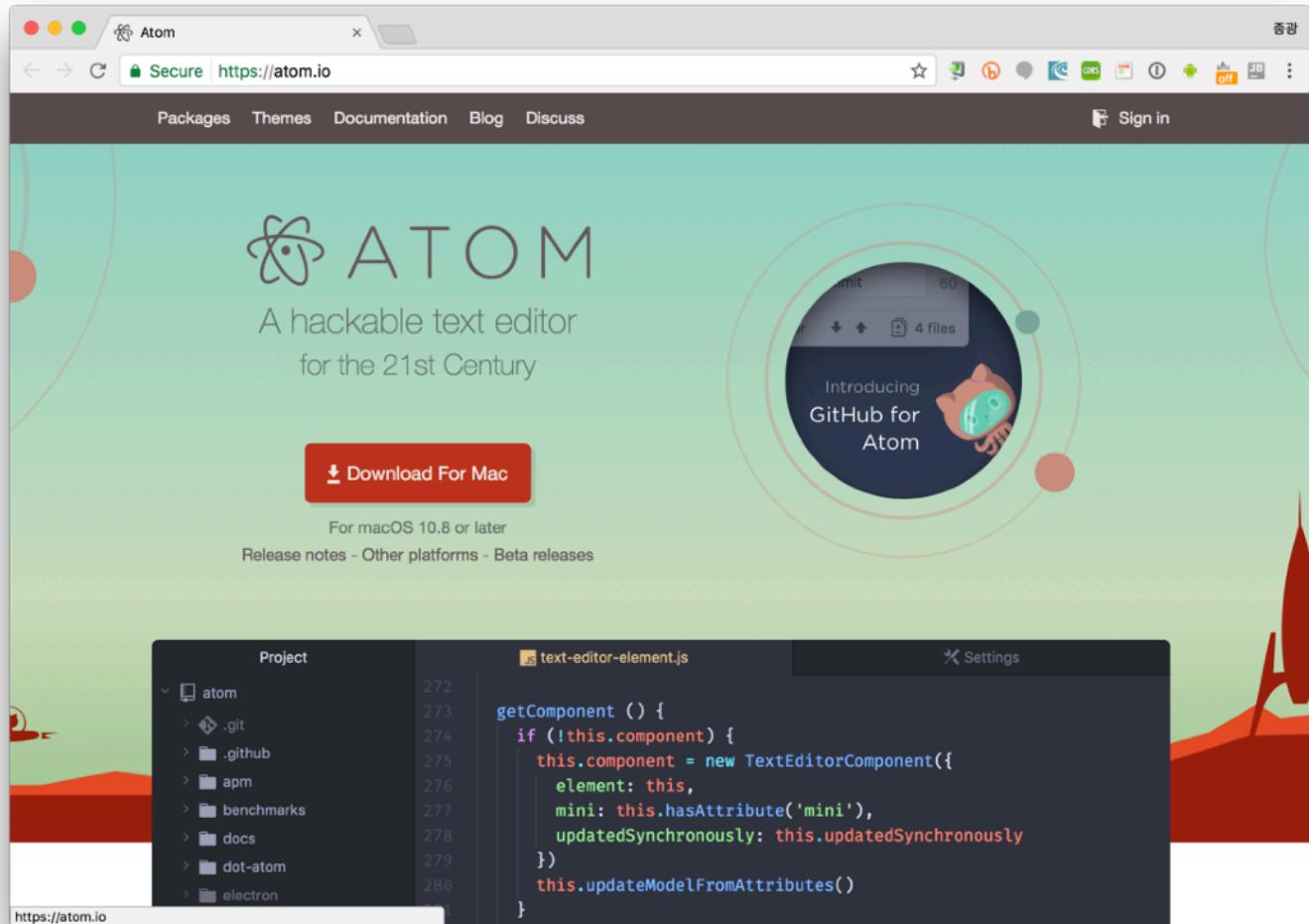
Git - 그림설명

ATOM

Text editor

# ATOM Install

› <https://atom.io>



Git - 그림설명

SourceTree

# SourceTree Install

› <https://www.sourcetreeapp.com>

The screenshot shows the SourceTree website with the Mac OS X version of the application open. On the left, there's a promotional text: "Simplicity and power in a beautiful Git GUI". Below it are two download buttons: "Download for Mac OS X" and "Also available for Windows". On the right, the SourceTree application window is displayed, showing a Git commit history graph and a detailed list of commits. The commits are as follows:

Commit	Author	Description	Date
b7358c7	Rahul Chhab...	[r master] [r origin/master] [r origin/HEAD] Removing ol...	Mar 3, 2016, 11...
bdb8bef	Rahul Chhab...	Merged in update-google-verification (pull request #14)	Feb 18, 2016, 1:3...
dfe975d	Tyler Tadej...	[r origin/update-google-verification] Update google verificati...	Feb 11, 2016, 2:2...
3bc3290	Tyler Tadej...	Replace outdated Atlassian logo in footer with base-64 en...	Feb 11, 2016, 2:1...
dba4719	Tyler Tadej...	Add glistenore	Feb 11, 2016, 1:3...
ff67b45	Mike Minns...	Updated Mac min-spec to 10.10	Feb 15, 2016, 11:...
72d32a8	Michael Min...	Merged in hero_images (pull request #13)	Feb 15, 2016, 10:...
246c4ff	Joel Unger...	[r origin/hero_images] [r hero_images] Used TinyPng to c...	Feb 11, 2016, 3:3...
9d9438c	Joel Unger...	Replacing hero images with new version of SourceTree	Feb 9, 2016, 2:59...
ce75b63	Michael Min...	Merged in bug/date-https (pull request #12)	Feb 15, 2016, 10:...
85367bb	Patrick Tho...	[r origin/bug/date-https] fixed date and https errors	Jan 7, 2016, 12:2...
4f9b557	Joel Unger...	New Favicon	Feb 8, 2016, 3:55...
384e6d5	Rahul Chhab...	[r origin/search-console-access] search console google ver...	Feb 3, 2016, 2:09...
6fa47e9	Mike Minns...	updated to move supported version to OSX 10.9+	Dec 15, 2015, 2:0...
8dd87bb	Mike Minns...	remove extra , when a line is skipped due to empty server	Nov 23, 2015, 2:2...
faa195e	Mike Minns...	Skip records with empty server/user id as gas rejects them	Nov 23, 2015, 2:1...
0cdfe96	Mike Minns...	corrected paths after merge	Nov 23, 2015, 2:0...
051ab1b	Mike Minns...	corrected column counting	Nov 23, 2015, 1:5...
a723bc2	Mike Minns...	Merge branch 'au2gex'	Nov 23, 2015, 1:5...
65fd580	Mike Minns...	deal with invalid instanceids	Nov 23, 2015, 1:5...
500a892	Michael Min...	Merged in au2gex (pull request #11)	Nov 23, 2015, 1:0...

**A free Git client for Windows and Mac**

SourceTree simplifies how you interact with your Git repositories so you can focus on coding. Visualize and manage your repositories through SourceTree's simple Git GUI.

# SourceTree Install

## ▶ Github 선택

The screenshot shows the SourceTree setup process. On the left, a sidebar lists several options with checkboxes: 설치 (checked), 라이선스 계약 (checked), Atlassian 계정 (checked), 원격 (unchecked), Install tools (unchecked), 시작 저장소 (unchecked), and Clone repository (unchecked). On the right, a "Connect an Account" window is open. It displays three service icons: Bitbucket, Bitbucket Server, and GitHub. The GitHub icon is highlighted with a red rectangle. Below the icons, there are input fields for "호스트 URL:" containing "https://github.com/" and "인증" dropdown set to "OAuth". At the bottom are two buttons: "설정 건너뛰기" and "계속".

Atlassian  
**SourceTree**

- 설치
- 라이선스 계약
- Atlassian 계정
- 원격
- Install tools
- 시작 저장소
- Clone repository

Connect an Account

Connect to a remote server to clone existing repositories. If you don't already have a Bitbucket account, you can [sign up for free](#).

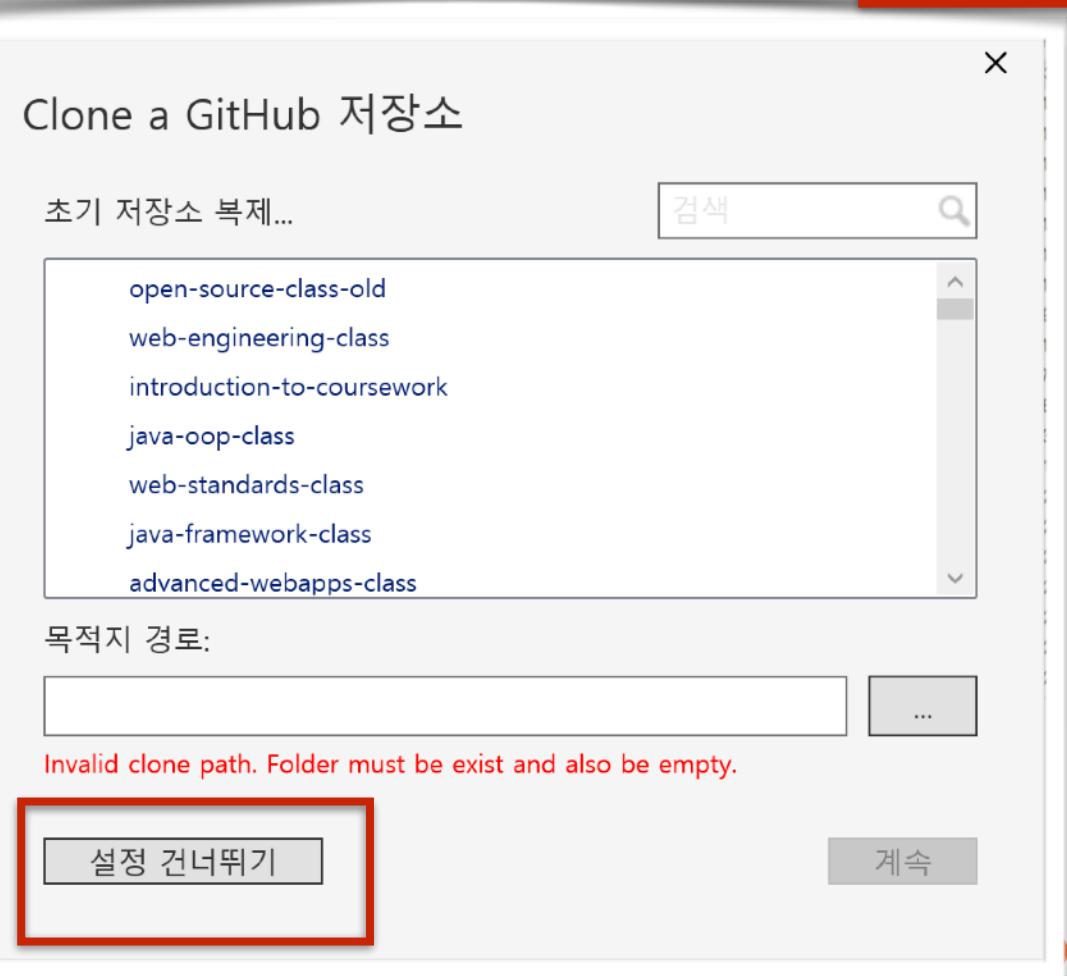
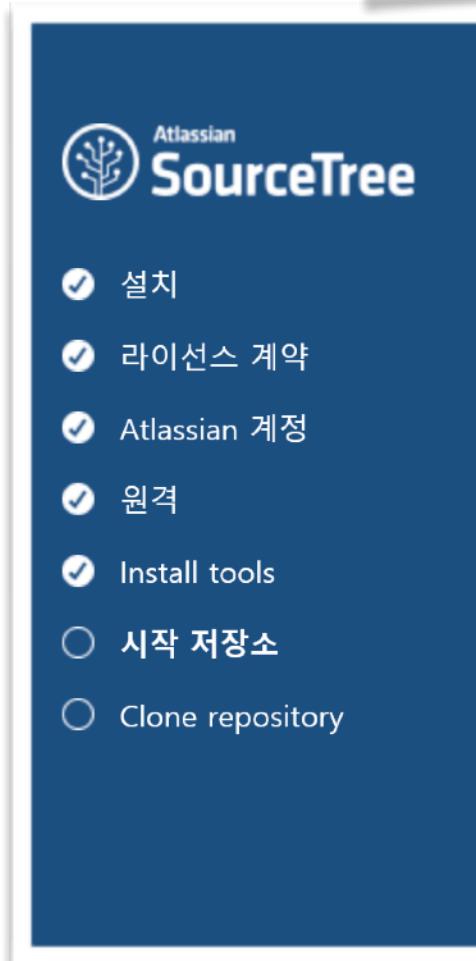
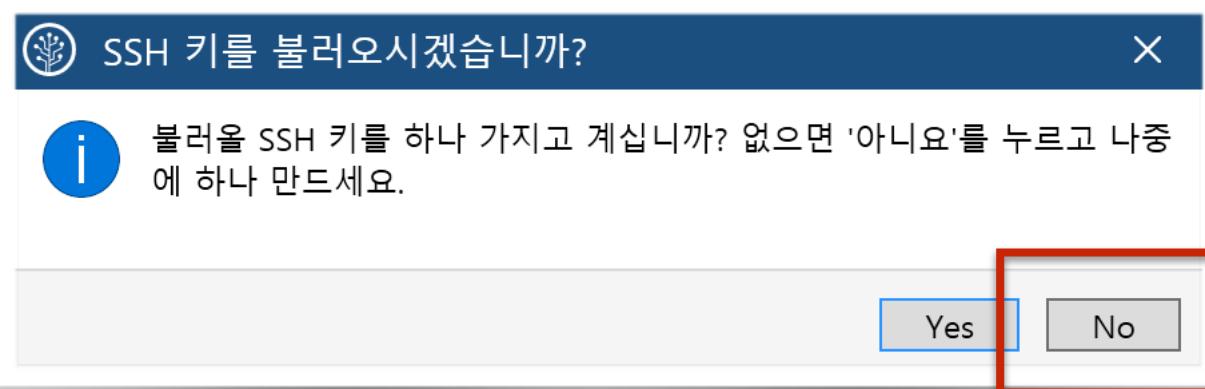
Bitbucket      Bitbucket Server      GitHub

호스트 URL:

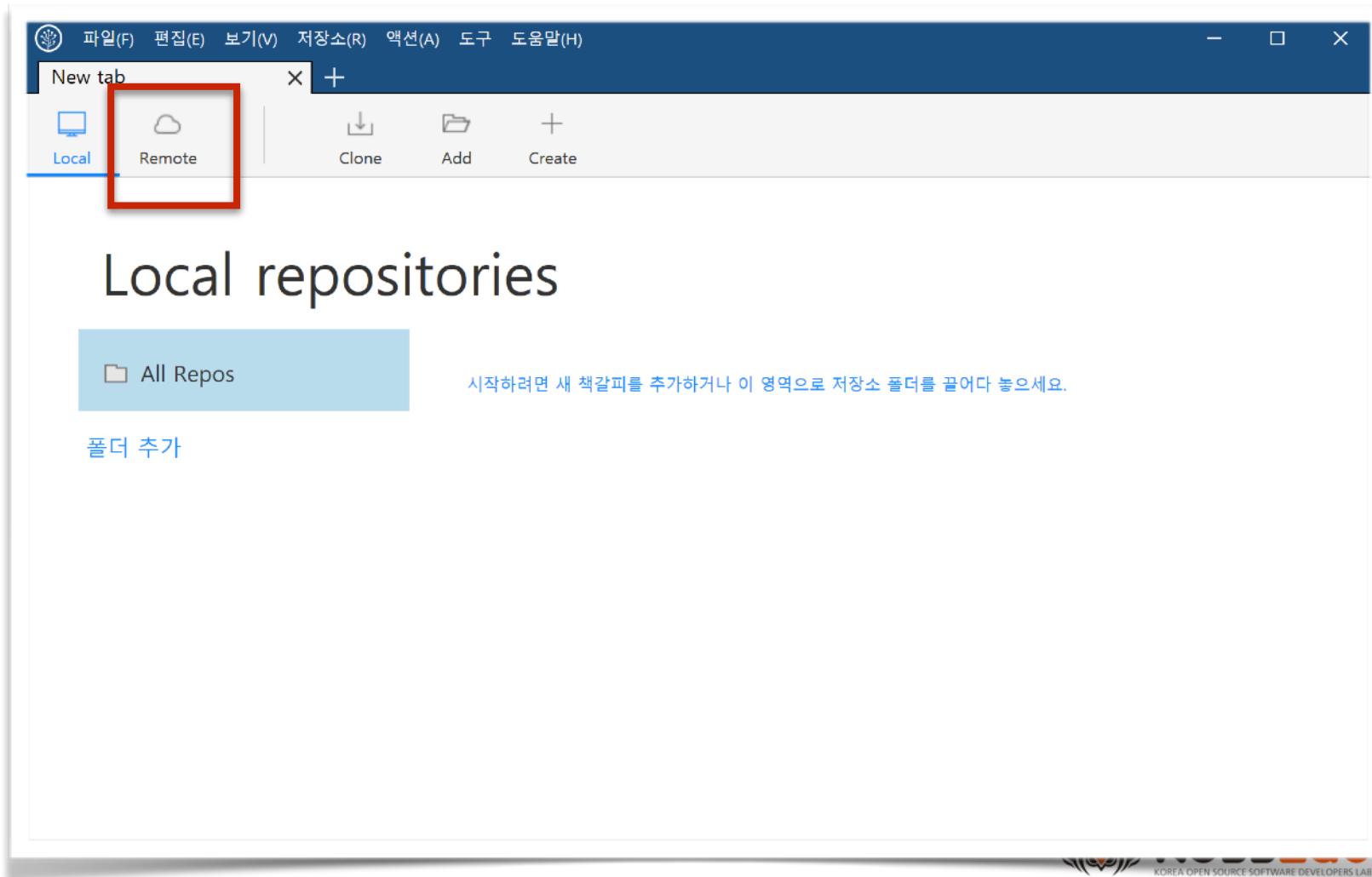
인증

설정 건너뛰기      계속

# SourceTree



# SourceTree Install



# SourceTree Install

The screenshot shows the SourceTree application interface. At the top, there is a menu bar with Korean labels: 파일(F), 편집(E), 보기(V), 저장소(R), 액션(A), 도구, and 도움말(H). Below the menu is a toolbar with a 'New tab' button, a close button ('X'), and a plus sign button. The toolbar also includes icons for Local (computer monitor) and Remote (cloud), along with Clone, Add, and Create buttons.

The main area displays 'Remote repositories'. On the left, there is a profile for 'jongkwang GitHub' with a '+' button and a '계정 추가...' link. In the center, a search bar contains the text 'training'. To the right of the search bar is a '새로고침' (refresh) button. Below the search bar, there is a repository entry for 'github-training GitHub'. This entry features a blue circle icon with white '</>' symbols, the repository name 'github-training', and the word 'GitHub'. To the right of this entry is a 'Clone' button, which is highlighted with a red rectangular box.

At the bottom right of the application window, there is a small footer text: 'KOREA OPEN SOURCE SOFTWARE DEVELOPERS LAB'.

# SourceTree Install

The screenshot shows the SourceTree application interface. The title bar includes the SourceTree logo, file menu (파일(F)), edit menu (편집(E)), view menu (보기(V)), location menu (저장소(R)), options menu (옵션(A)), tools menu (도구), and help menu (도움말(H)). A tab bar shows 'New tab' and a '+' icon. Below the tabs are icons for Local (computer monitor) and Remote (cloud), followed by 'Clone' (highlighted with a blue underline), 'Add', and 'Create'. The main area is titled 'Clone' and contains instructions: 'Cloning is even easier if you set up a [remote account](#)'. It has three input fields: 'Repository URL' with the value 'https://github.com/jongkwang/github-training', a 'Browse' button, and a note '저장소 종류: Git 저장소입니다'; 'Local Path' with the value 'C:\Users\JongKwang\Documents\github-training', a 'Browse' button; and 'Local Folder' with the value '[루트]', a dropdown arrow, and a note '(▼) 고급 옵션'. A red box highlights the 'Clone' button at the bottom.

