

Introduction to Git

Database Systems
DataLab, CS, NTHU
Spring, 2025

Outline

- Version control system
- Git basics
- Git branch
- Remote repository

Outline

- Version control system
- Git basics
- Git branch
- Remote repository

Why Version Control ?

Version Control System

- Store the projects, keep your **revision history**
- **Synchronization** between modifications made by different developers

Outline

- Version control system
- **Git basics**
- Git branch
- Remote repository

Git

- Git is a version control system which is
 - Fast
 - Easy to use
 - Distributed
 - Able to handle large project
(ex. Linux Kernel 27.8 million lines)
- A git repository is a mini database that tracks your files

Installation

- Please check this link
 - <http://git-scm.com/book/en/Getting-Started-Installing-Git>

Configuration

- Modify `~/.gitconfig`
- Or, type in following commands

```
git config --global user.name "your name"
```

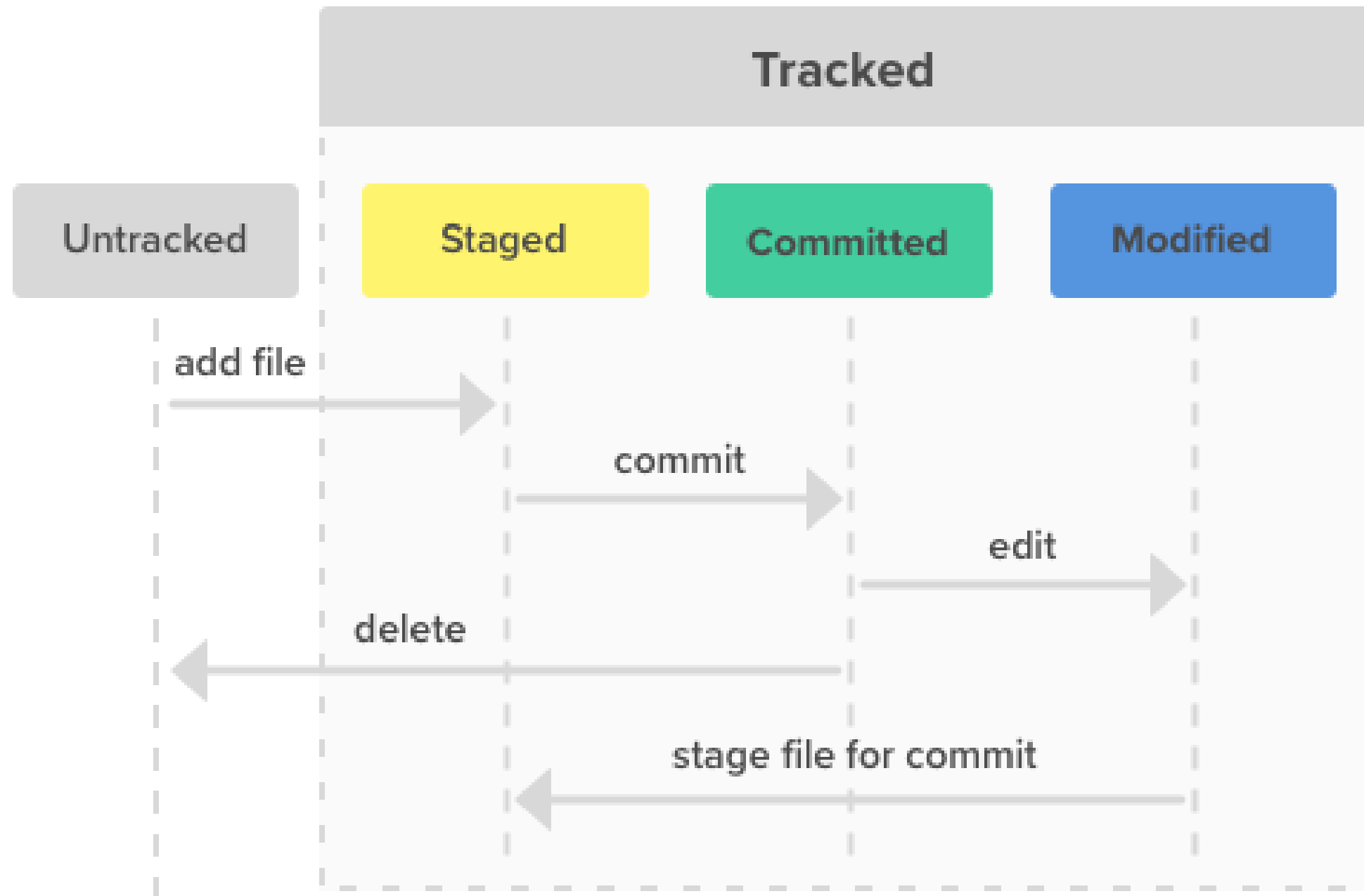
```
git config --global user.email "your@email.com"
```

* For more information, please refer this [link](#).

Creating a new Repository

- Two ways to create a repository
 - Initializing a Repository in an Existing Directory
`git init`
 - Cloning an Existing Repository
 - We will talk about it later
- The repository information will be stored in the `.git` directory