New tab +

Local Remote Clone Add Create

## Clone

Cloning is even easier if you set up a [remote account](#)

Repository URL:  Browse  
저장소 종류: (?) Git 저장소입니다

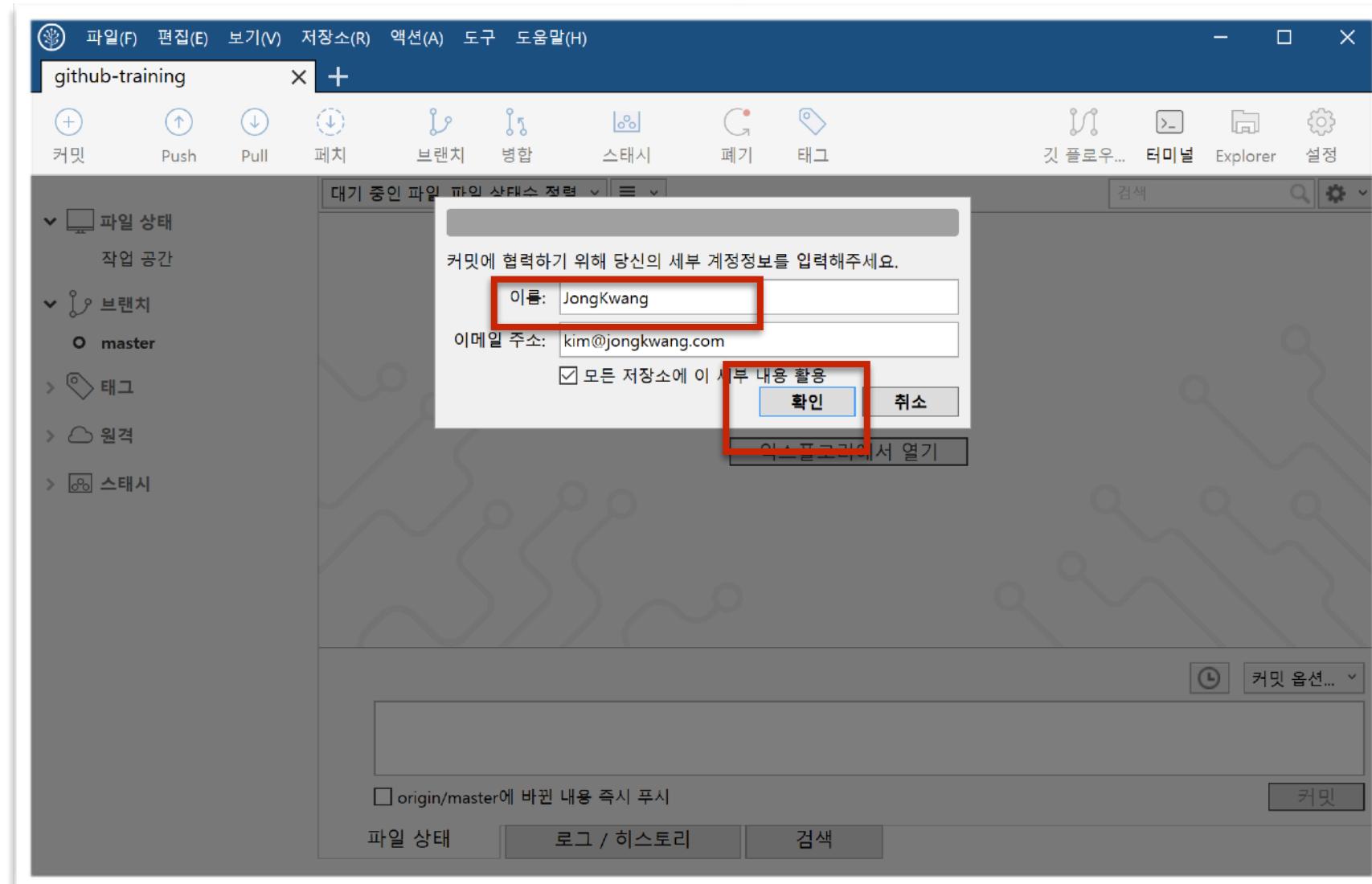
Local Path:  Browse

Local Folder:  ▼

(▼) 고급 옵션

클론

# SourceTree Install

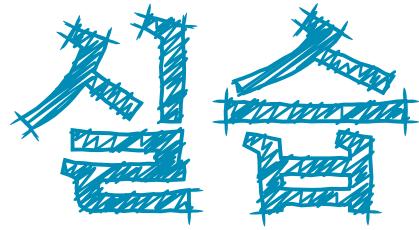


# SourceTree Install

## › 여기까지 오신분은

- 탐색기에서 해당 폴더를 살펴 봅니다
- SourceTree 를 살펴봅니다

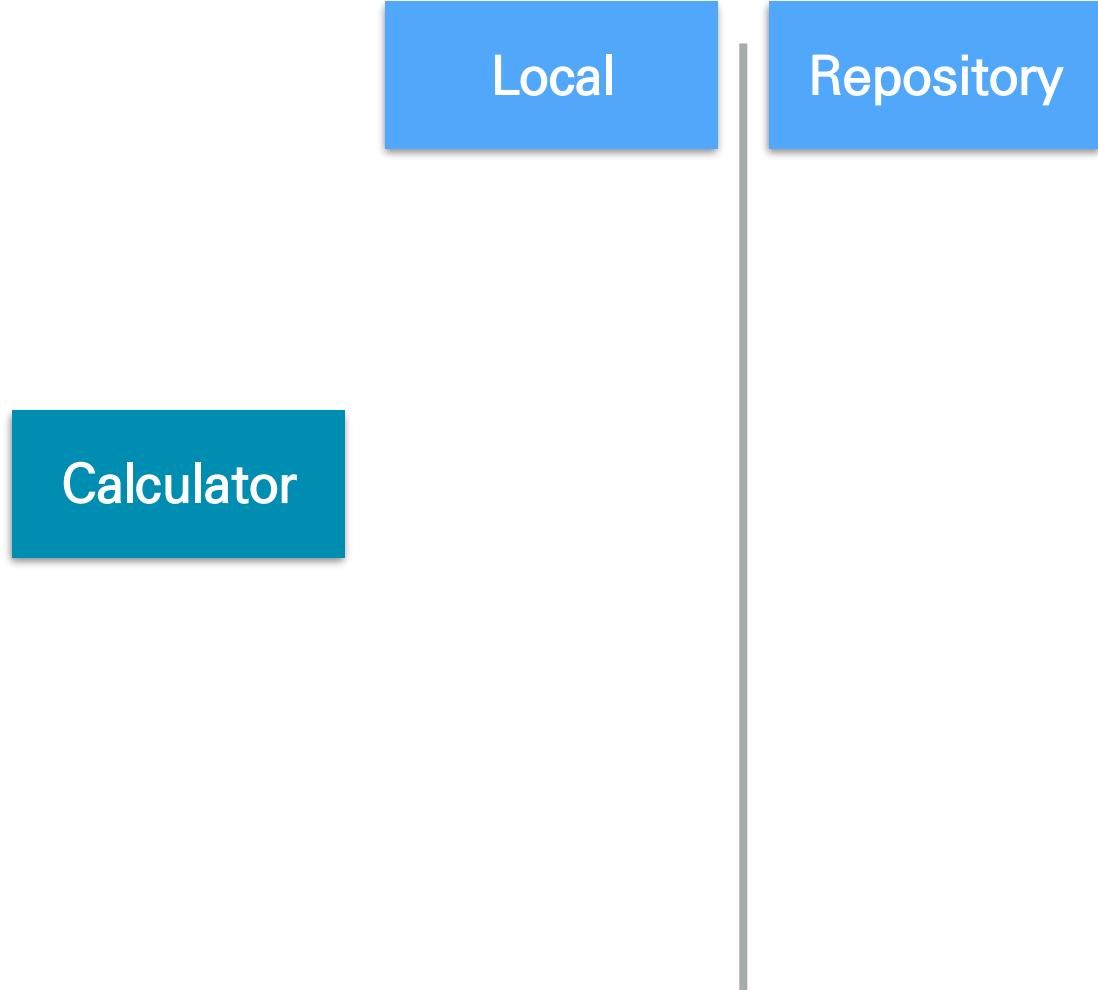
# Git - 그림설명



# Preview

# Git - 그림 설명

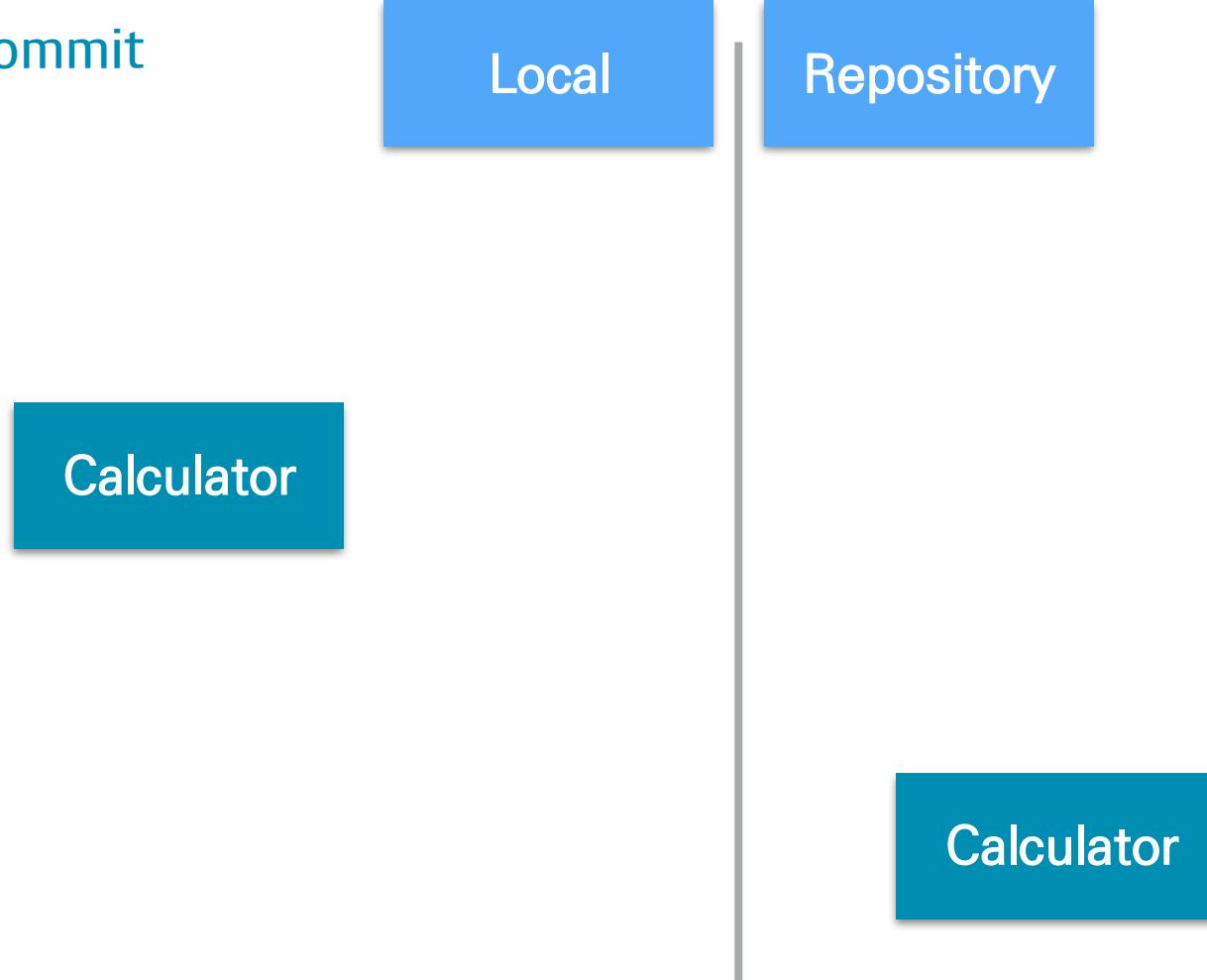
- › 계산기 클래스 생성



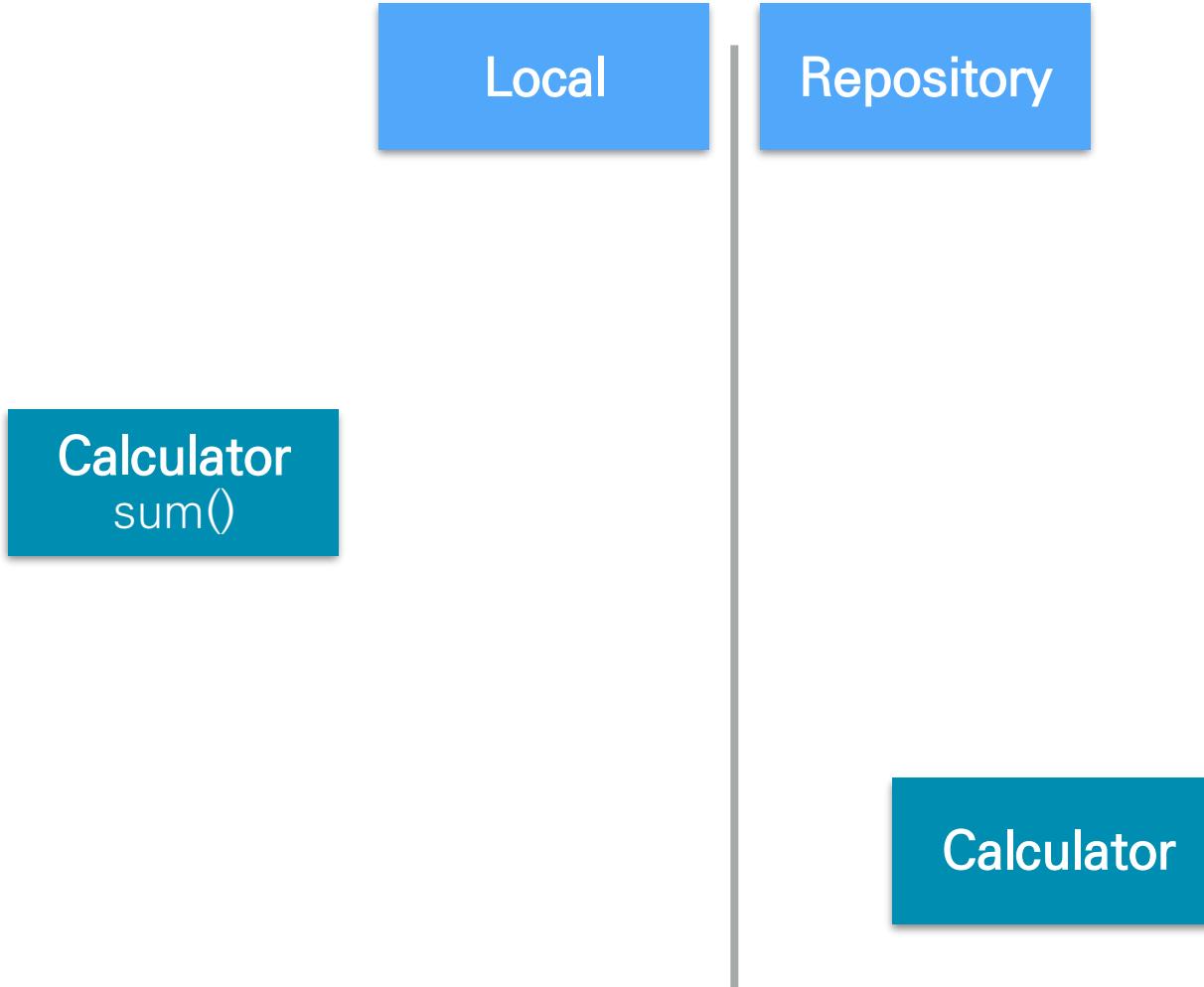
# Git – 그림설명

## ▶ Repository 에 저장

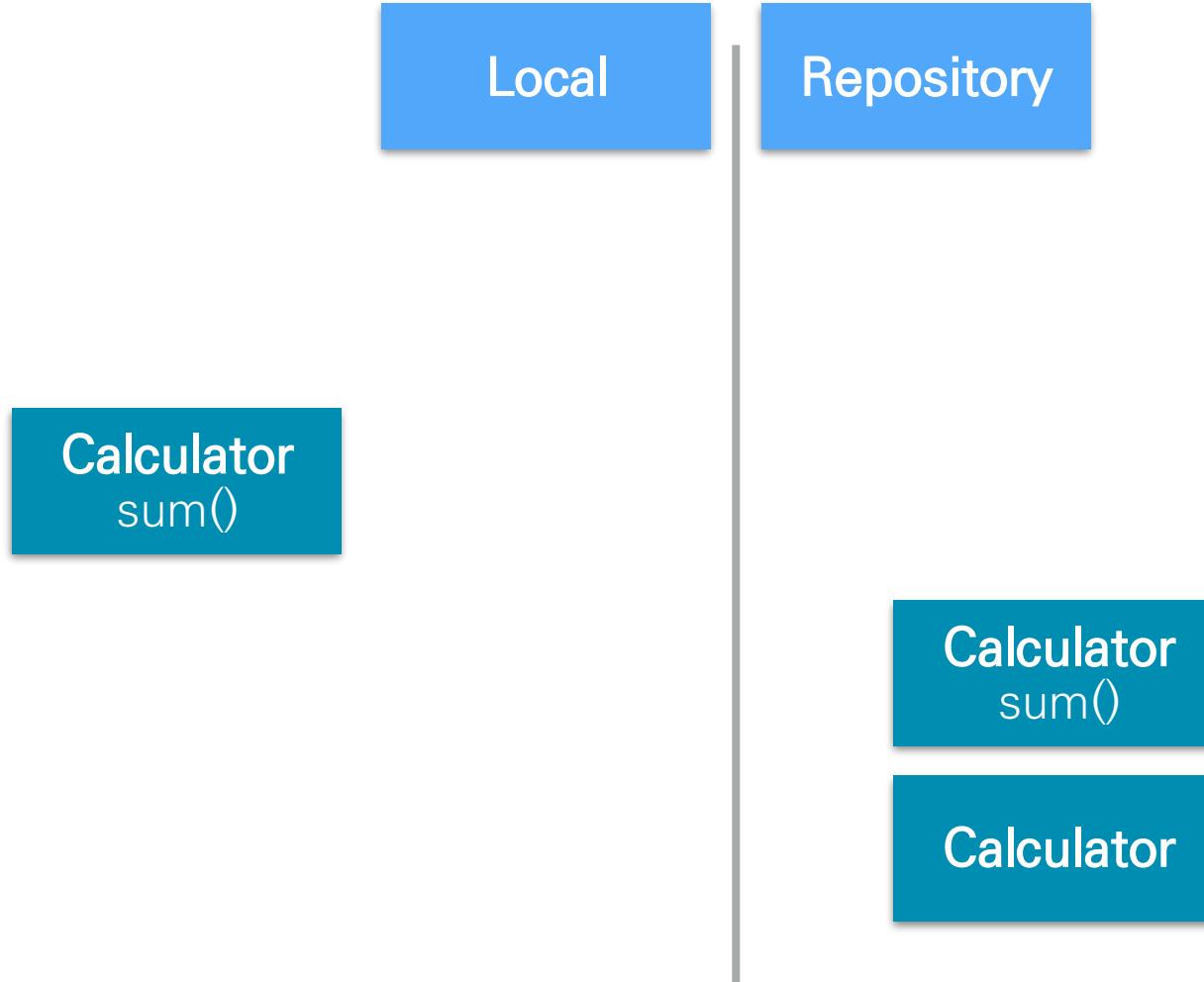
- Commit



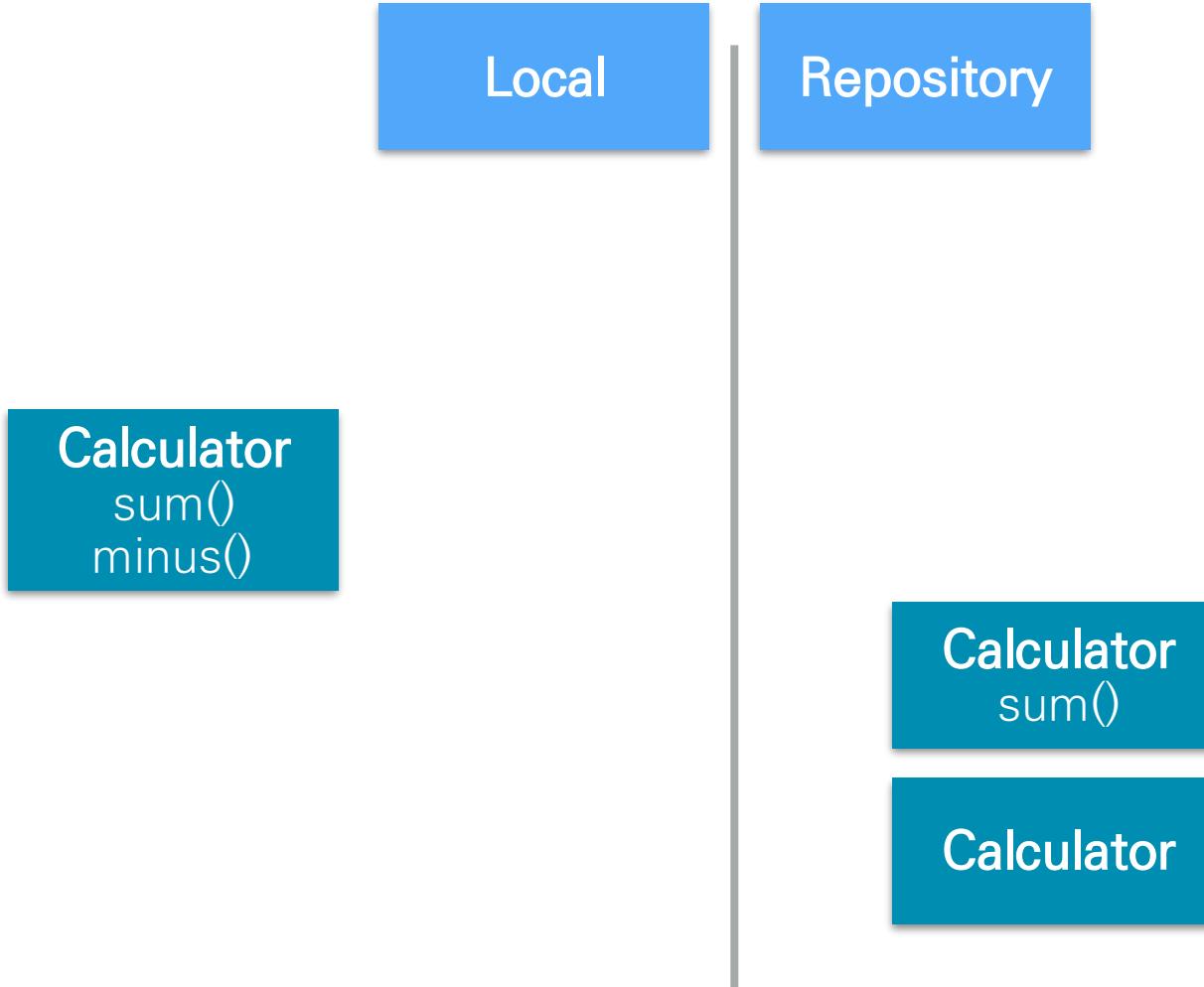
# Git - 그림 설명



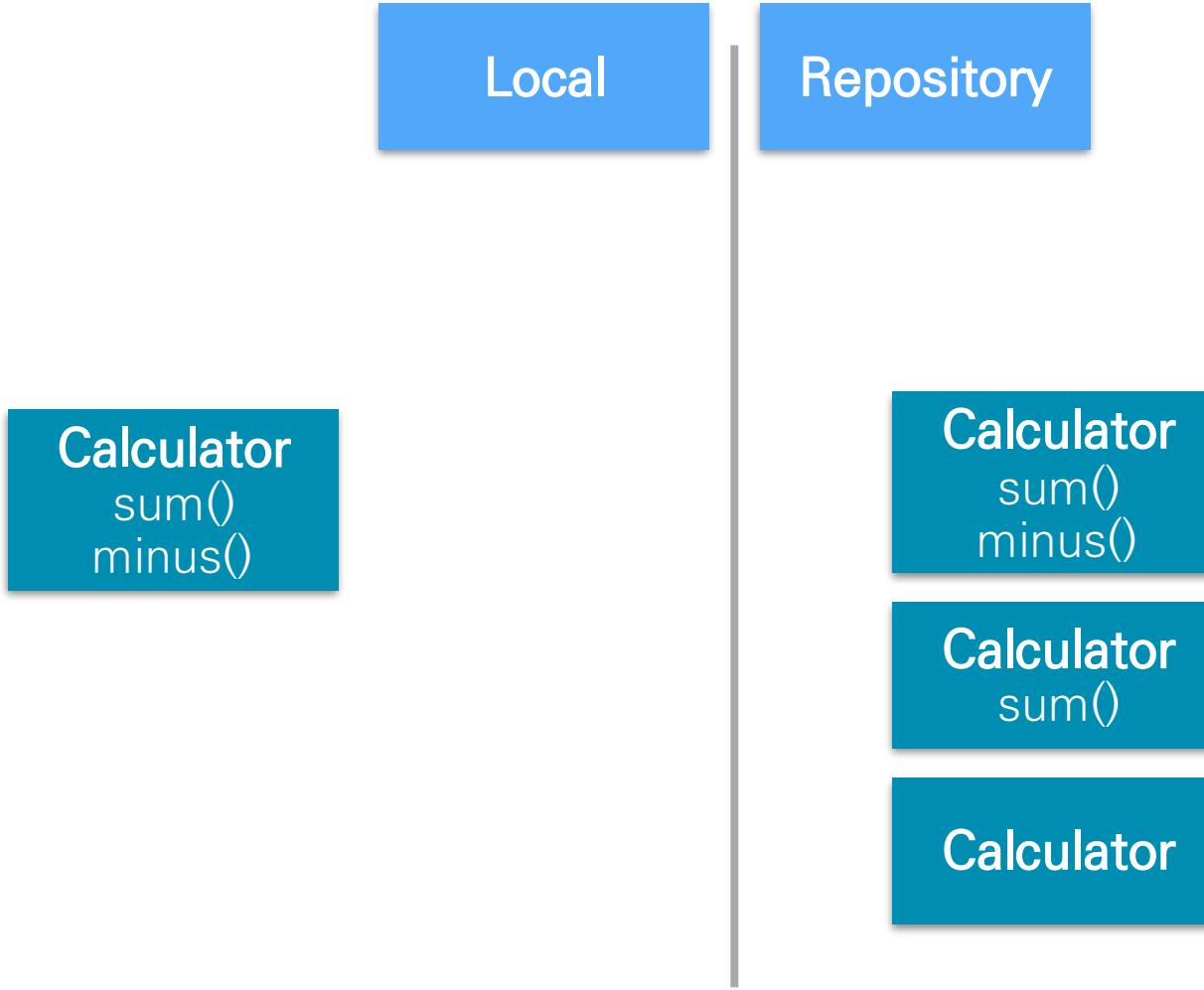
# Git - 그림 설명



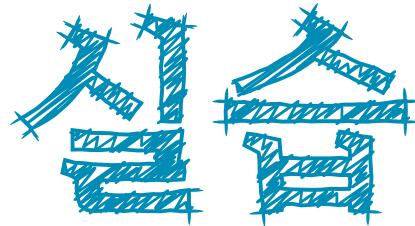
# Git – 그림 설명



# Git – 그림 설명



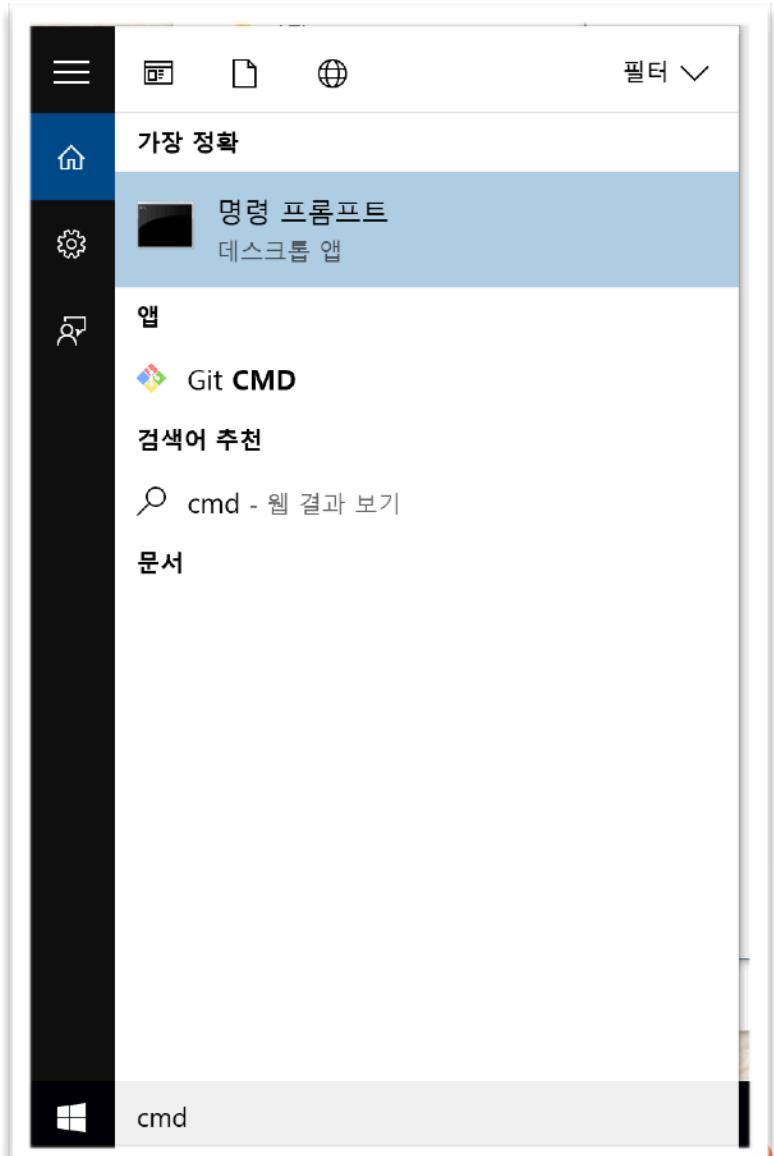
설습 – with SourceTree



with SourceTree

# 실습 – with SourceTree

## ▶ cmd 실행





## 명령 프롬프트

Microsoft Windows [Version 10.0.10240]  
(c) 2015 Microsoft Corporation. All rights reserved.

C:\Users\JongKwang>cd C:\Users\JongKwang\Documents\test-github

C:\Users\JongKwang\Documents\test-github>dir

C 드라이브의 볼륨: BOOTCAMP

볼륨 일련 번호: FCDE-4CF5

C:\Users\JongKwang\Documents\test-github 디렉터리

2017-08-07	오후 11:31	<DIR>	.
2017-08-07	오후 11:31	<DIR>	..
2017-08-07	오후 11:31		294 .gitignore
2017-08-07	오후 11:31		1,087 LICENSE
2017-08-07	오후 11:31		13 README.md
		3개 파일	1,394 바이트
		2개 디렉터리	32,227,774,464 바이트 남음

C:\Users\JongKwang\Documents\test-github>atom .

# 실습 - with SourceTree

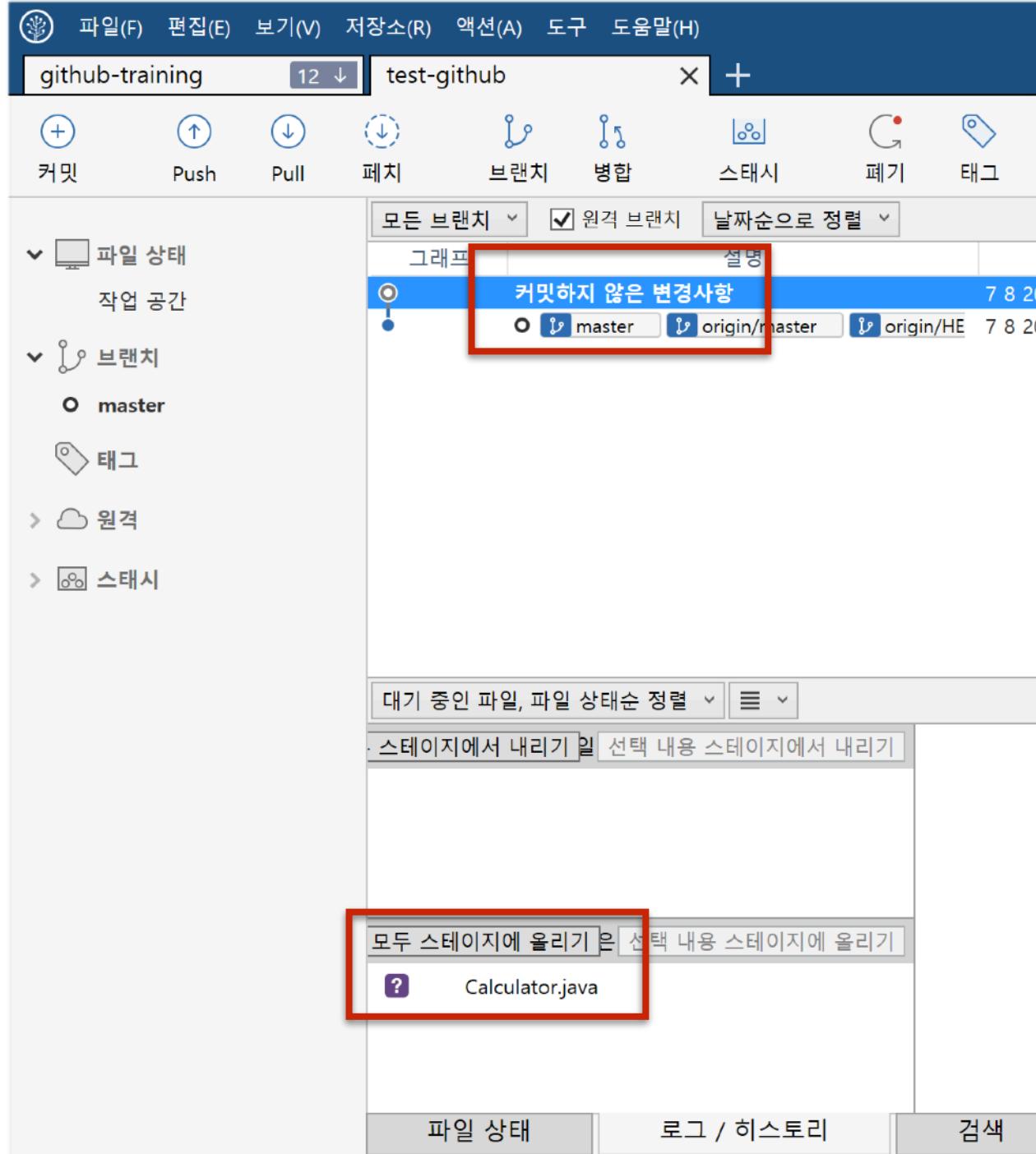
- ▶ Calculator.java 생성
- ▶ Calculator 코딩

The screenshot shows the Atom code editor interface. The title bar reads "Calculator.java — C:\Users\JongKwang\Documents\test-github — Atom". The menu bar includes File, Edit, View, Selection, Find, Packages, and Help. The left sidebar is labeled "Project" and shows a tree view of the project structure: "test-github" expanded, showing ".git", ".gitignore", "Calculator.java" (which is highlighted with a red box), and "LICENSE". Below these are "README.md" and ".idea". The main editor area has a red border around it and displays the following Java code:

```
1 package com.mycompany.test;  
2  
3 public class Calculator {  
4  
5     public static void main(String[] args) {  
6         }  
7     }  
8 }
```