Committing A Version



Committing A Version

- Staging (adding) a file

```
git add [file name]
```

- Staging all files in the current directory

```
git add -A
```

- Committing

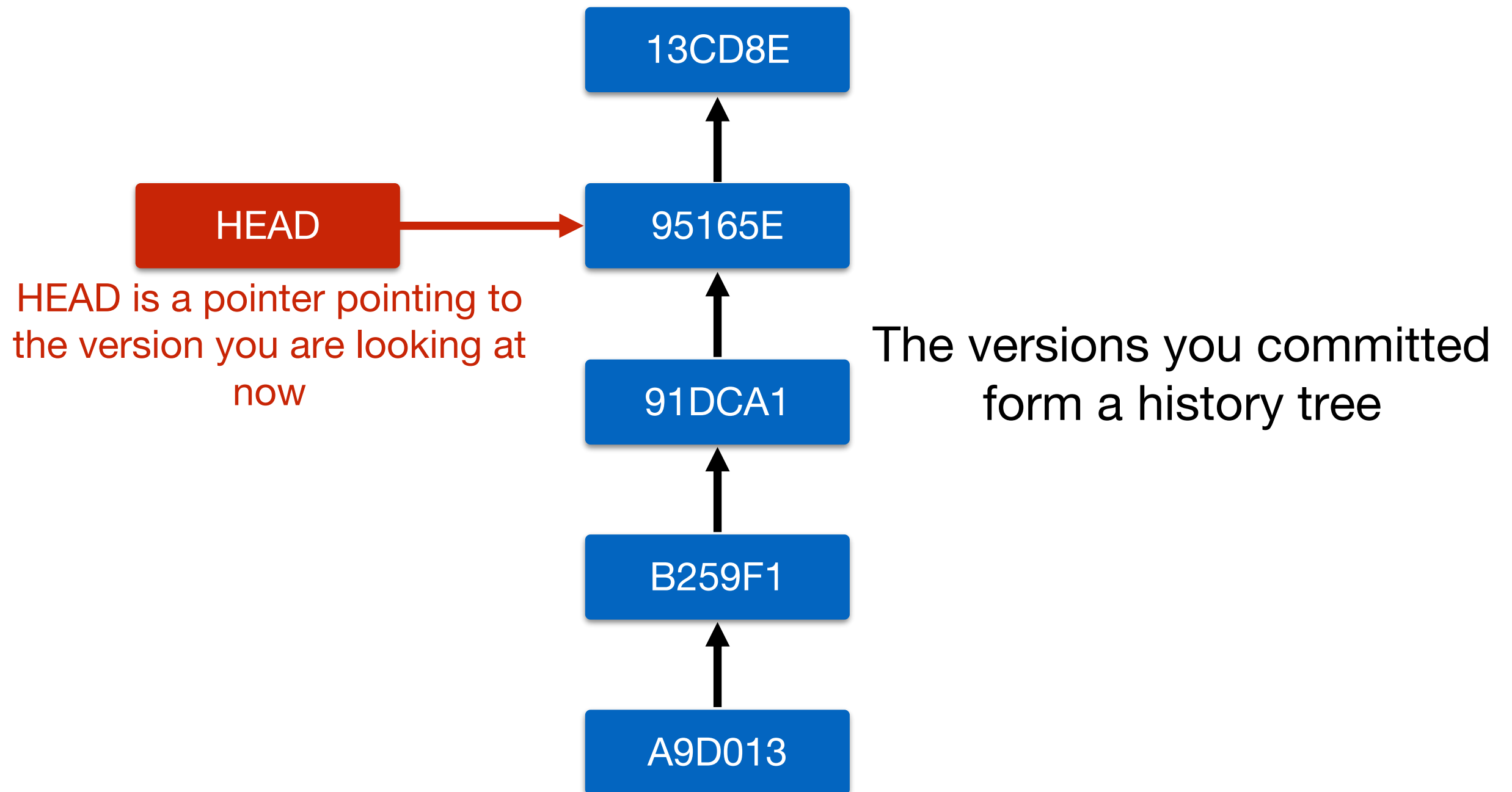
```
git commit -m "[message]"
```

Status

- Checking the current status and the current branch

```
git status
```

A History Tree



Logs

- Listing the log

```
git log
```

- Listing each log in one line

```
git log --oneline
```

```
user@DESKTOP-UUF072K MINGW64 ~/Desktop/test/practice-submission (master)
```

```
$ git log
```

```
commit 25f75ca42a96a900a6f87994a6b4fd70583d5561 (HEAD -> master, origin/master, origin/HEAD)
```

```
Author: kkeevin123456 <kkeevin123456@gmail.com>
```

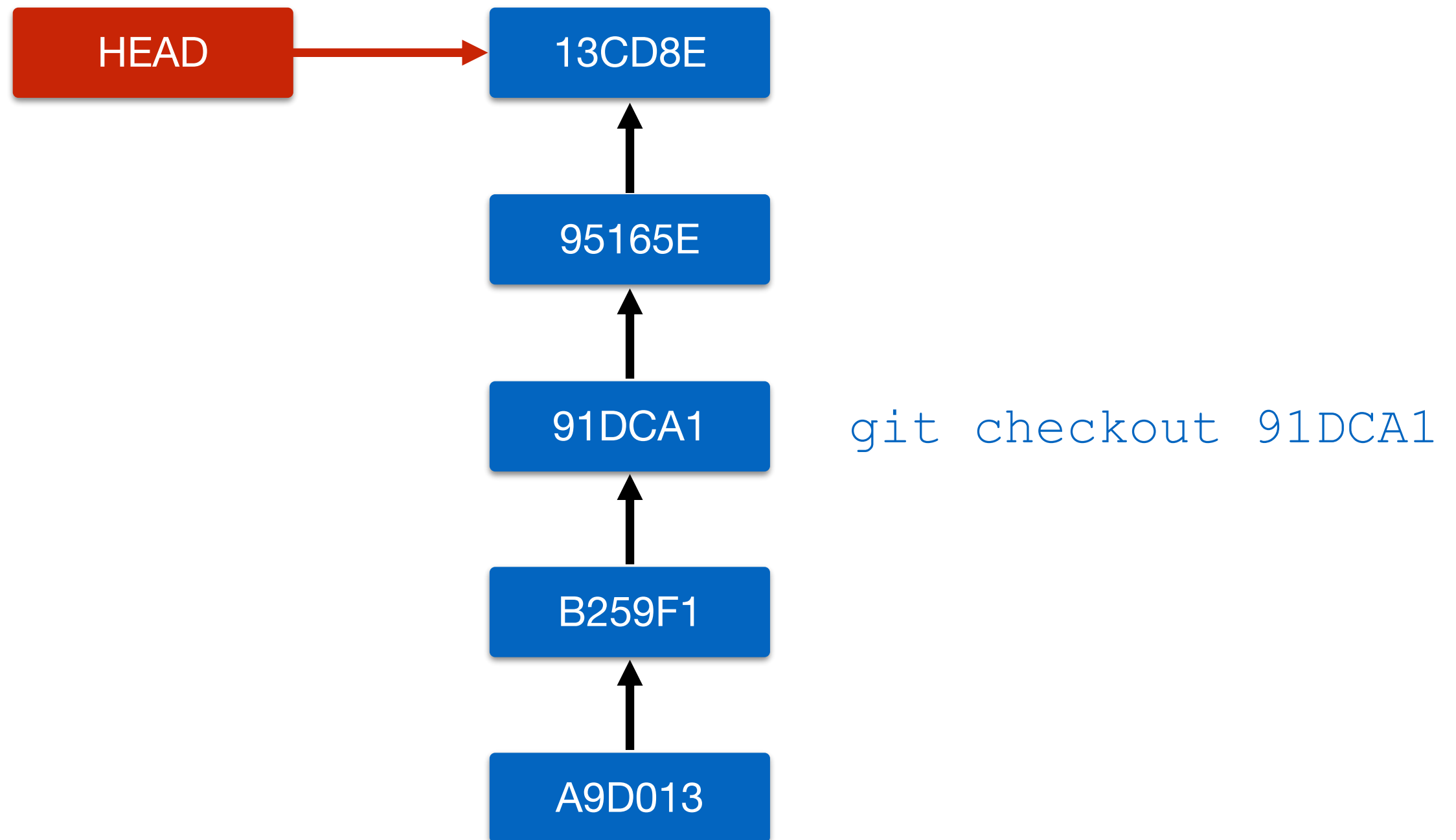
```
Date: Sat Mar 6 11:02:23 2021 +0800
```

```
add branch
```

```
commit d7533bf6016e3faf834e9e38f8dfd552960728d1 (origin/p123786579, origin/109065541, origin/109062608, origin/107062301, origin/107062261, origin/107062233, origin/107062231, origin/107062230, origin/107062228, origin/107062123, origin/107062115, origin/107060023, origin/107060015, origin/107060008, origin/107060007, origin/107034054, origin/107000115, origin/107000105, origin/106072108, origin/106070034, origin/106070033, origin/106070007, origin/106062133, origin/106062110, origin/106062109, origin/106033233, origin/106021225, origin/106012042, origin/104061701, origin/101000030, p123786579, 109065541, 109062608, 108065702, 108003809, 107062381, 107062325, 107062301, 107062261, 107062233, 107062231, 107062230, 107062228, 107062123, 107062115, 107060023, 107060015, 107060008, 107060007, 107034054, 106070033, 106070007, 106062315, 106062306, 106062214, 106062205, 106062142, 106062137, 106062133, 106062110, 104061701, 101000030)
```

```
Author: chanchishen <csch@datalab.cs.nthu.edu.tw>
```

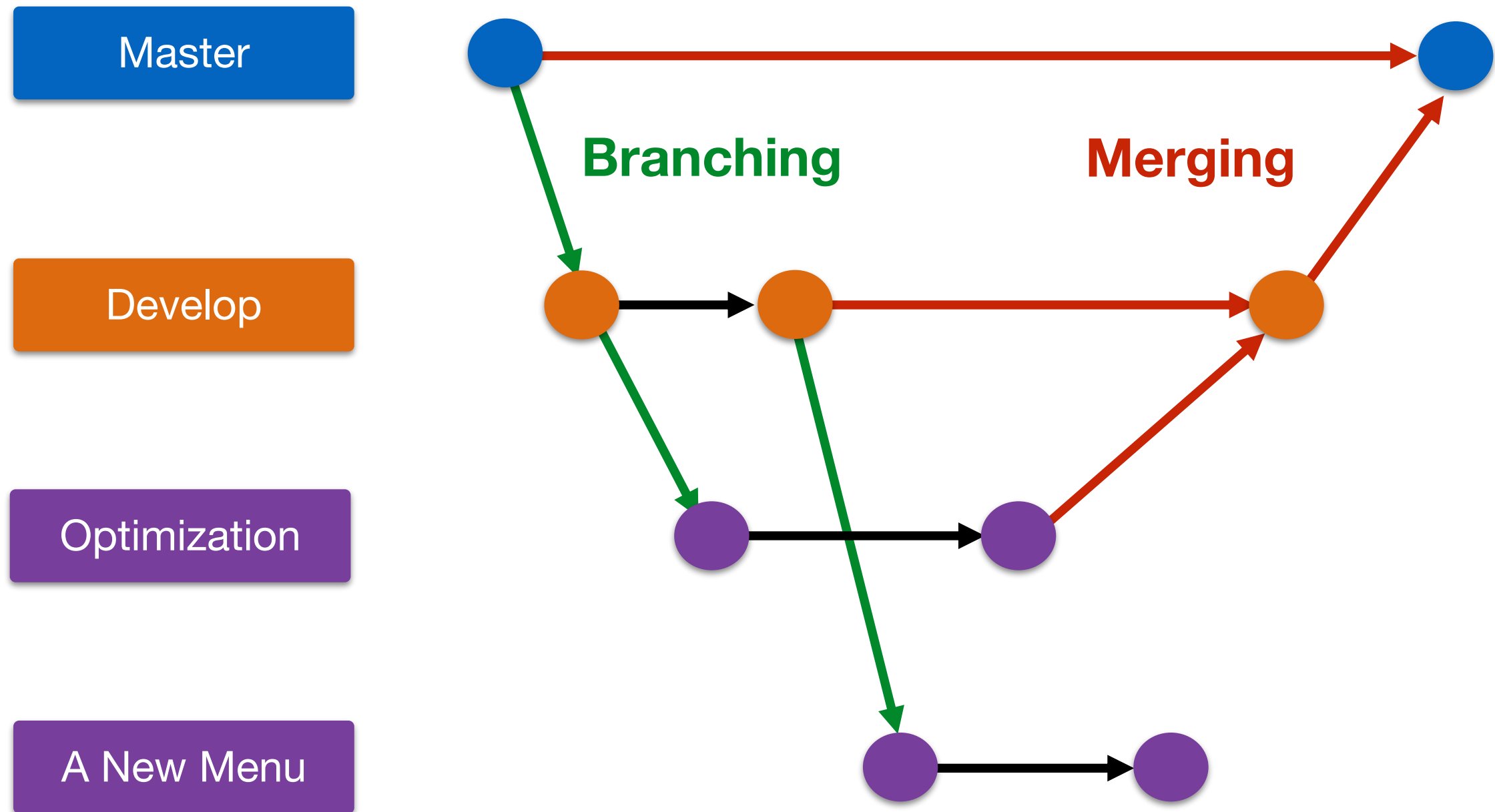
Checking Out A Version



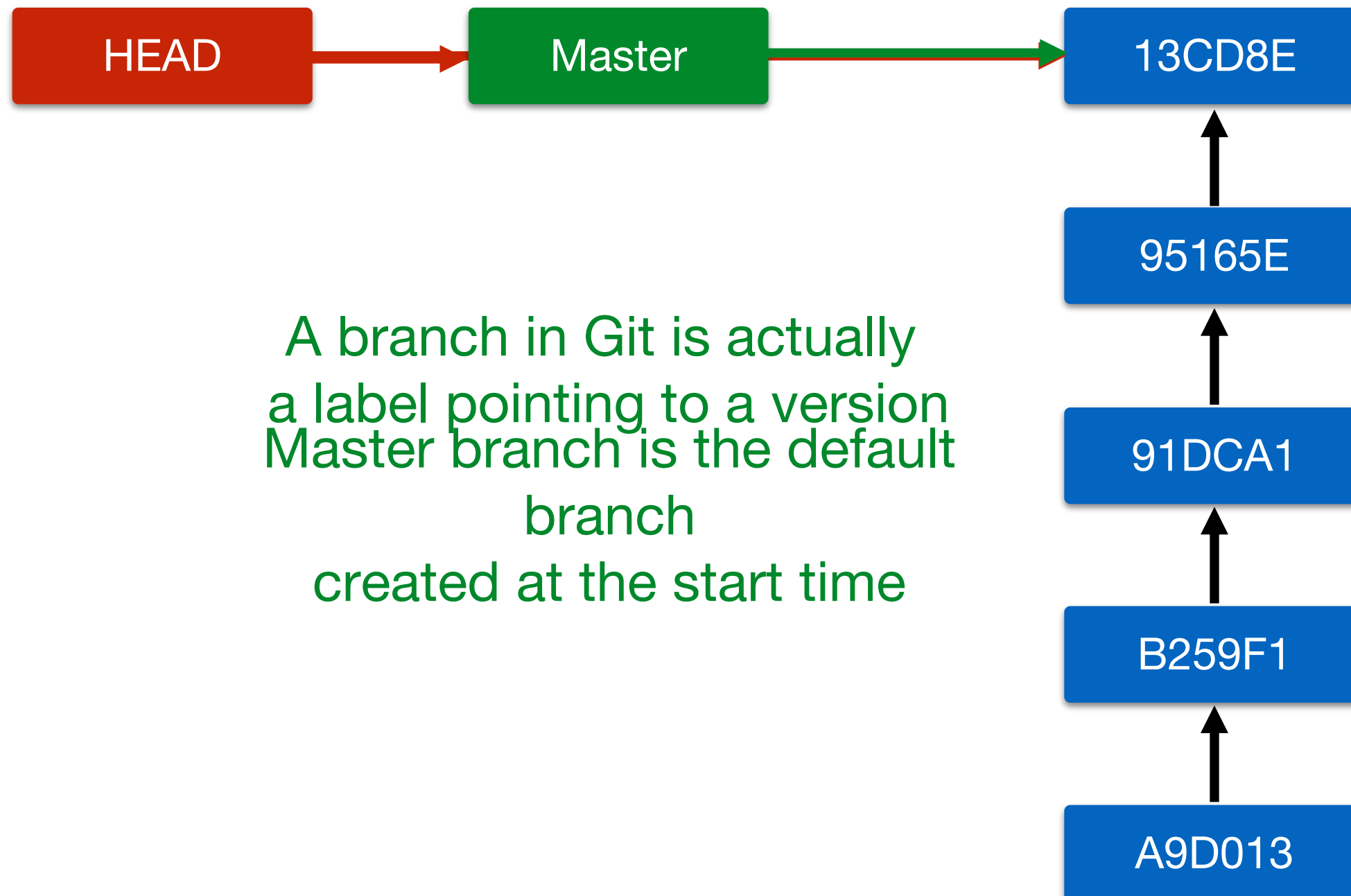
Outline

- Version control system
- Git basics
- **Git branch**
- Remote repository

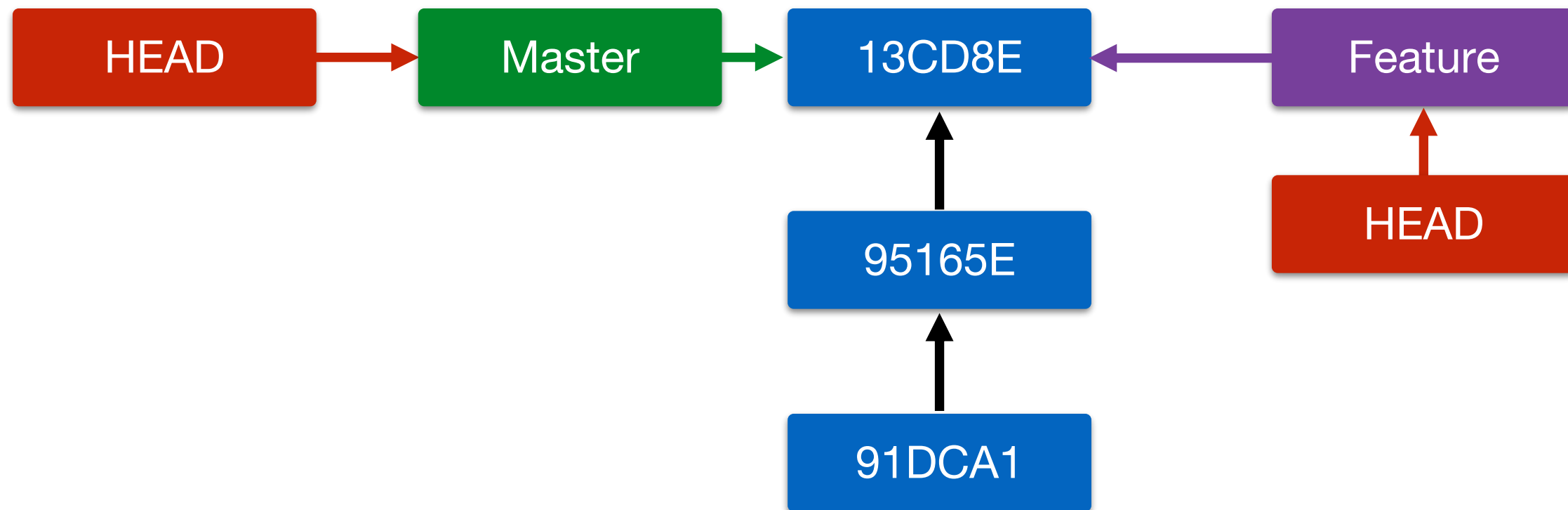