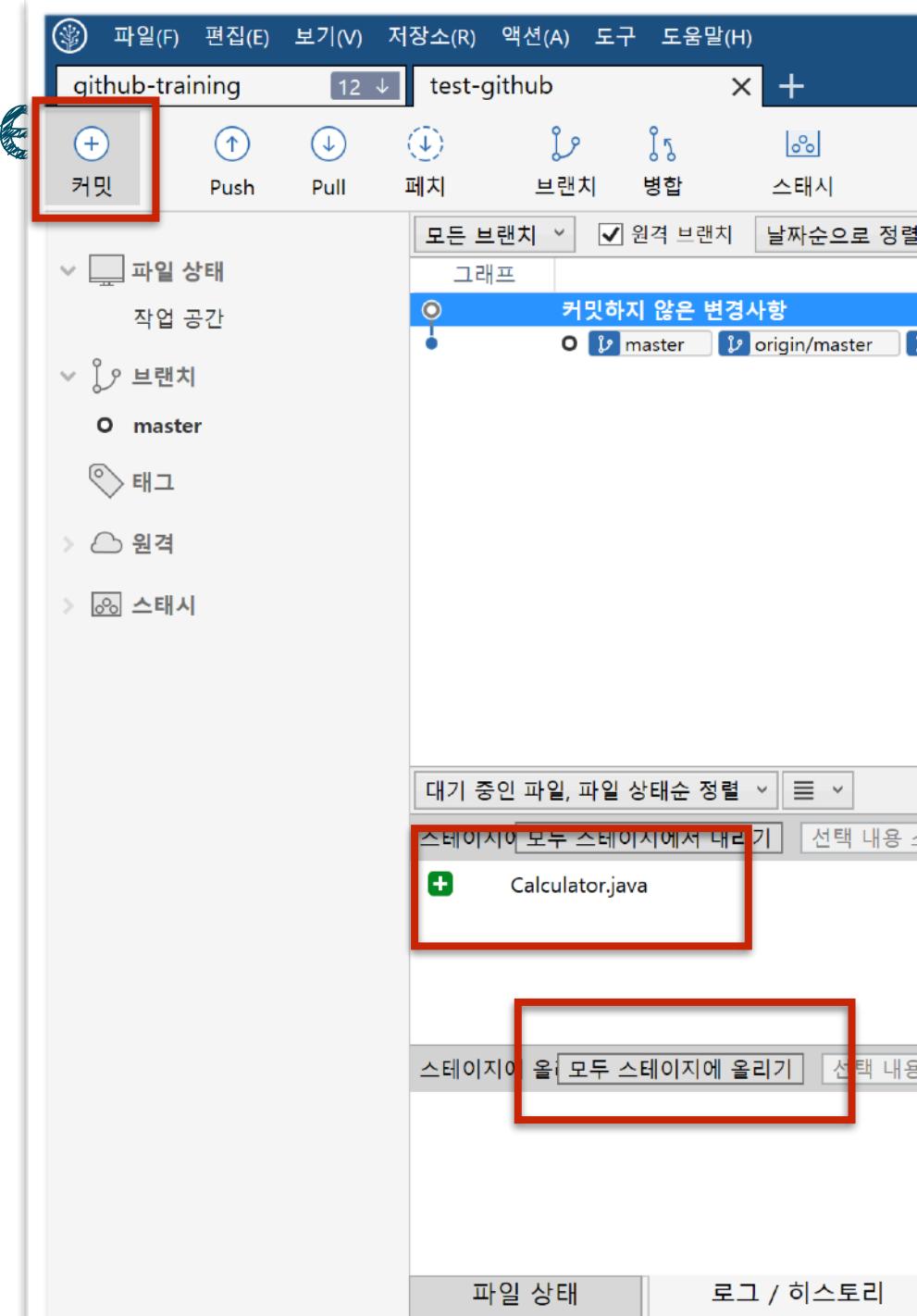
# 실습 - with

- ▶ Unstage에 파일이
- ▶ 커밋하지 않은 변경



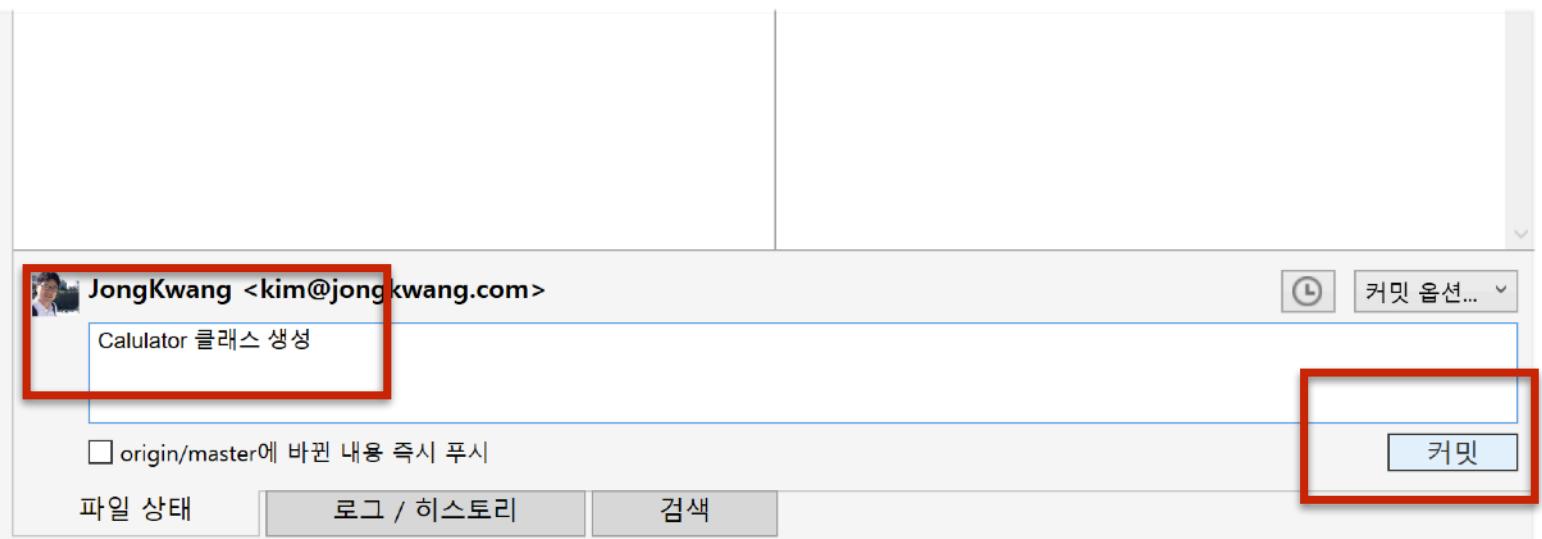
# 실습 - with Source

- ▶ 모두 스테이지에 올리고
- ▶ 커밋



# 실습 - with SourceTree

- ▶ 커밋로그 작성
- ▶ 커밋



파일(F) 편집(E) 보기(V) 저장소(R) 액션(A) 도구 도움말(H)

github-training 12 test-github X +

커밋 Push Pull 폐지 브랜치 병합 스테시 폐기 태그 깃 플로우... 터미널 Explorer 설정

파일 상태 작업 공간  
브랜치 master 1↑  
태그 원격  
스테시

모든 브랜치 원격 브랜치 날짜순으로 정렬  
그래프 설명 날짜 작성자 커밋  
master 1 Calculator 클래스 생성 7 8 2017 23:38 JongKwang <kim@...> 09dd07f  
origin/master origin/HEAD Initial commit 7 8 2017 23:30 JongKwang <kim@...> bdc66b8

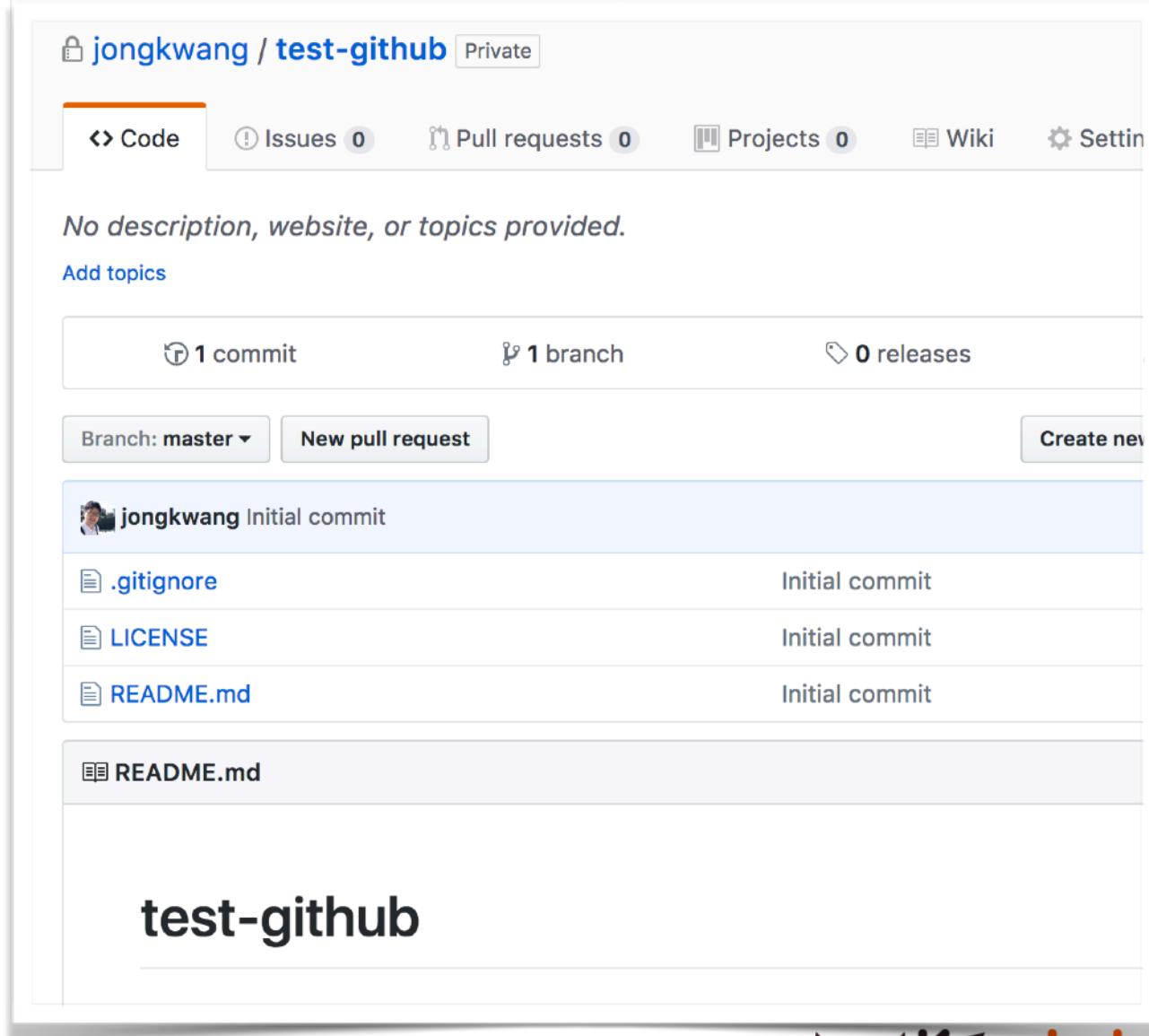
건너뛰기:

파일 상태순 정렬 김색  
커밋: 09dd07fa6f36d1294ffb4fe86df0a4d6d0161908 [09dd07f]  
상위 항목: bdc66b83e6  
작성자: JongKwang <kim@jongkwang.com>  
날짜: 2017년 8월 7일 월요일 오후 11:38:26  
커밋한 사람: JongKwang  
  
Calculator 클래스 생성  
  
Calculator.java 파일 내용 코드 봉지 되돌리기  
1 + package com.mycompany.test;  
2 +  
3 + public class Calculator {  
4 +  
5 + public static void main(String[] args)  
6 + }  
7 + }

파일 상태 로그 / 히스토리 검색

# 실습 – with SourceTree

▶ 하지만  
Github 에는 없다



A screenshot of a GitHub repository page for 'test-github'. The repository is private, created by 'jongkwang'. It has 1 commit, 1 branch, and 0 releases. The branch dropdown shows 'master'. There is a 'New pull request' button and a 'Create new' button. The repository contains three files: '.gitignore', 'LICENSE', and 'README.md', all of which are initial commits. The file 'README.md' is shown expanded at the bottom.

jonkwang / test-github Private

Code Issues 0 Pull requests 0 Projects 0 Wiki Settings

No description, website, or topics provided.

Add topics

1 commit 1 branch 0 releases

Branch: master New pull request Create new

jongkwang Initial commit

.gitignore Initial commit

LICENSE Initial commit

README.md Initial commit

README.md

test-github

# 실습 - wi

## › sum() 코딩

```
Calculator.java
package com.mycompany.test;

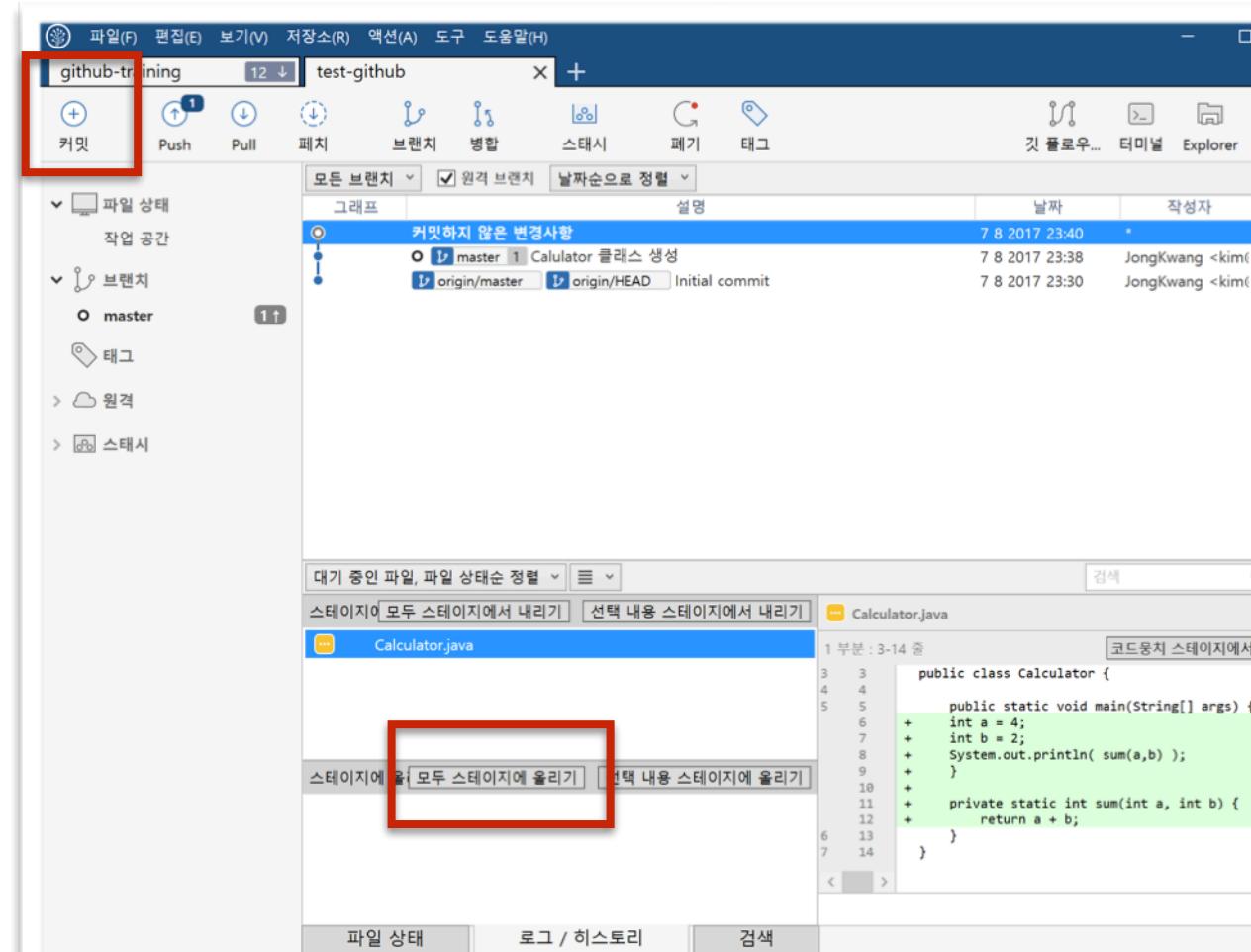
public class Calculator {
    public static void main(String[] args) {
        int a = 4;
        int b = 2;
        System.out.println( sum(a,b) );
    }

    private static int sum(int a, int b) {
        return a + b;
    }
}
```

# 실습 - with SourceTree

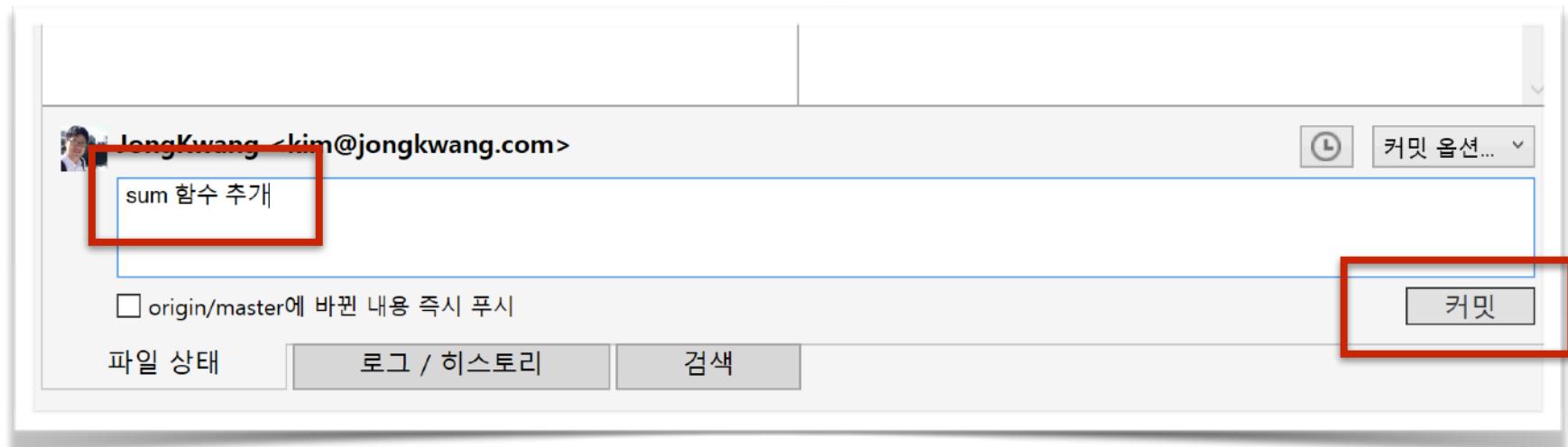
▶ Stage에 올리고

▶ 커밋



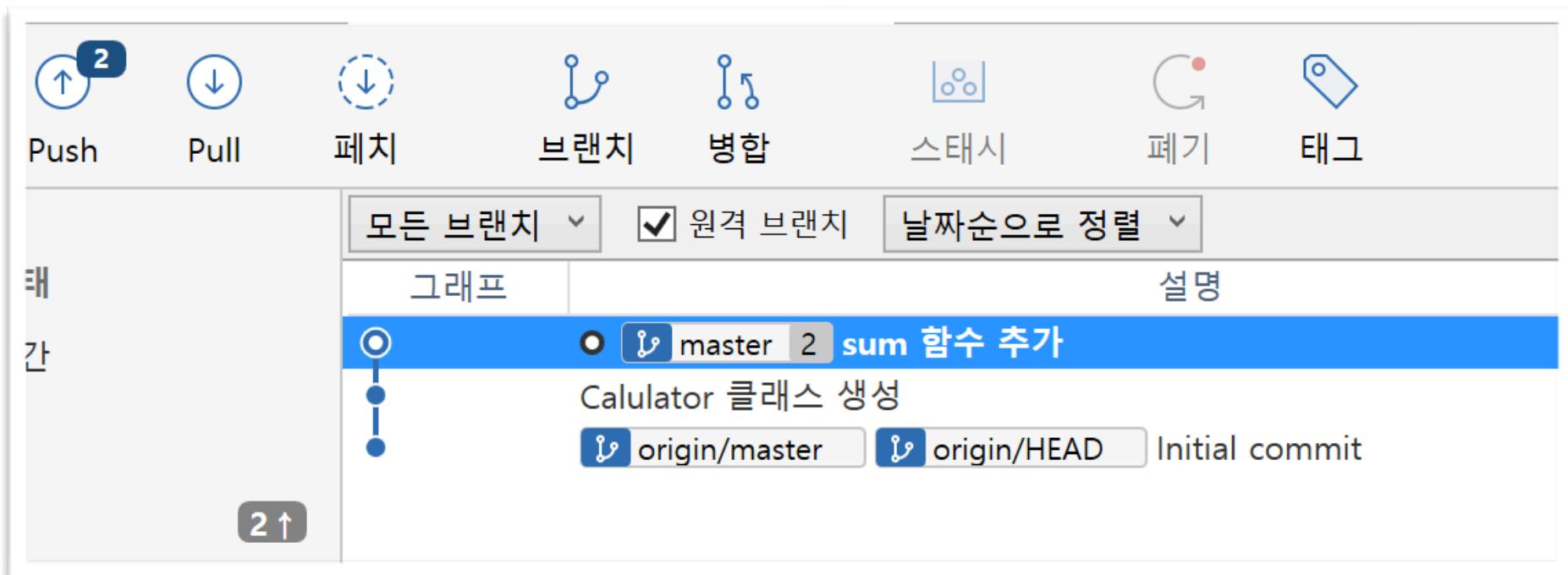
# 실습 - with SourceTree

- ▶ 커밋 로그 작성 후
- ▶ 커밋



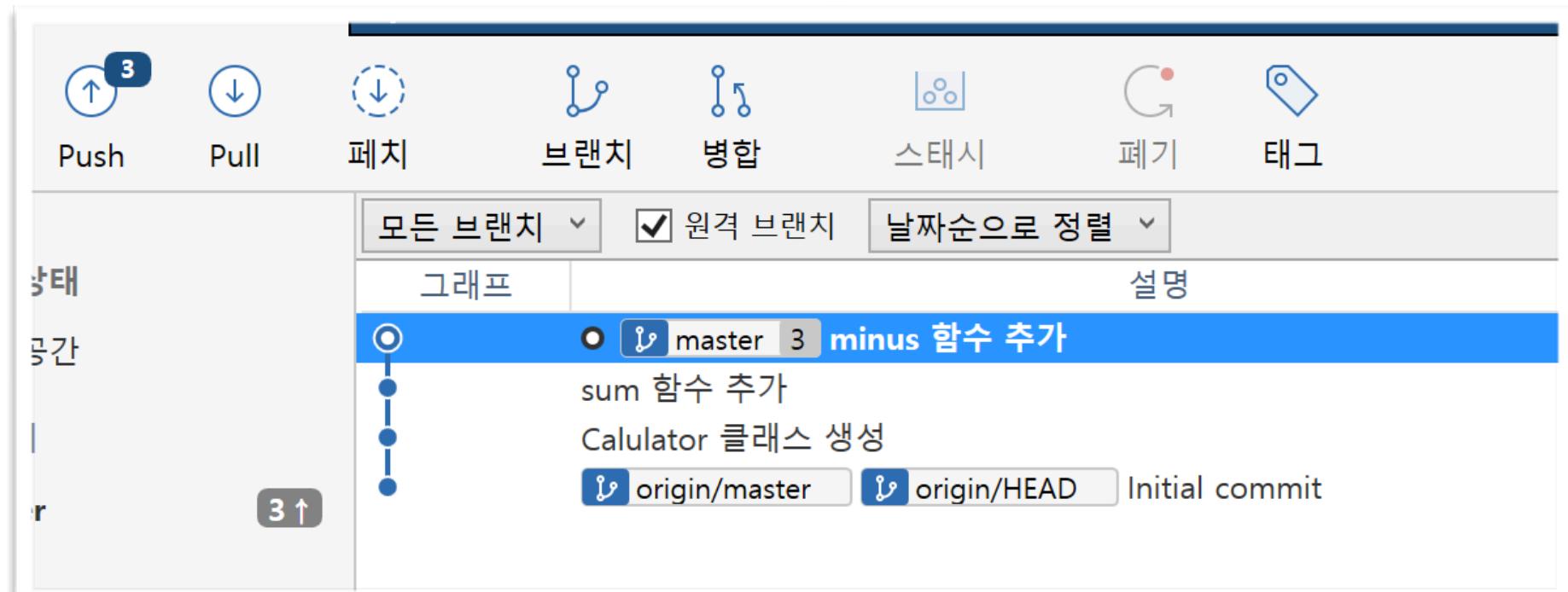
# 실습 – with SourceTree

- ▶ “sum 함수 추가” 확인



# 실습 – with SourceTree

- ▶ 같은 방식으로
- ▶ minus() 함수 추가



실습 – with SourceTree

Calculator2.java

파일로  
입력이

문서에  
제작

해주세요

# Git - 그림설명

Git → CLI

# 실습 - with CLI

## ▶ Clone

- git clone <https://github.com/jongkwang/test-github.git>

## ▶ 중복 방지를 위해 “cli” 폴더 생성

The screenshot shows a Windows Command Prompt window with the title "명령 프롬프트". The command history is as follows:

```
C:\#Users\JongKwang>cd Documents
C:\#Users\JongKwang\Documents>mkdir cli
C:\#Users\JongKwang\Documents>cd cli
C:\#Users\JongKwang\Documents\cli>git clone https://github.com/jongkwang/test-github.git
```

The last command, "git clone https://github.com/jongkwang/test-github.git", is highlighted with a red rectangle. The output of this command is displayed below:

```
Cloning into 'test-github'...
remote: Counting objects: 5, done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (5/5), done.
```

At the bottom, the prompt "C:\#Users\JongKwang\Documents\cli>" is visible.

# 실습 - with CLI

## ▶ 해당 Repository 로 이동

```
C:\ 명령 프롬프트
C:\Users\JongKwang\Documents\cli>dir
C 드라이브의 볼륨: BOOTCAMB
볼륨 일련 번호: FCDE-4CF5

C:\Users\JongKwang\Documents\cli 디렉터리

2017-08-08 오전 12:32 <DIR> .
2017-08-08 오전 12:32 <DIR> ..
2017-08-08 오전 12:32 <DIR> test-github
          0개 파일           0 바이트
          3개 디렉터리  31,707,774,976 바이트 남음

C:\Users\JongKwang\Documents\cli>cd test-github
C:\Users\JongKwang\Documents\cli\test-github>
```

# 실습 – with CLI

- ▶ “atom .” ATOM 실행

```
명령 프롬프트

::#Users#JongKwang#Documents#cli#test-github>dir
C 드라이브의 볼륨: BOOTCAMP
볼륨 일련 번호: FCDE-4CF5

C:#Users#JongKwang#Documents#cli#test-github 디렉터리

017-08-08 오전 12:32 <DIR> .
017-08-08 오전 12:32 <DIR> ..
017-08-08 오전 12:32 294 .gitignore
017-08-08 오전 12:32 1,087 LICENSE
017-08-08 오전 12:32 13 README.md
017-08-08 오전 12:32 3개 파일 1,394 바이트
017-08-08 오전 12:32 2개 디렉터리 31,707,774,976 바이트 남음

::#Users#JongKwang#Documents#cli#test-github>atom .
```

# 실습 - with CLI

The screenshot shows a code editor interface with two main panes. On the left, the 'Project' pane displays a file tree for a project named 'test-github'. The files listed are '.git', '.gitignore', 'Calculator.java', 'LICENSE', and 'README.md'. The 'Calculator.java' file is highlighted with a red box. On the right, the 'Calculator.java' code editor pane shows the following Java code:

```
1 package com.mycompany.test;  
2  
3 public class Calculator {  
4  
5     public static void main(String[] args) {  
6         }  
7     }  
8 }
```

The code editor pane also has a red border around the code area.

# 실습 - with CLI

## ▶ git status

- 현재 상황을 표시한다
- Calculator.java 파일이 Unstage 상태로 보인다

명령 프롬프트

```
C:\Users\JongKwang\Documents\cli\test-github>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)

    Calculator.java

nothing added to commit but untracked files present (use "git add" to track)

C:\Users\JongKwang\Documents\cli\test-github>
```

# 실습 - with CLI

## ▶ git add Calculator.java

- Calculator.java 파일을 Stage에 올린다
- add 명령어는 신규 파일을 추가하기도 하지만 기존파일을 Stage에 올리기도 한다

```
C:\Users\JongKwang\Documents\cli\test-github>git add Calculator.java
```

```
C:\Users\JongKwang\Documents\cli\test-github>git status
```

```
On branch master
```

```
Your branch is up-to-date with 'origin/master'.
```

```
Changes to be committed:
```

```
(use "git reset HEAD <file>..." to unstage)
```

```
new file:   Calculator.java
```

```
C:\Users\JongKwang\Documents\cli\test-github>
```

# 실습 - with CLI

## ▶ git commit

- Stage 파일을 Commit 한다

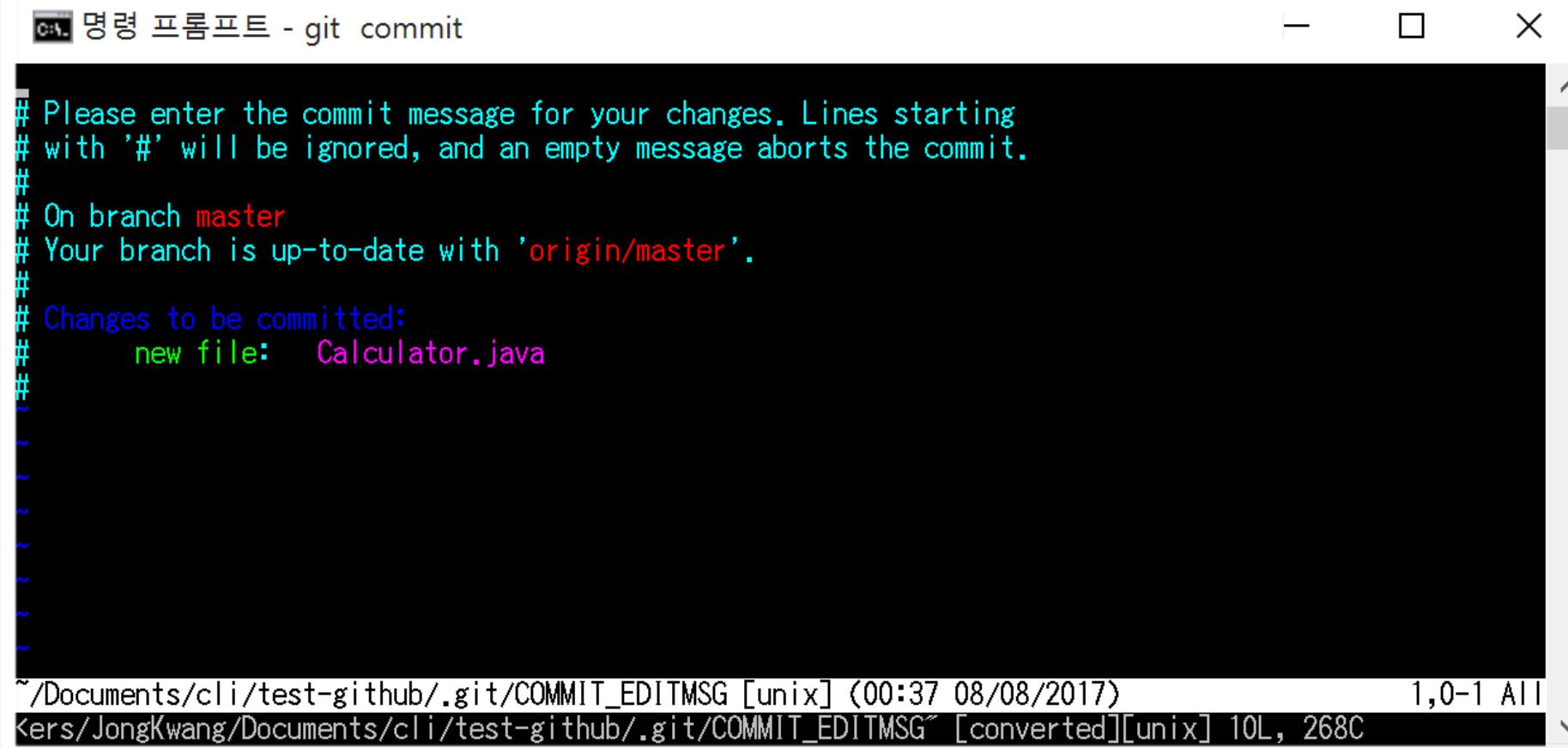
명령 프롬프트

```
C:\Users\JongKwang\Documents\cli\test-github>git commit
```

# 실습 - with CLI

## ▶ commit message 를 작성한다.

- 첫째줄이 비워져 있고, 이곳에 작성한다
- 하단 주석들은 시스템에서 git status 값을 보여준다



A screenshot of a terminal window titled "명령 프롬프트 - git commit". The window shows the following text:

```
# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# On branch master
# Your branch is up-to-date with 'origin/master'.
#
# Changes to be committed:
#   new file:   Calculator.java
#
# ~
# ~
# ~
# ~
# ~
# ~
# ~
# ~
```

The terminal window has a dark background with light-colored text. The title bar says "명령 프롬프트 - git commit". The bottom status bar shows the path "/Documents/cli/test-github/.git/COMMIT\_EDITMSG [unix]" and the date/time "(00:37 08/08/2017)". It also shows "1,0-1 All" and "Kers/JongKwang/Documents/cli/test-github/.git/COMMIT\_EDITMSG [converted][unix] 10L, 268C".

# 실습 - with CLI

## ▶ 저장은

- ESC + wq + 엔터

The screenshot shows a terminal window titled "명령 프롬프트 - git commit". The terminal displays the following text:

```
Init Calculator class
# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# On branch master
# Your branch is up-to-date with 'origin/master'.
#
# Changes to be committed:
#   new file:   Calculator.java
#
~
```

The bottom status bar indicates the path: "/Documents/cit/test-github/.git/COMMIT\_EDITMSG[+]" and the time: "(00:37 08/08/2017)". The bottom left corner shows the command "wq\_". A red box highlights the command line area.

# 실습 – with CLI

## ▶ 커밋 성공

- Commit message 를 남기지 않으면 커밋이 되지 않는다 (Default)
- 커밋이 성공하여 “1be5cfc” 라는 Hash 값이 생겼다

```
명령 프롬프트

C:\Users\JongKwang\Documents\cli\test-github>git commit
[master 1be5cfc] Init Calculator class
 1 file changed, 7 insertions(+)
 create mode 100644 Calculator.java

C:\Users\JongKwang\Documents\cli\test-github>
```

# 실습 – with CLI

## ▶ git status

- 모두 반영되어 깨끗하다

```
C:\Users\JongKwang\Documents\cli\test-github>git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean

C:\Users\JongKwang\Documents\cli\test-github>
```

# 실습 - with CLI

## ▶ git log

- 커밋 로그를 표시한다

C:\ 명령 프롬프트

```
C:\Users\JongKwang\Documents\cli\test-github>git log
commit 1be5cfcba61a7c4d8fbe7e090c38d9cbe1ebe505 (HEAD -> master)
Author: JongKwang <kim@jongkwang.com>
Date:   Tue Aug 8 00:37:00 2017 +0900

    Init Calculator class

commit bdc66b83e692c400250faec145377e2d2fe5357f (origin/master, origin/HEAD)
Author: JongKwang <kim@jongkwang.com>
Date:   Mon Aug 7 23:30:00 2017 +0900

    Initial commit

C:\Users\JongKwang\Documents\cli\test-github>
```

# 실습 - with CLI

- ▶ sum() 메소드 작성

```
package com.mycompany.test;

public class Calculator {

    public static void main(String[] args) {
        int a = 4;
        int b = 2;
        System.out.println( sum(a,b) );
    }

    private static int sum(int a, int b) {
        return a + b;
    }
}
```

# 실습 - with CLI

- ▶ git status
- ▶ git add Calculator.java

c:\ 명령 프롬프트

```
C:\Users\JongKwang\Documents\cli\test-github>git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   Calculator.java

no changes added to commit (use "git add" and/or "git commit -a")

C:\Users\JongKwang\Documents\cli\test-github>git add Calculator.java

C:\Users\JongKwang\Documents\cli\test-github>
```

# 실습 – with CLI

## ▶ git reset Calculator.java

- Stage 의 파일을 Unstage 로 옮긴다

 명령 프롬프트

```
C:\Users\JongKwang\Documents\cli\test-github>git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    modified:   Calculator.java

C:\Users\JongKwang\Documents\cli\test-github>git reset Calculator.java
Unstaged changes after reset:
M       Calculator.java

C:\Users\JongKwang\Documents\cli\test-github>
```

# 실습 - with CLI

- ▶ git reset 되어 Unstage 된 자바 파일

## 명령 프롬프트

```
C:\Users\JongKwang\Documents\cli\test-github>git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

    modified:   Calculator.java

no changes added to commit (use "git add" and/or "git commit -a")

C:\Users\JongKwang\Documents\cli\test-github>
```

# 실습 – with CLI

- ▶ commit message 작성하지 않으면
  - 커밋이 되지 않는다

## 명령 프롬프트

```
C:\Users\JongKwang\Documents\cli\test-github>git add Calculator.java
```

```
C:\Users\JongKwang\Documents\cli\test-github>git commit  
Aborting commit due to empty commit message.
```

```
C:\Users\JongKwang\Documents\cli\test-github>
```

# 실습 - with CLI

## ▶ 커밋 메세지 작성

```
C:\ 명령 프롬프트 - git commit
Add sum() method
# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# On branch master
# Your branch is ahead of 'origin/master' by 1 commit.
#   (use "git push" to publish your local commits)
#
# Changes to be committed:
#       modified:   Calculator.java
#
~  
~  
~  
~  
~  
~  
~  
~  
~/Documents/cli/test-github/.git/COMMIT_EDITMSG[+] [unix] (01:11 08/08/2018)
:wq
```

실습 – with SourceTree

Calculator2.java

파일로  
입력이

문서에  
제작

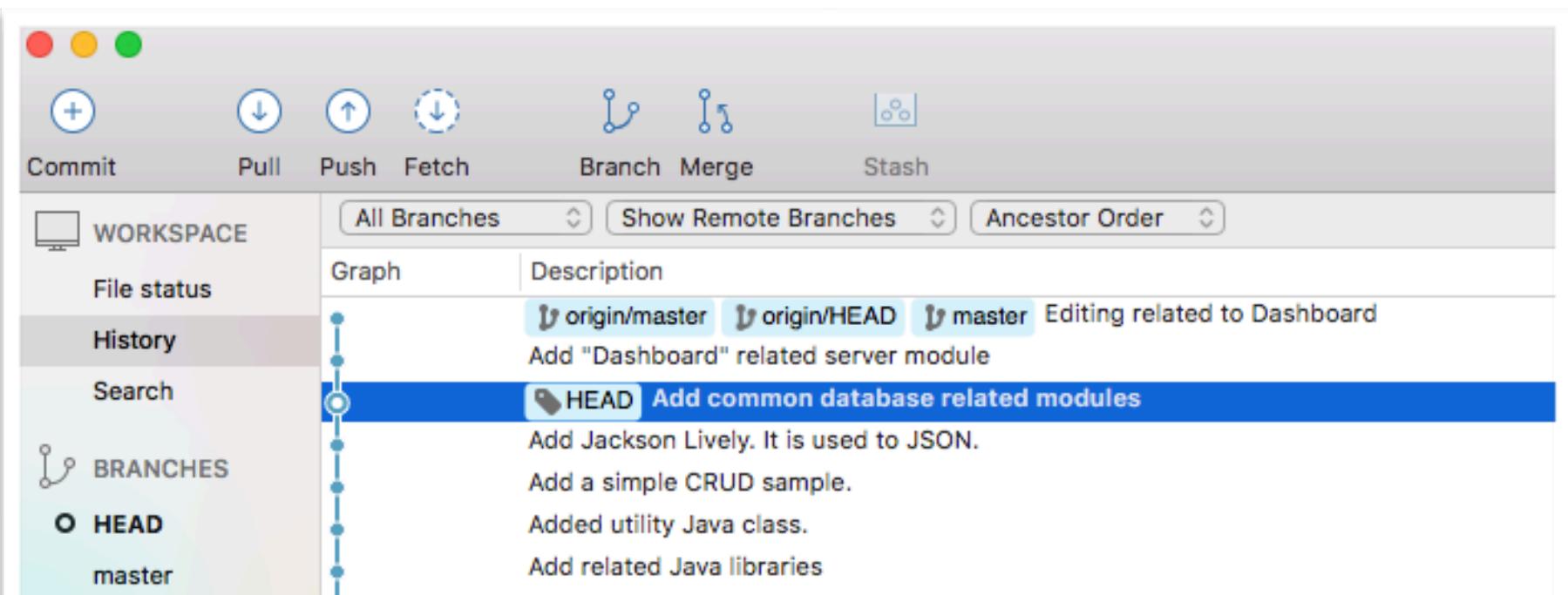
해주세요

Git - 그림설명

Pointer

# Git – Pointer

- ▶ HEAD tag
  - 작업폴더가 바라보고 있는 버전을 표시
- ▶ Master branch
  - 가장 기본이 되는 Brach
  - Master branch 는 Git 에서 기본으로 생성해준다