Branches



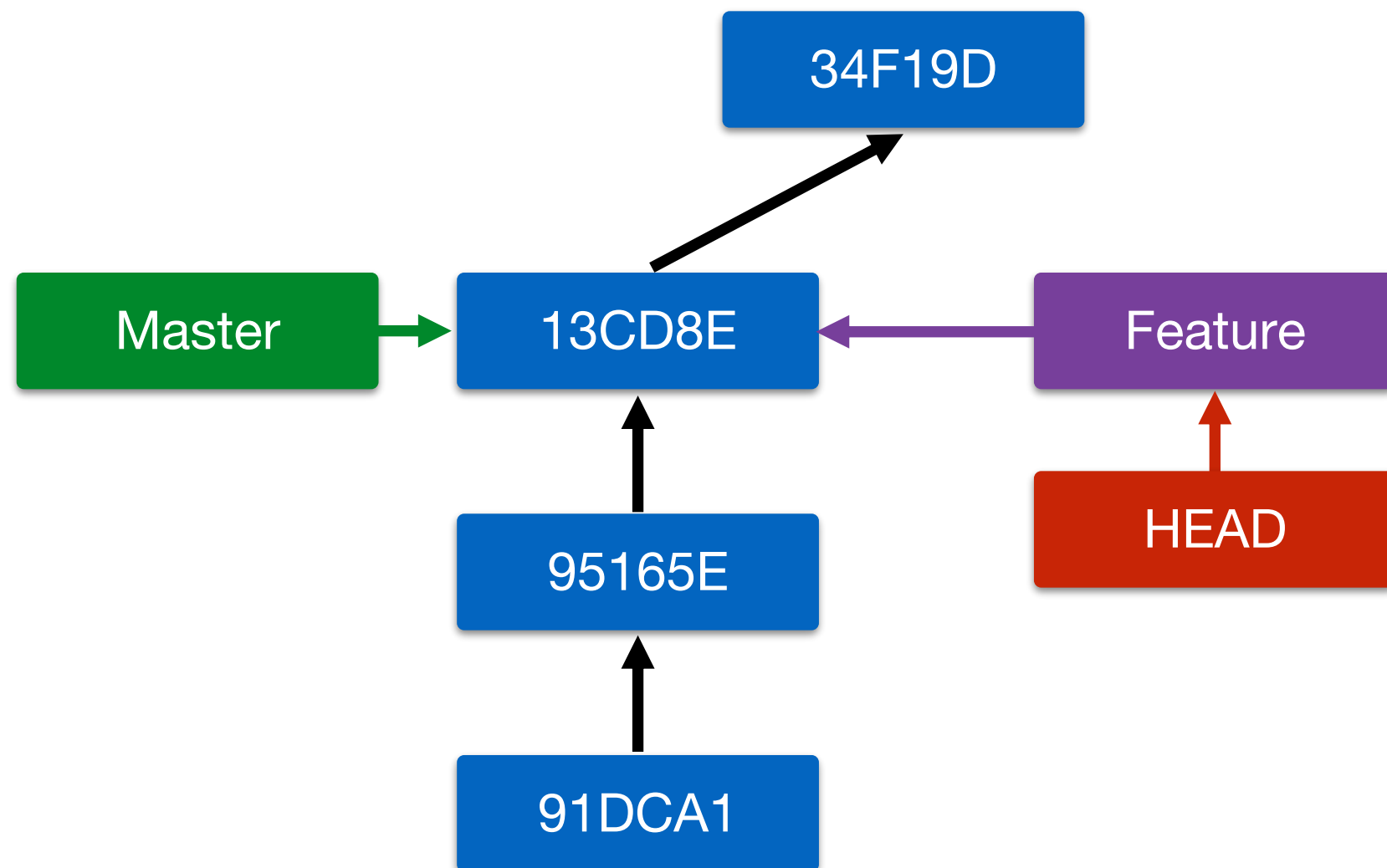
The Master Branch



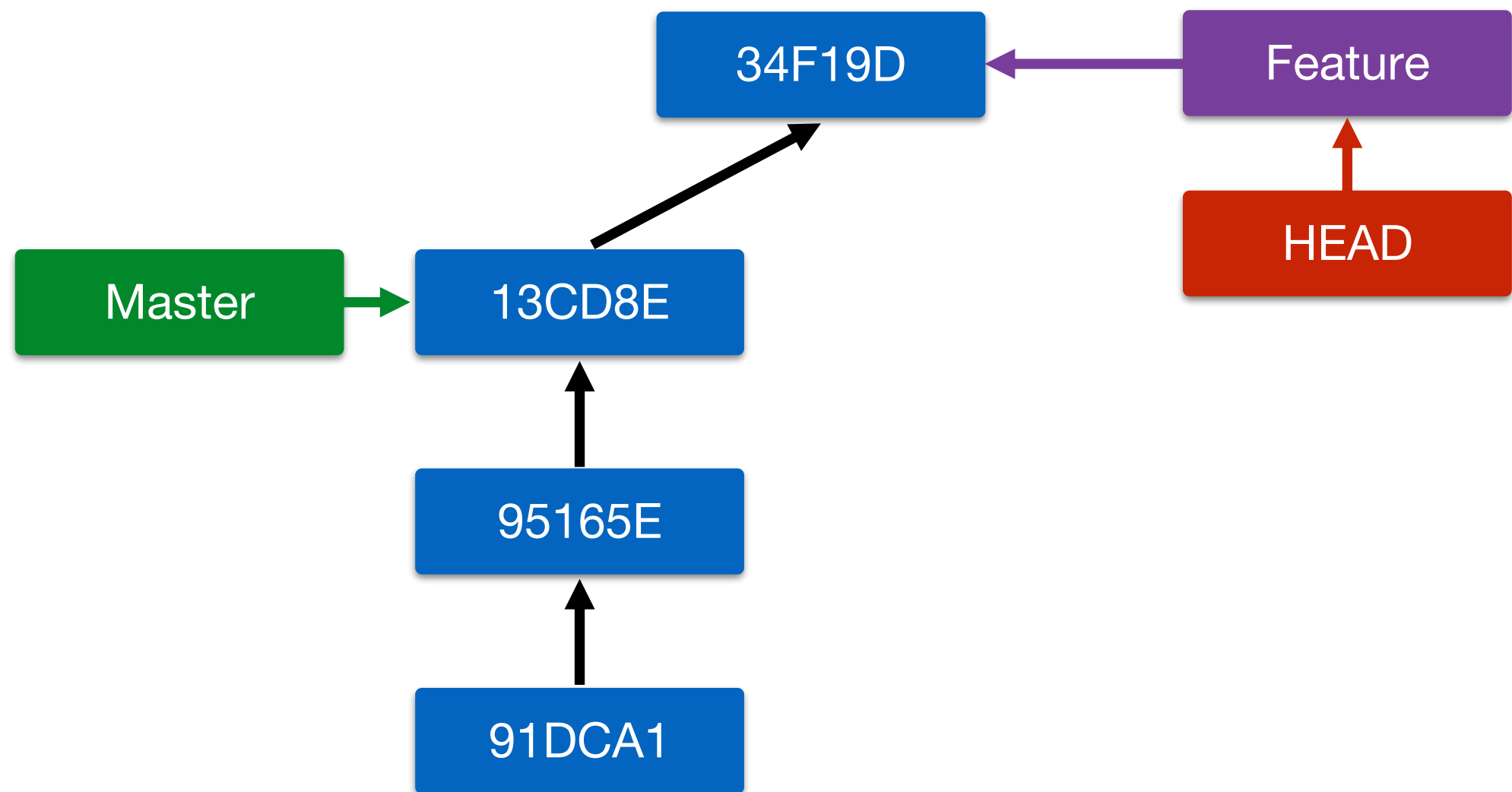
Branching



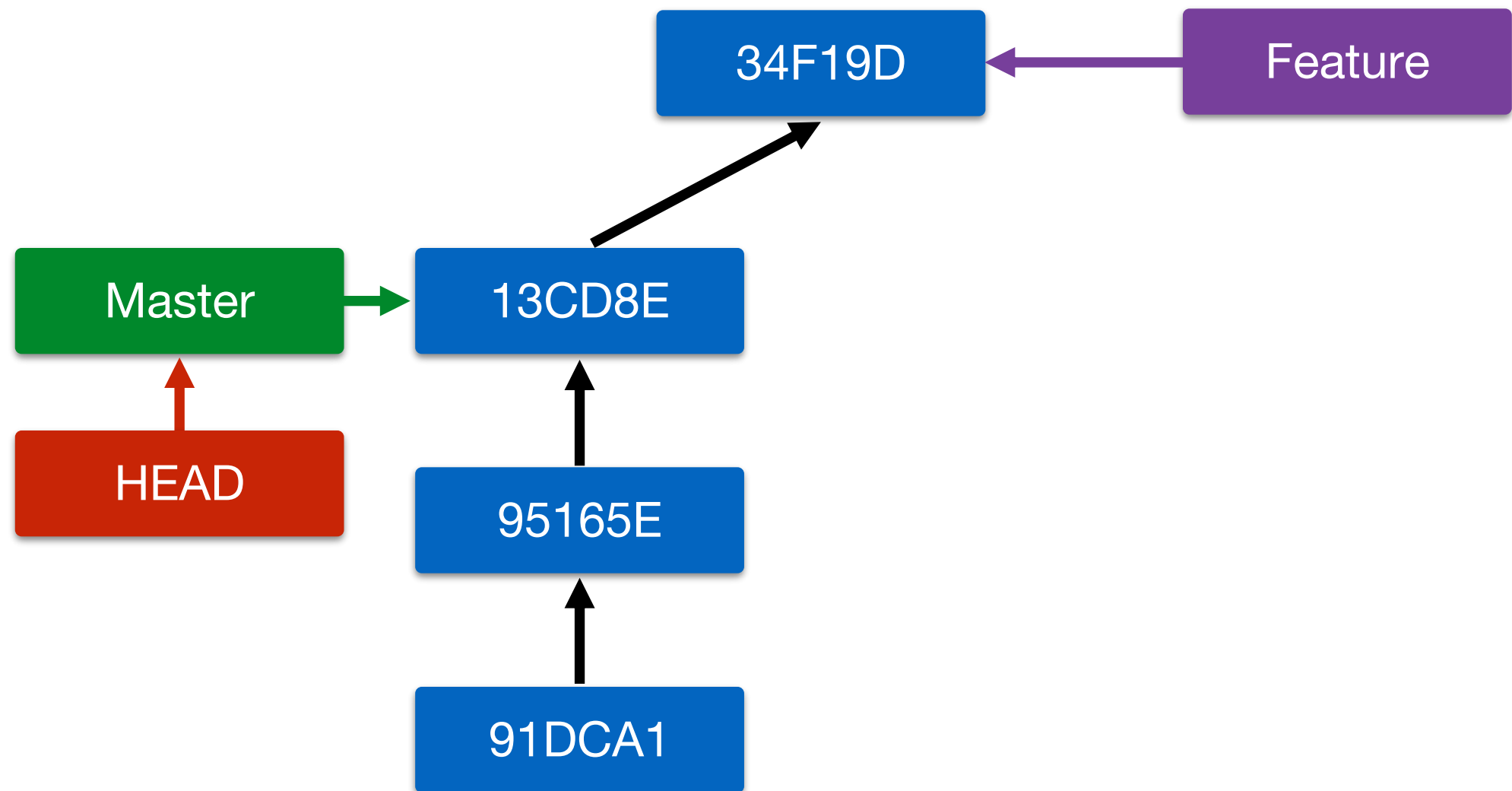
Branching



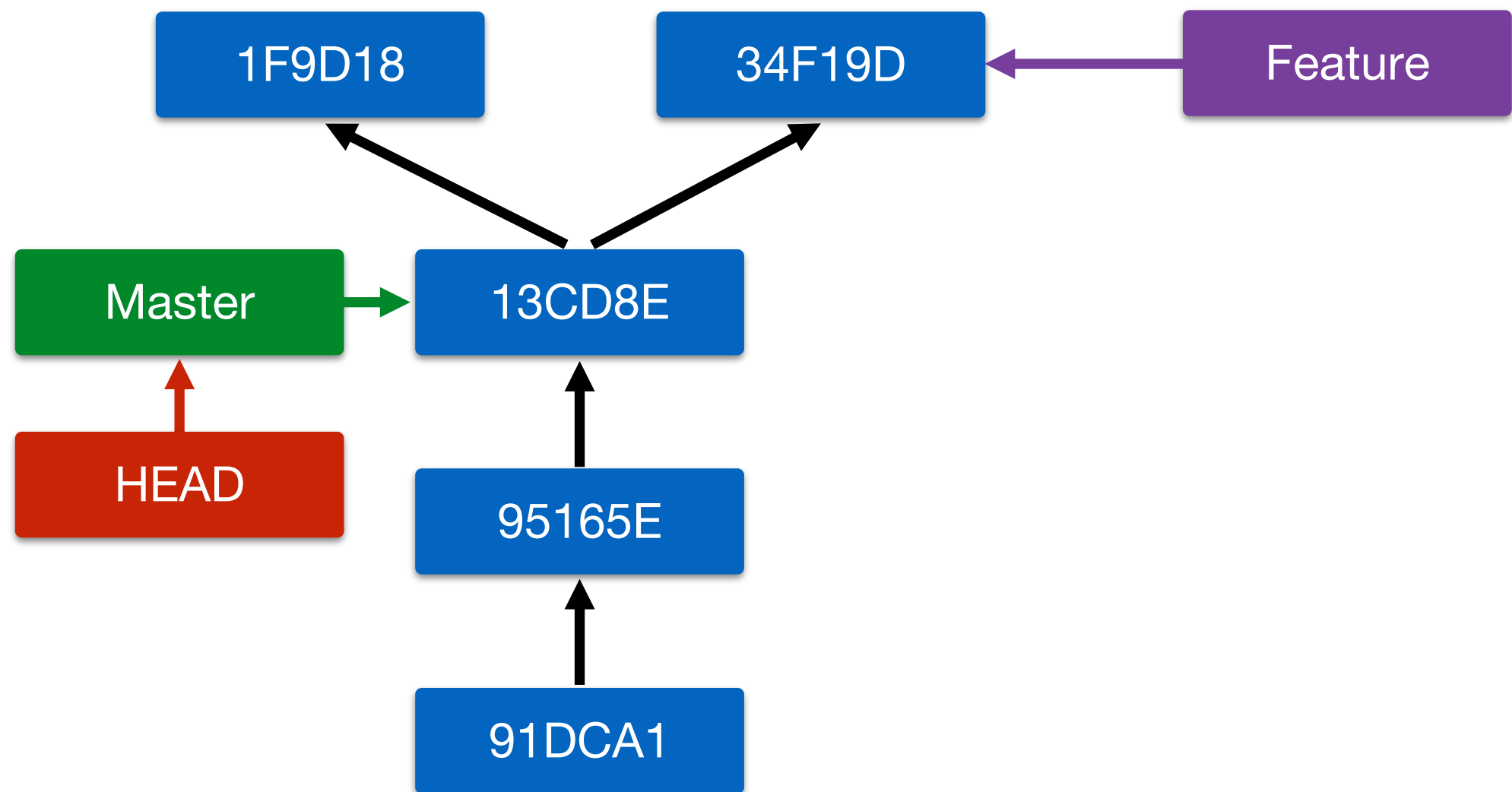