# Git – Pointer

## ▶ origin

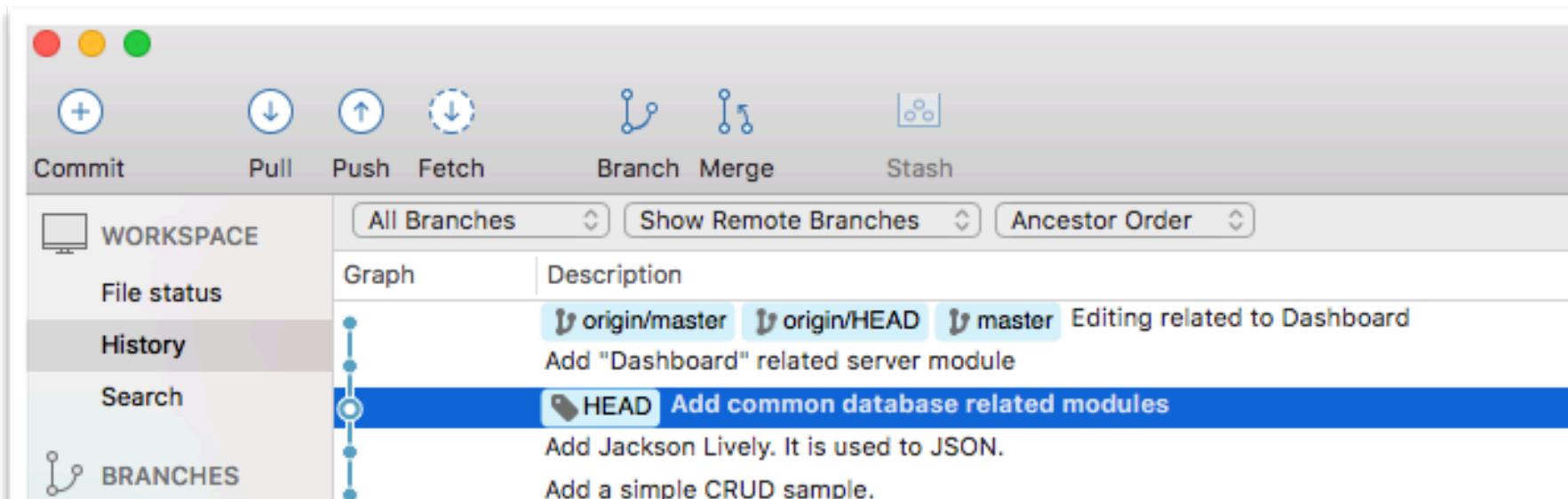
- 원격 Repository 를 뜻하는 Alias
- Alias 이름을 변경 할 수 있다

## ▶ origin/master

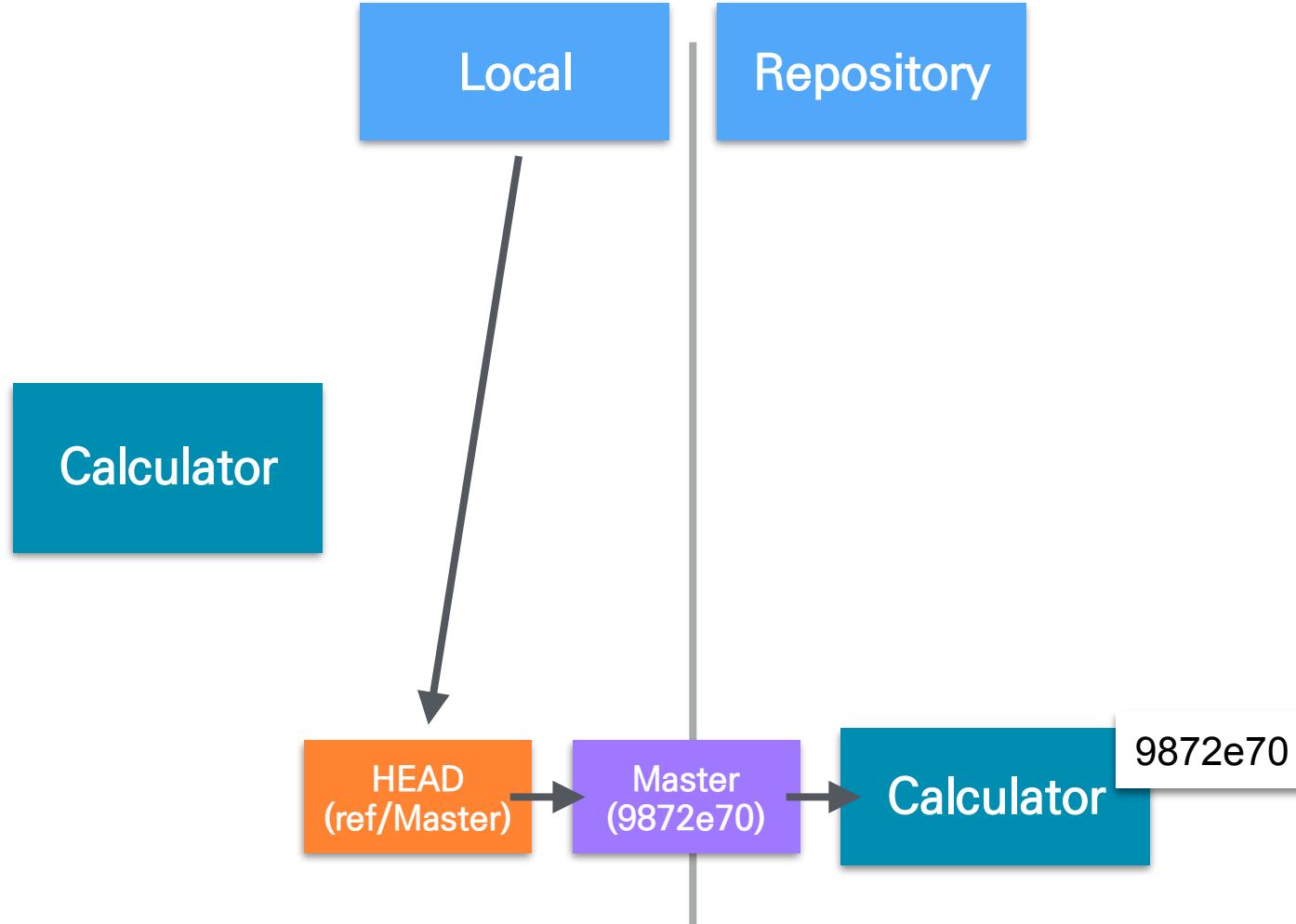
- 원격 Repository 의 master branch

## ▶ origin/HEAD

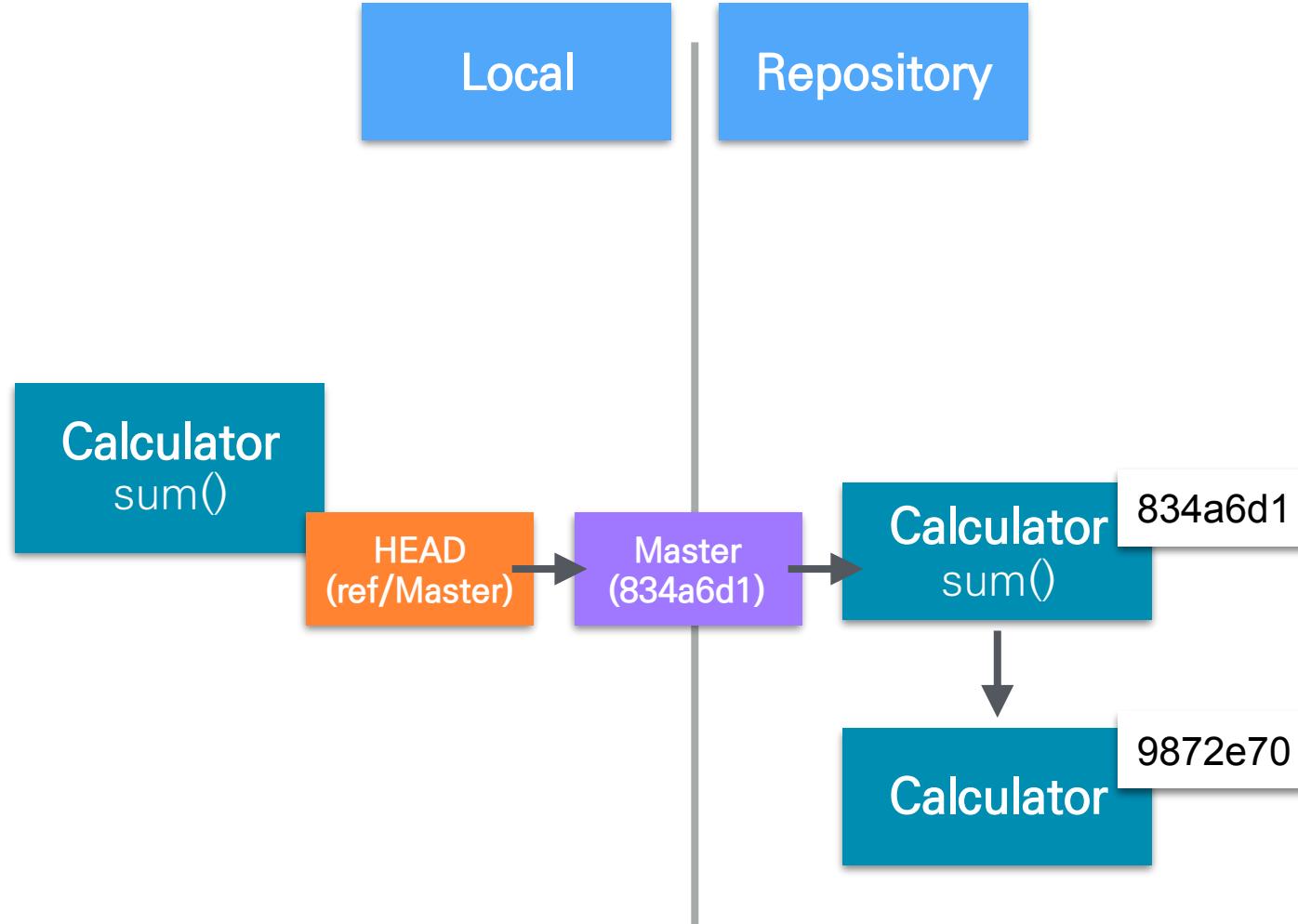
- 원격 Repository 가 바라보는 버전



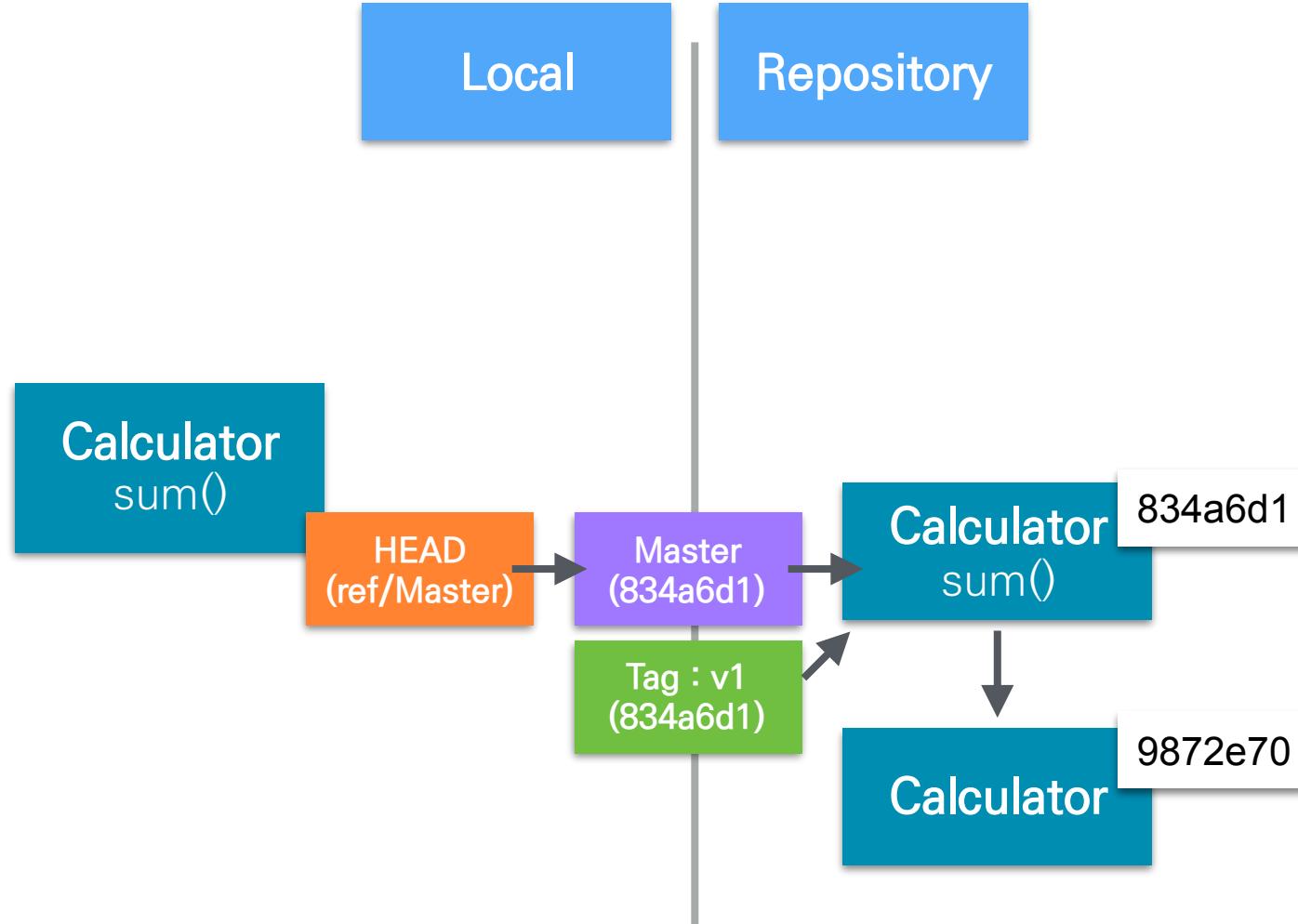
# Git - 그림 설명



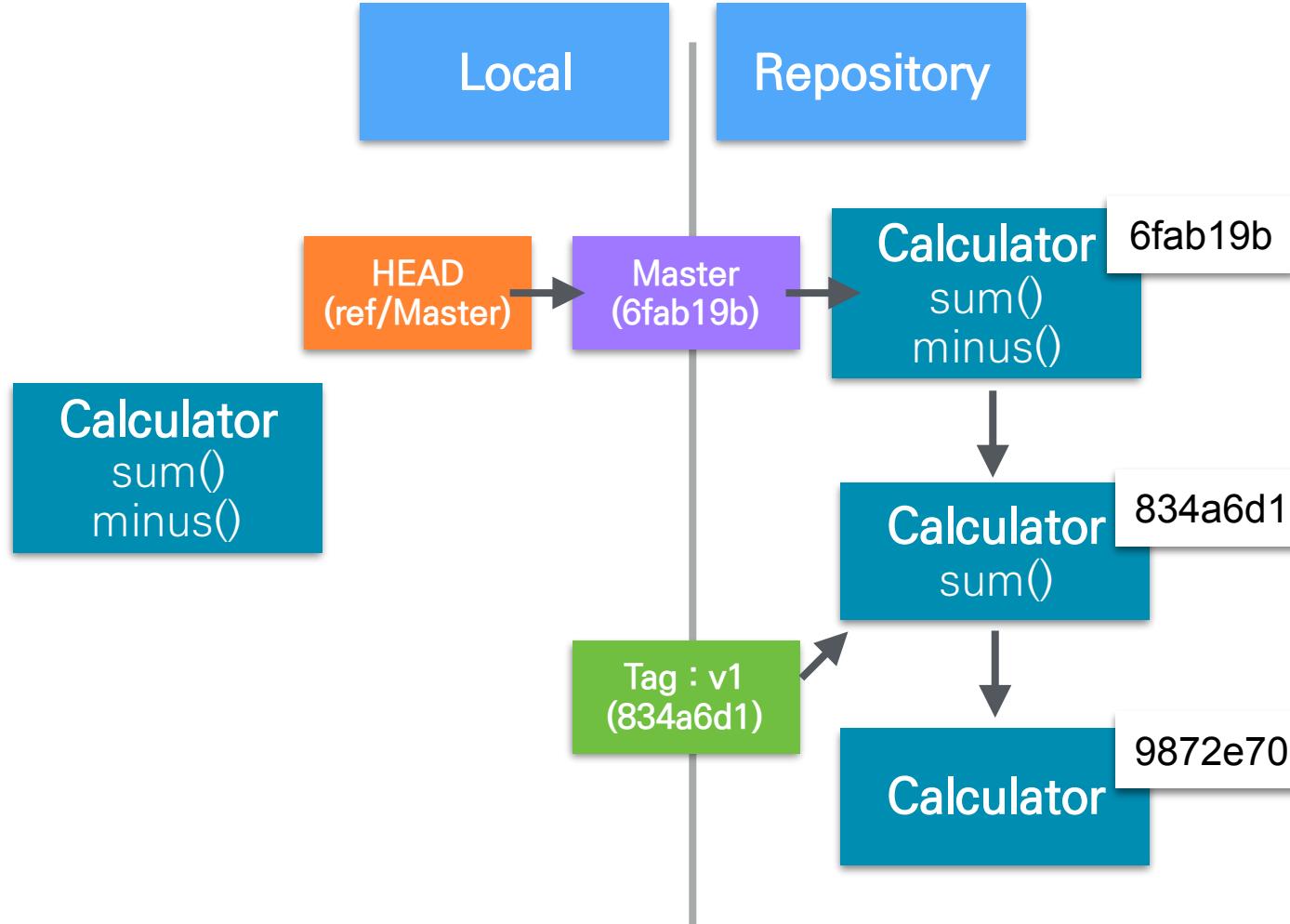
# Git – 그림 설명



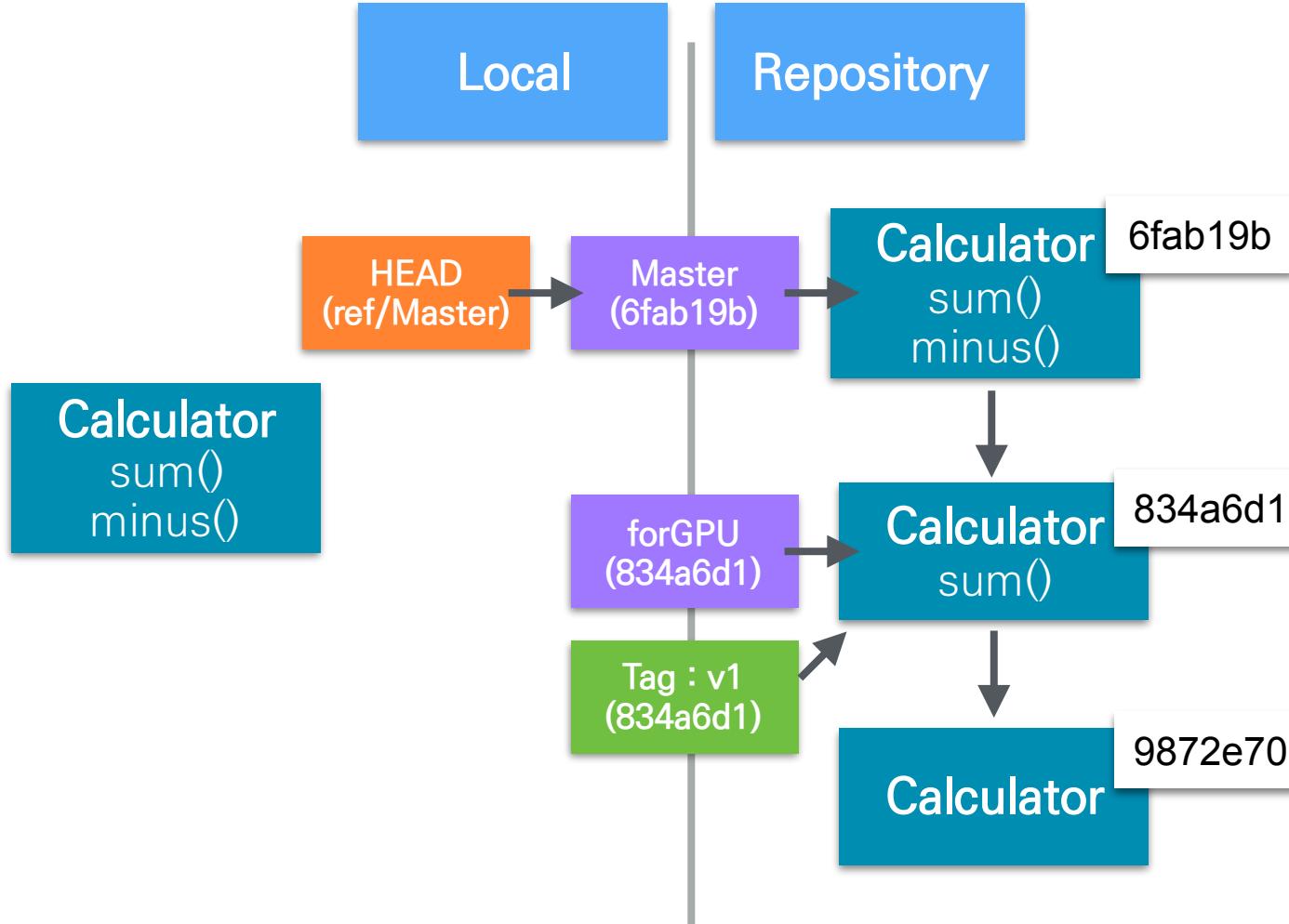
# Git – 그림 설명



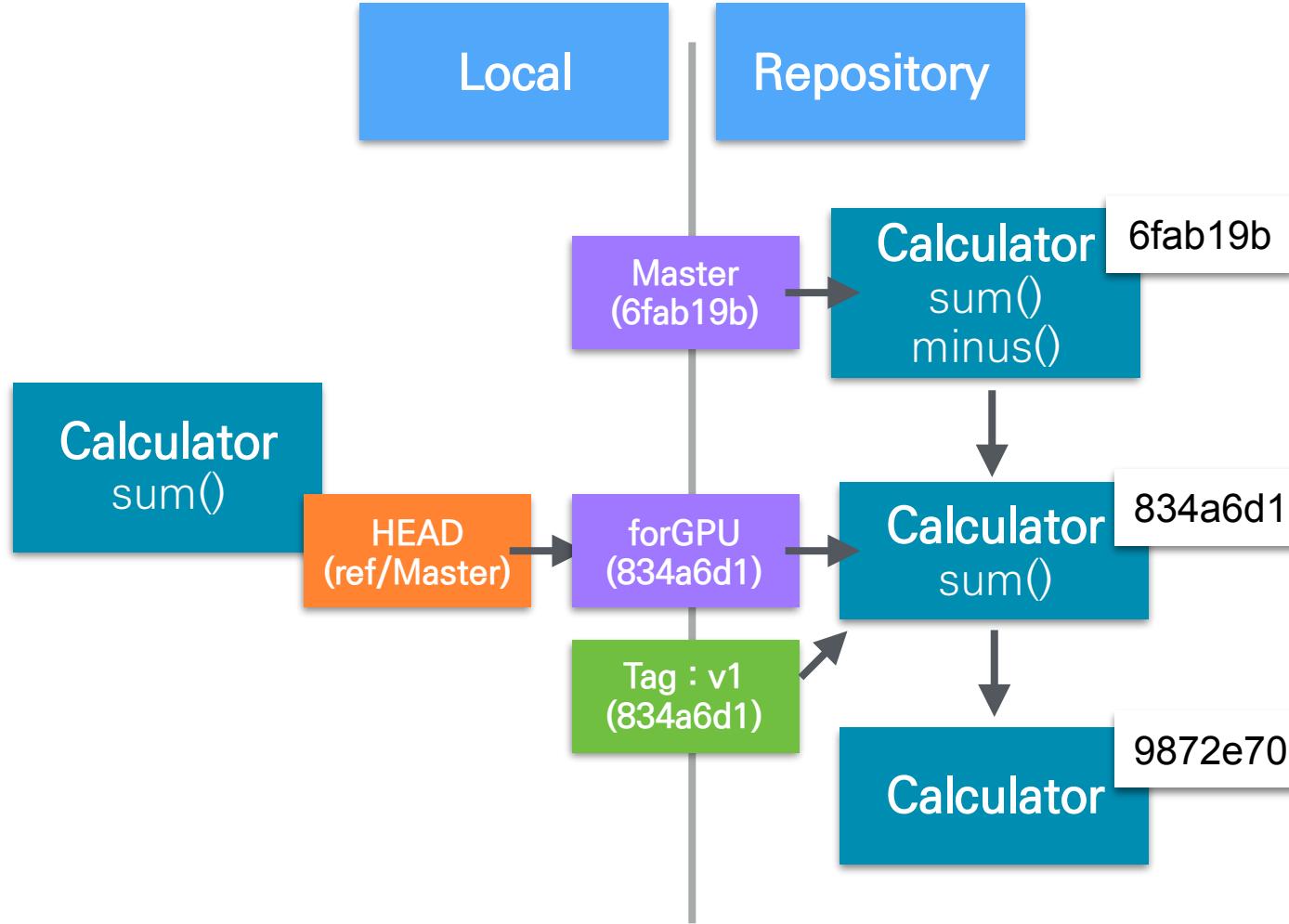