Branching



Branching



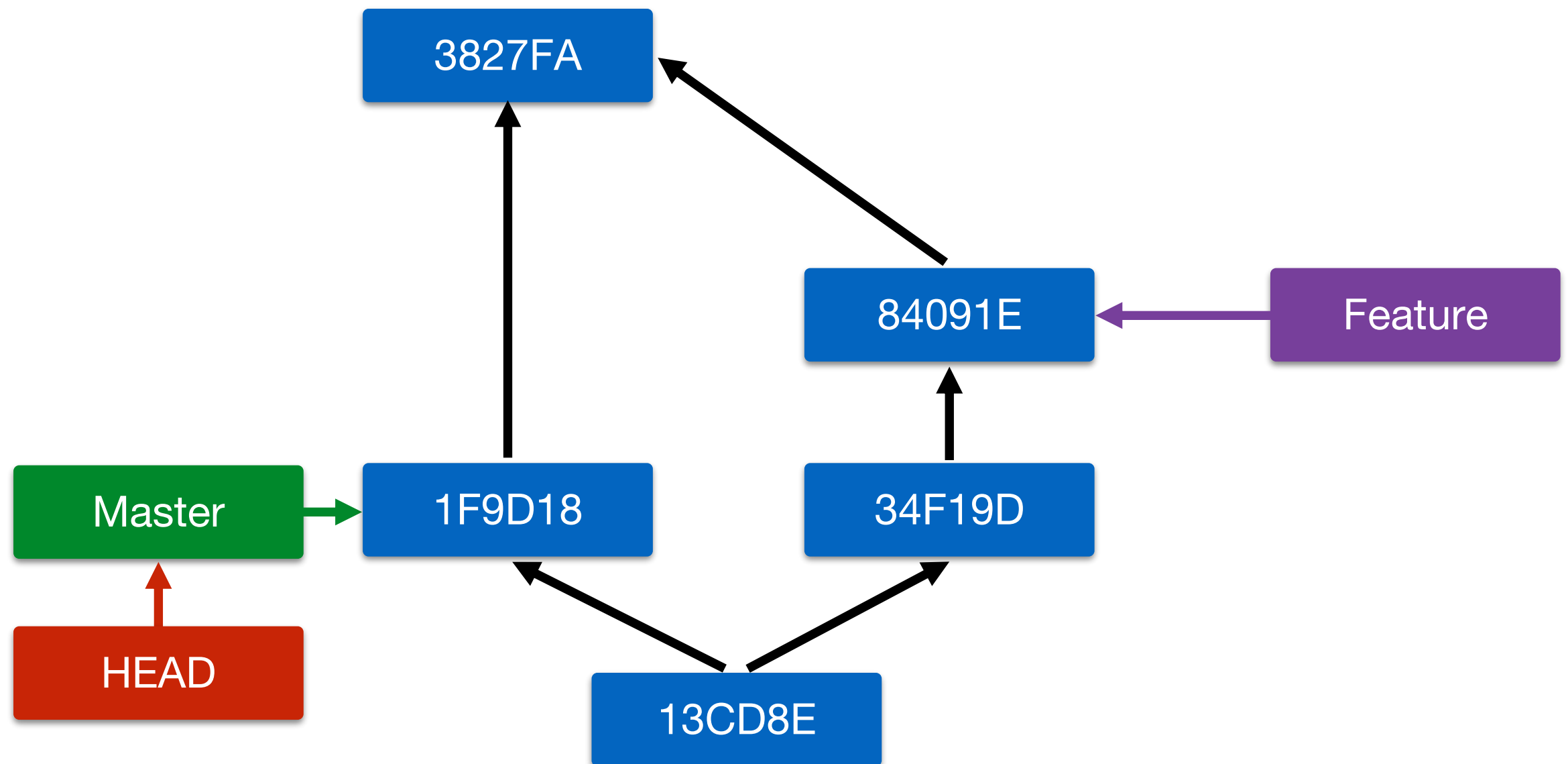
Branching



Git Branching

- Creating a new branch (label)
`git branch [branch name]`
- Checking out the branch (move the HEAD)
`git checkout [branch name]`
- Combining the above commands (create & checkout)
`git checkout -b [branch name]`

Merging



Git Merging

- Merging Steps
 - Checking out a branch to merge
`git checkout [branch 1 name]`
 - Merging another branch
`git merge [branch 2 name]`