# Git – 그림 설명



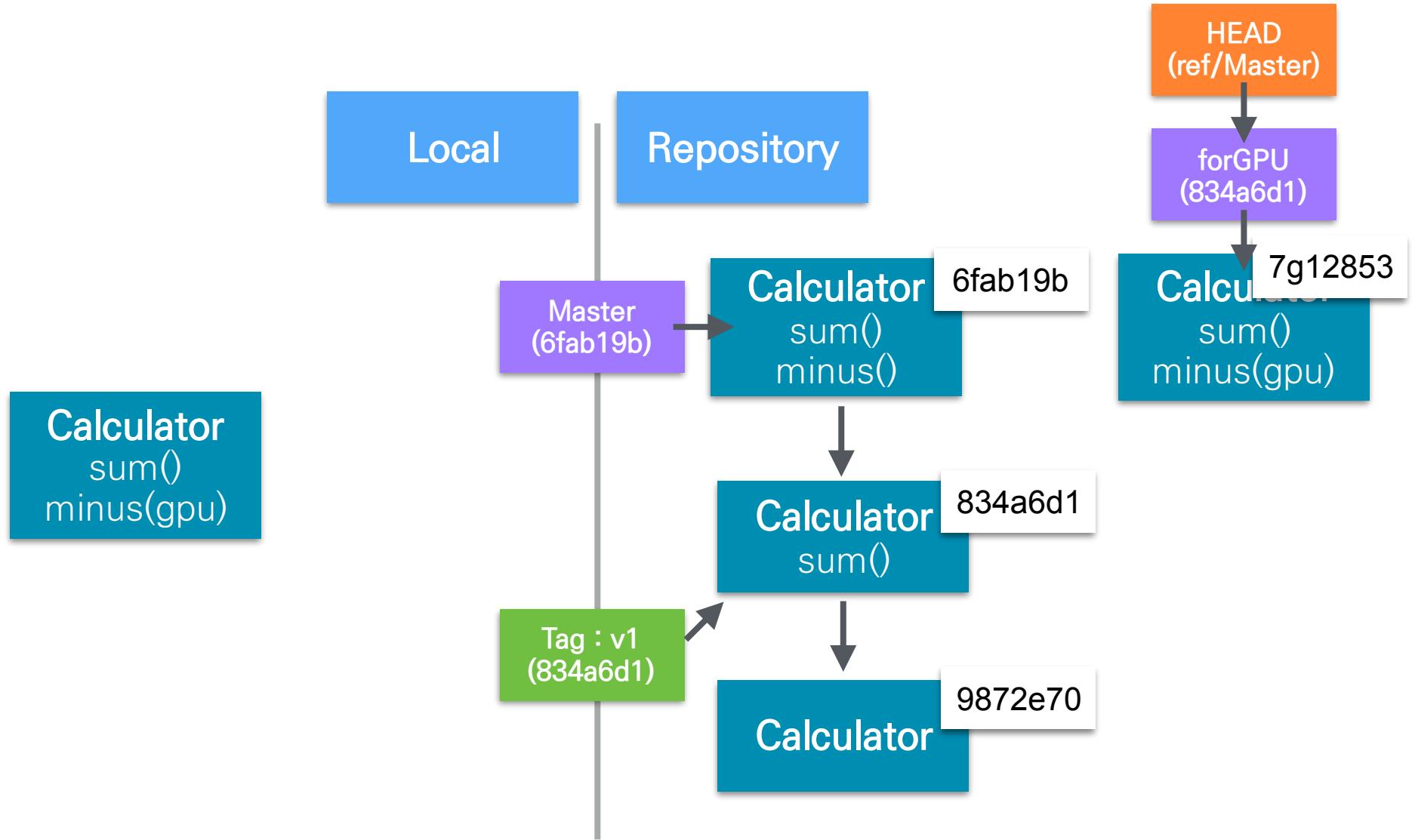
# Git – 그림 설명



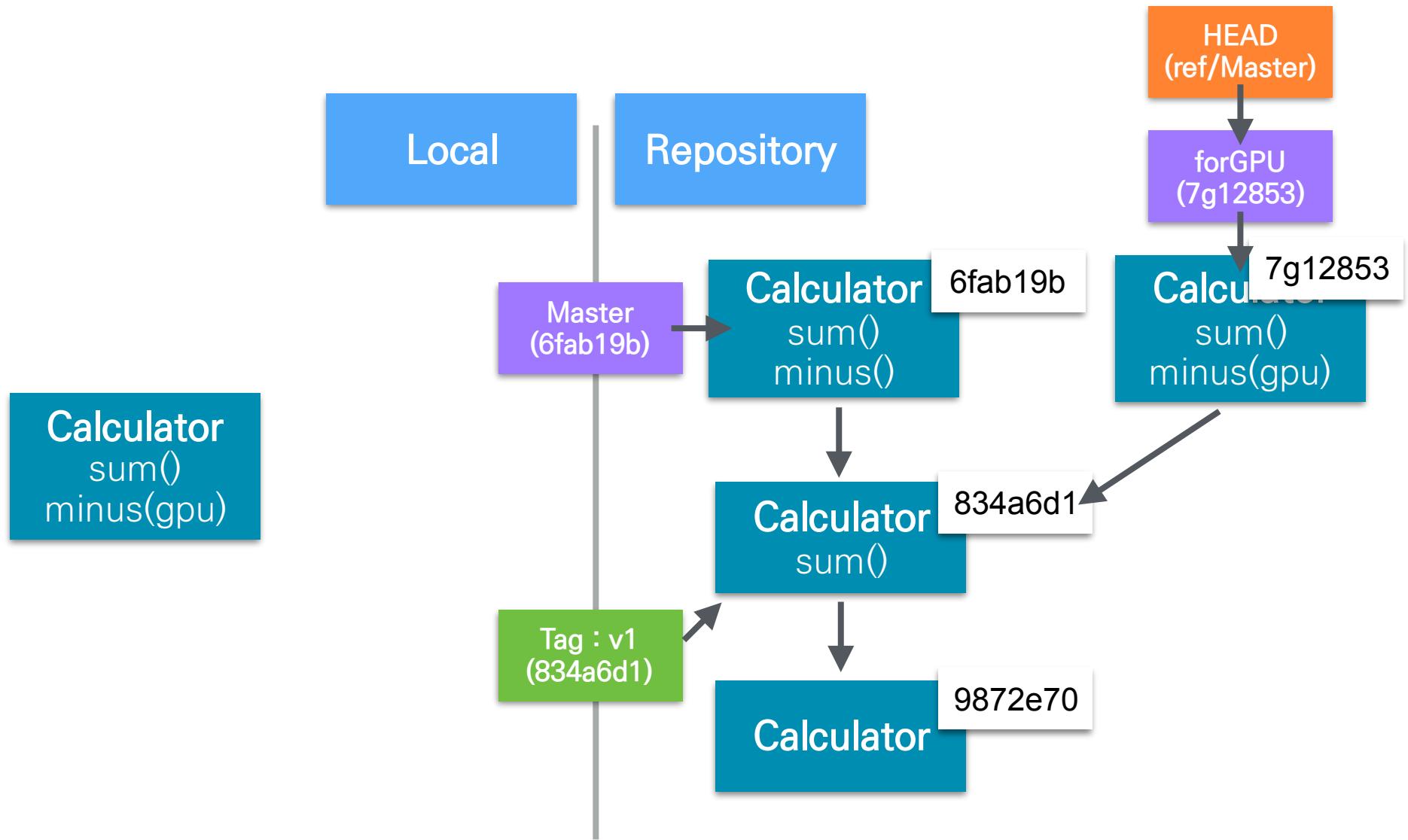
# Git – 그림 설명



# Git – 그림 설명



# Git – 그림 설명



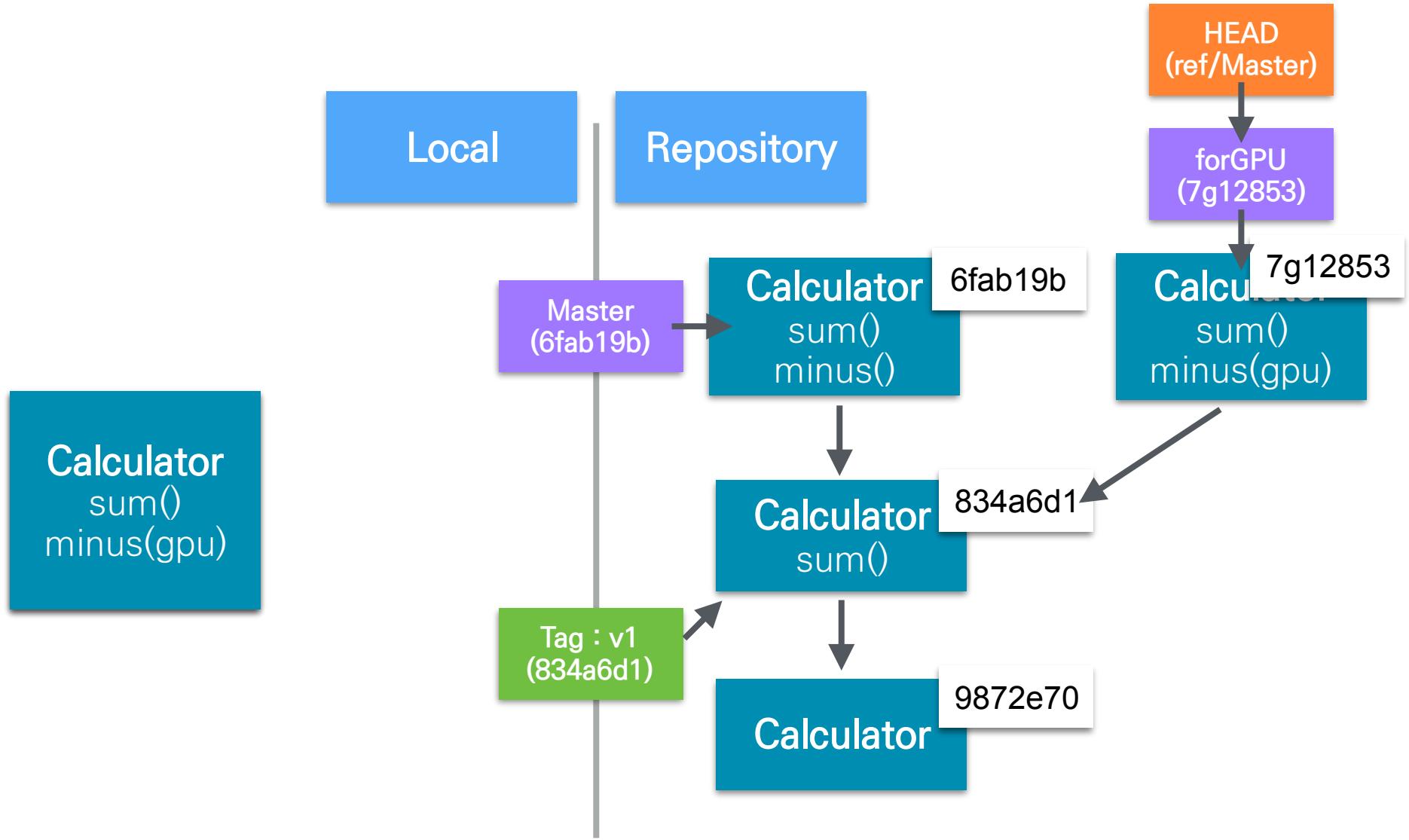
# Git - 그림설명

병합

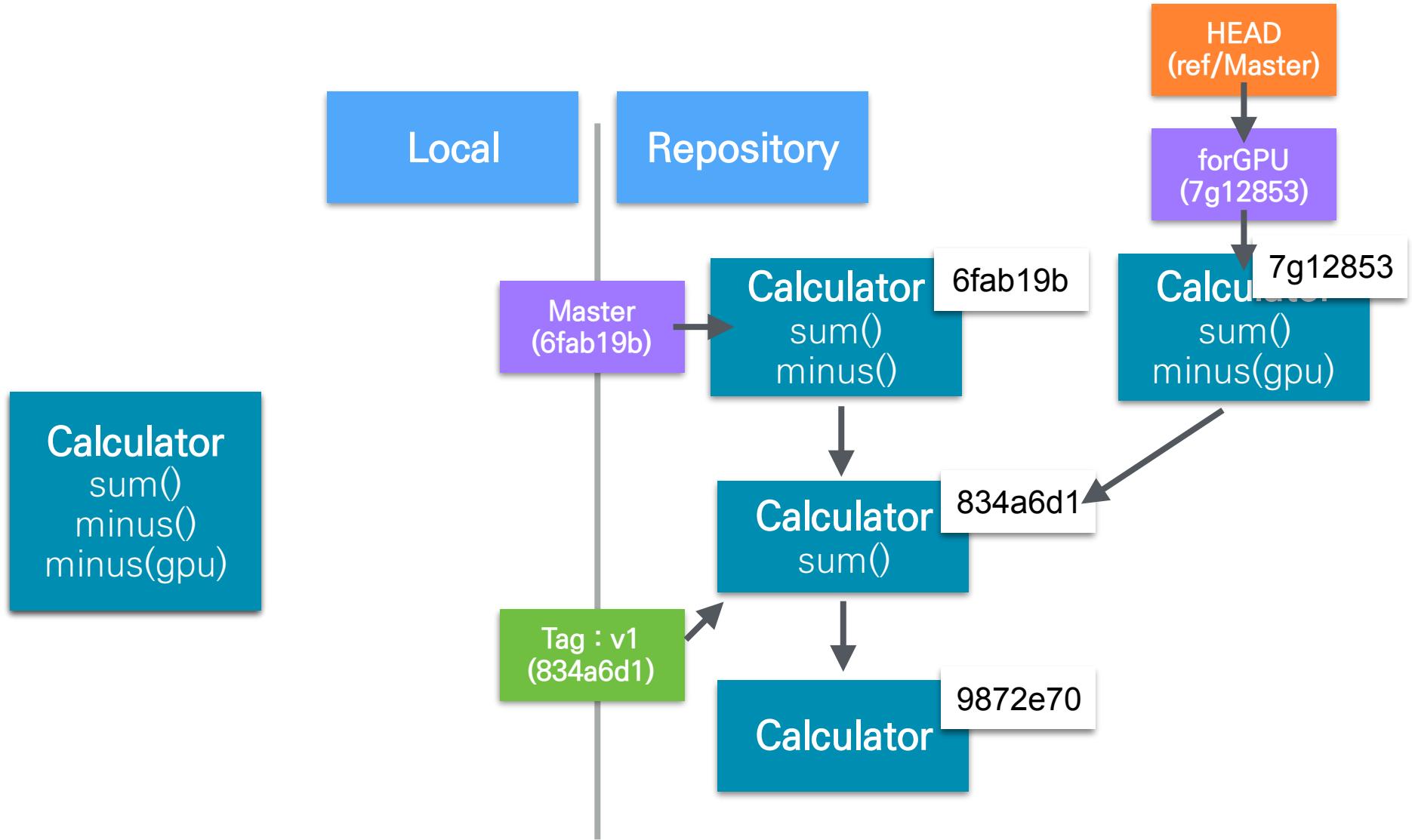


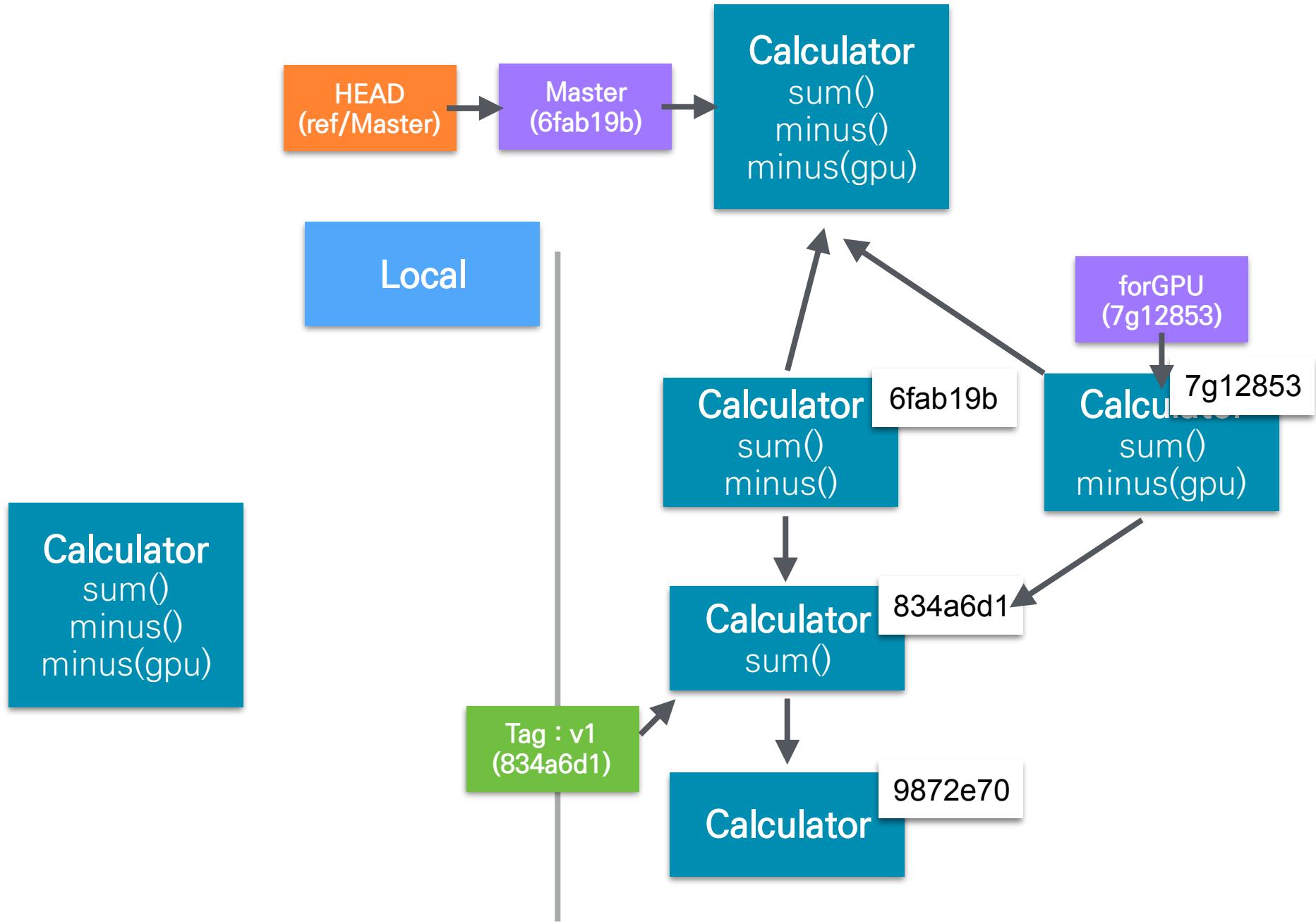
Merge

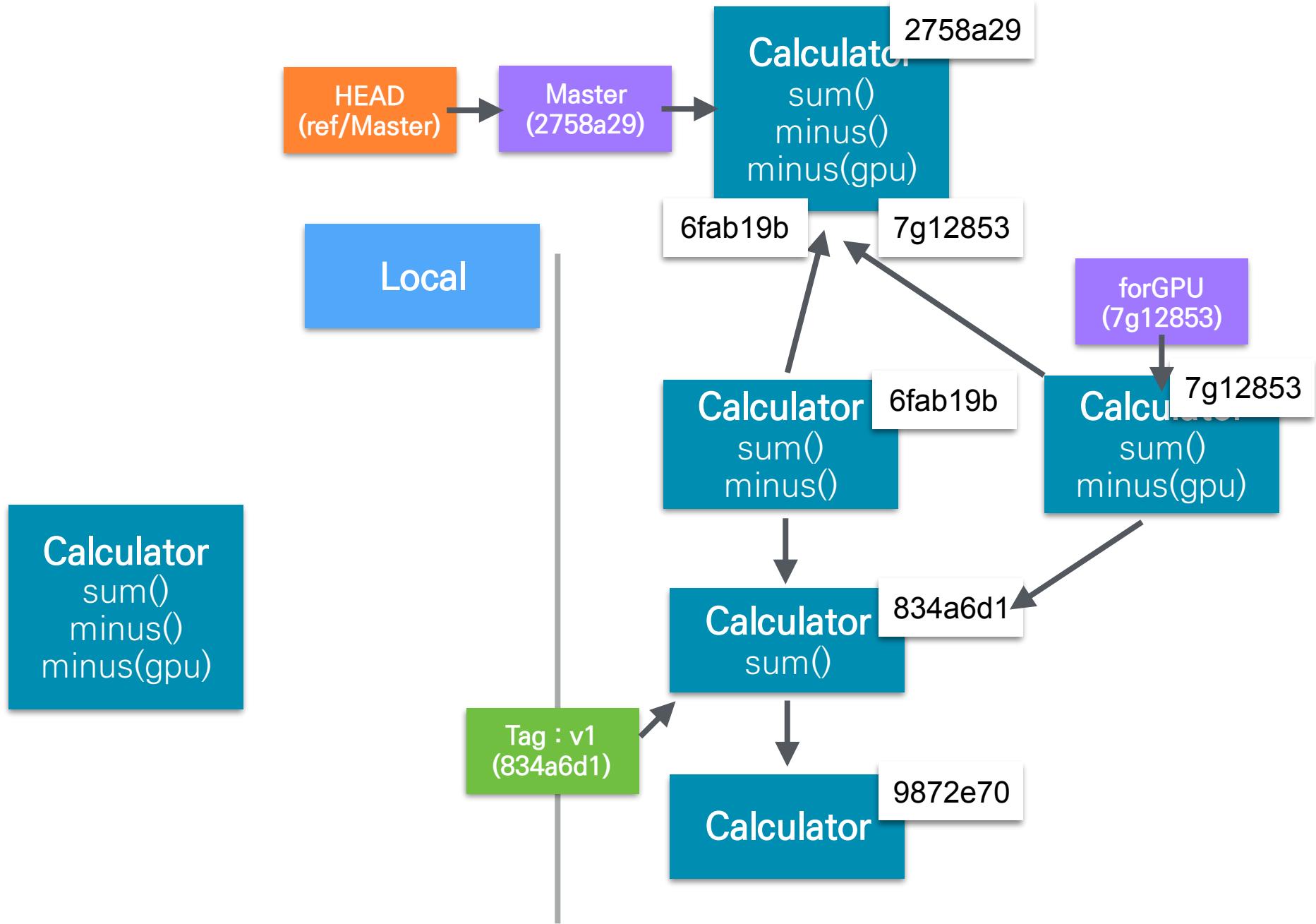
# Git – 그림 설명



# Git – 그림 설명

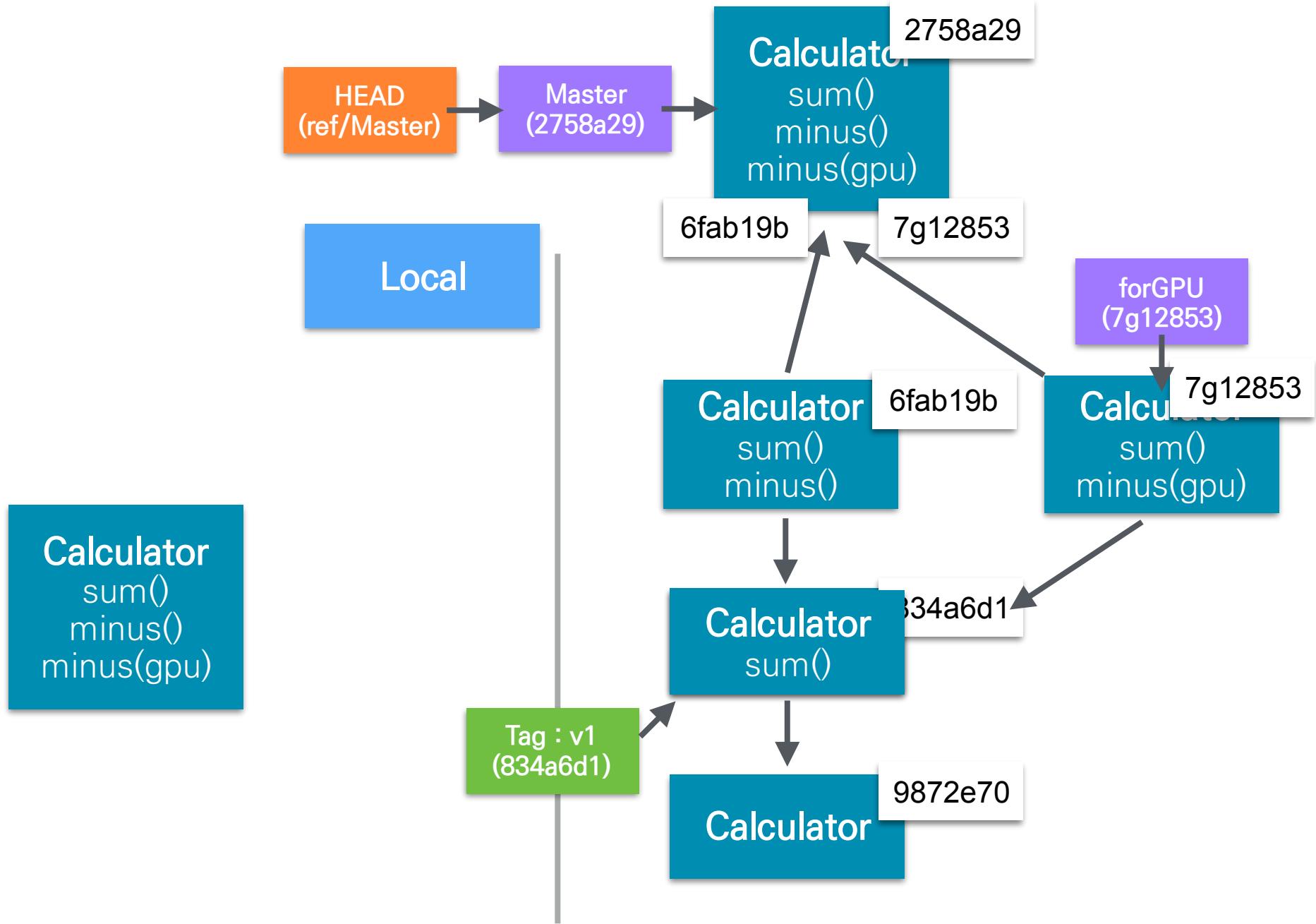