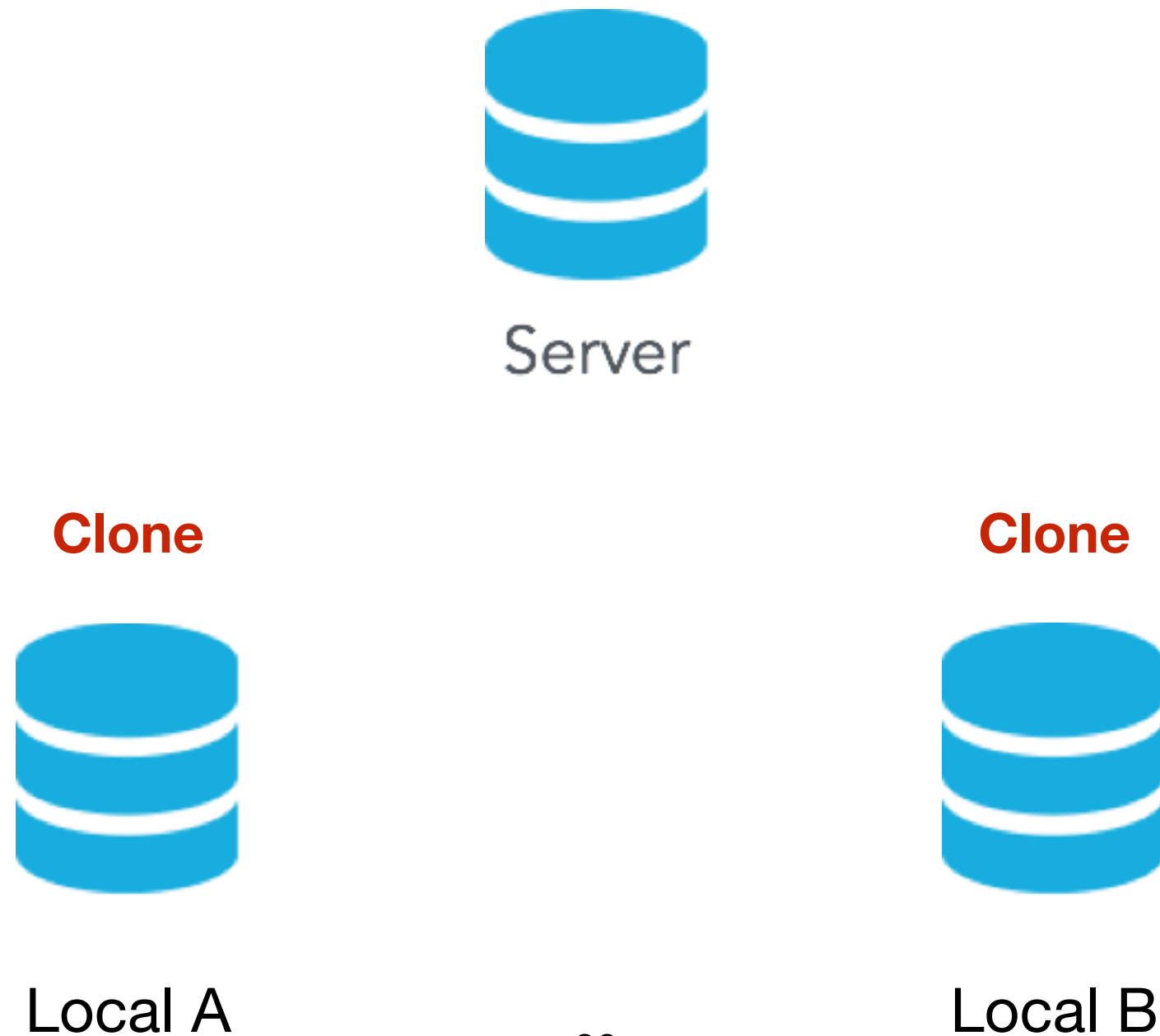
Outline

- Version control system
- Git basics
- Git branch
- **Remote repository**

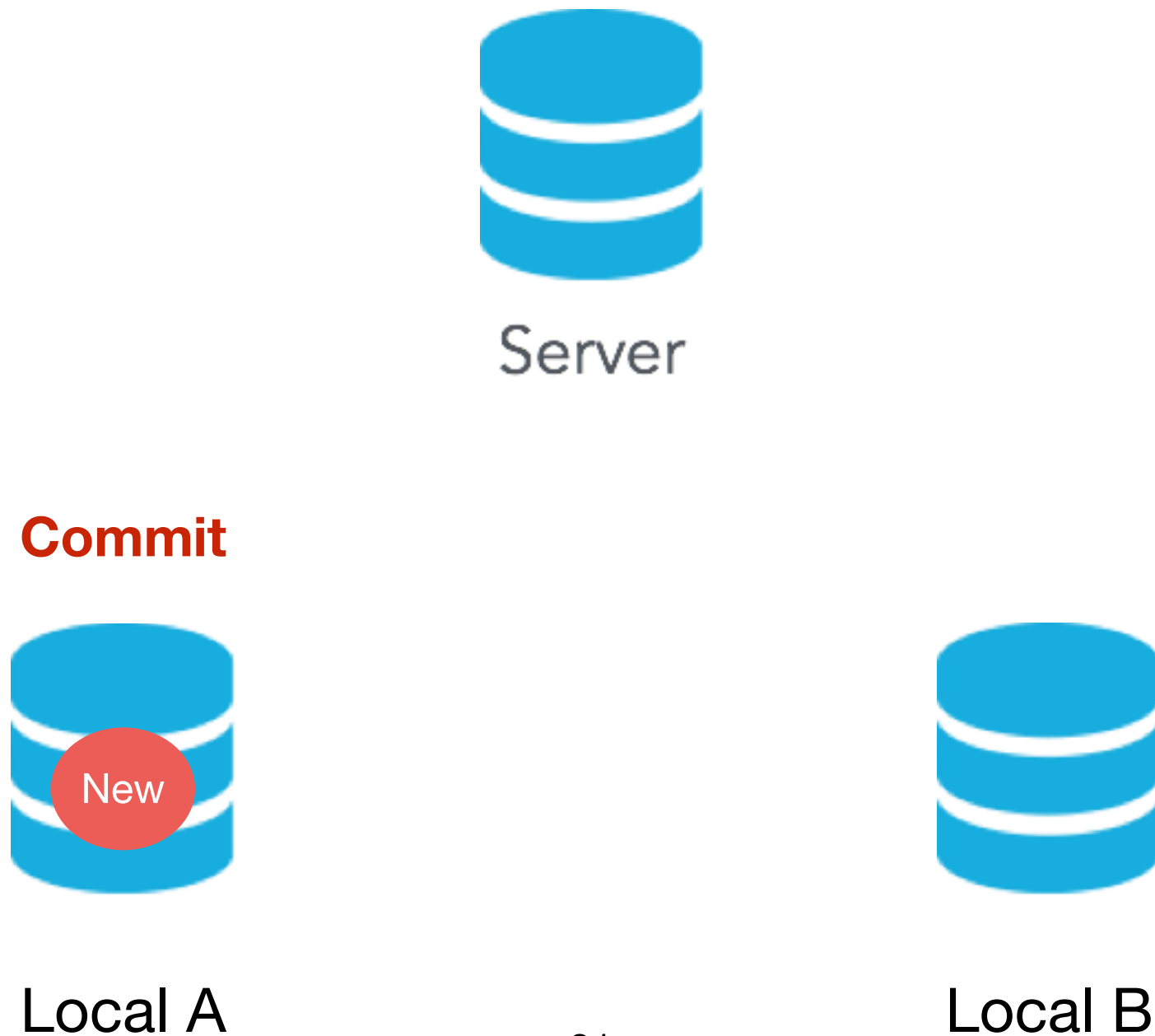
Collaboration with Git

- To work with others using git, you'll need a server that store the repository.