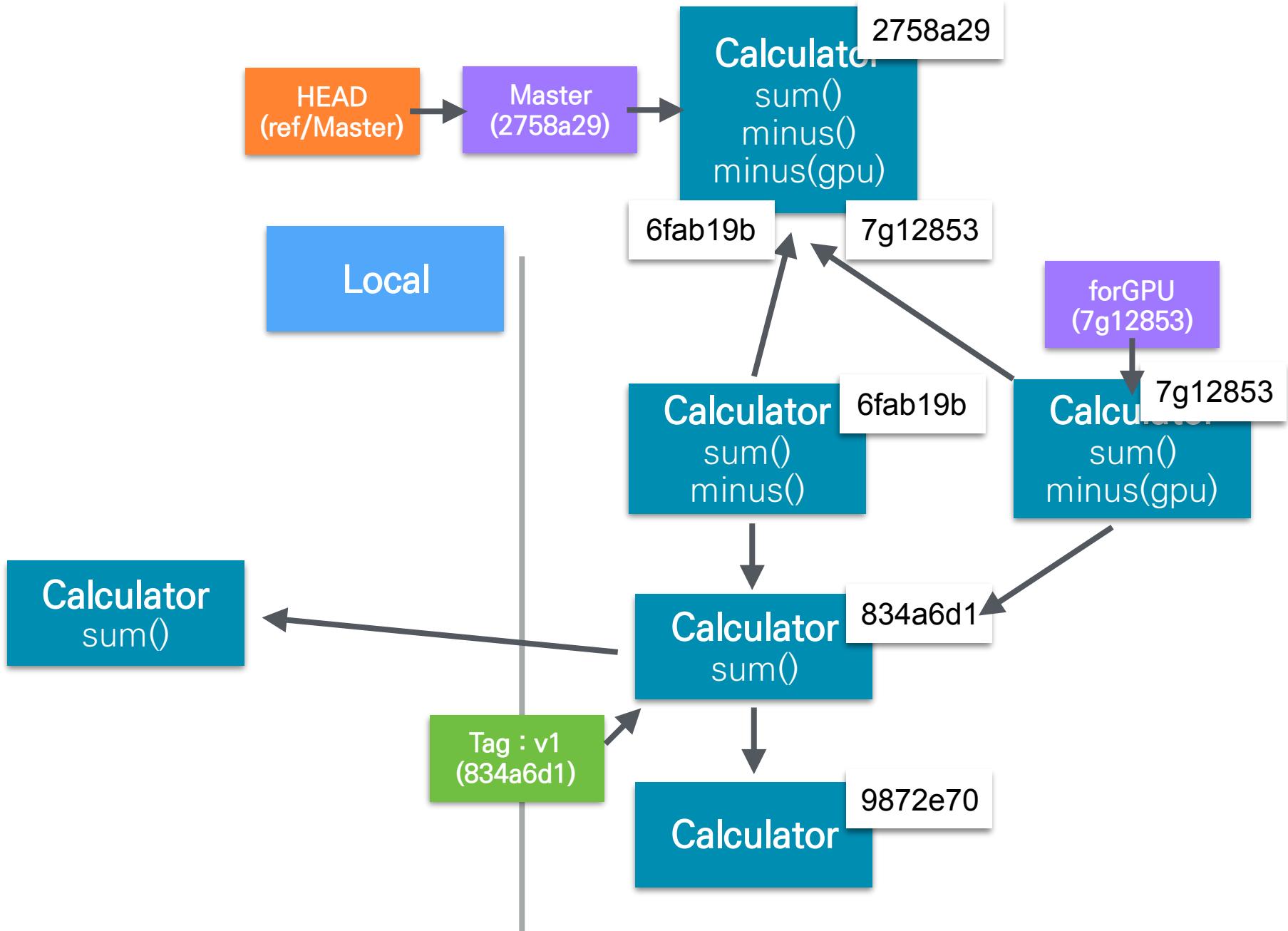


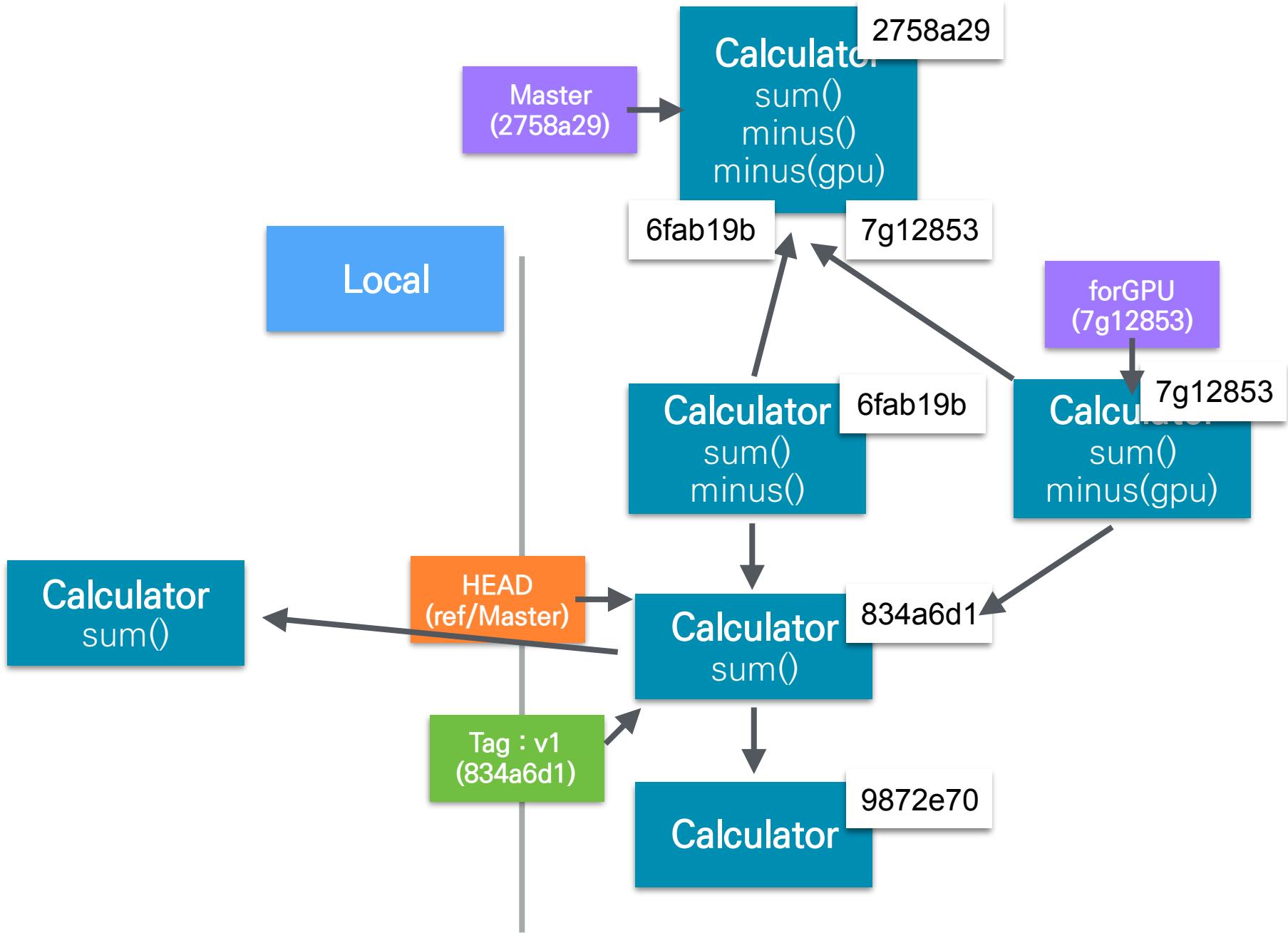


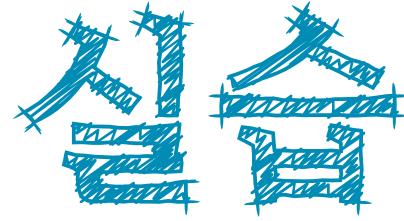
Git - 그림설명

Checkout









Git - 그림설명

# Reference

# Reference

## ▶ Git

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

## ▶ 이종은님 Github

- [jsdev.kr](http://jsdev.kr) 운영진
- Titanium 커뮤니티
- <https://github.com/yomybaby/gitlecture>

# Git / Github

깃허브