- Git is distributed, which means
 - Everyone can store a copy of the repository downloaded from the server
 - They can do their jobs independently

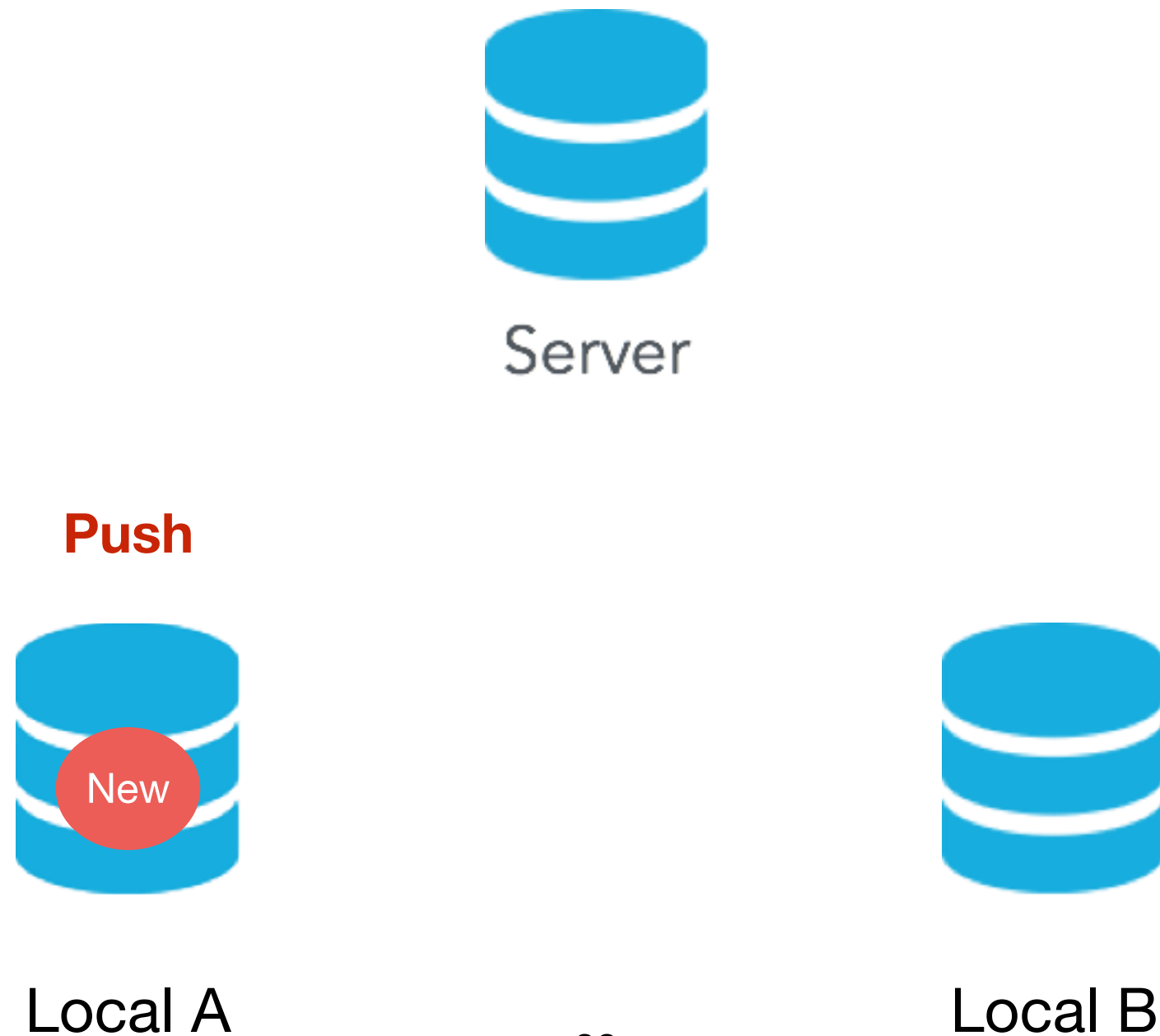
Collaboration workflow



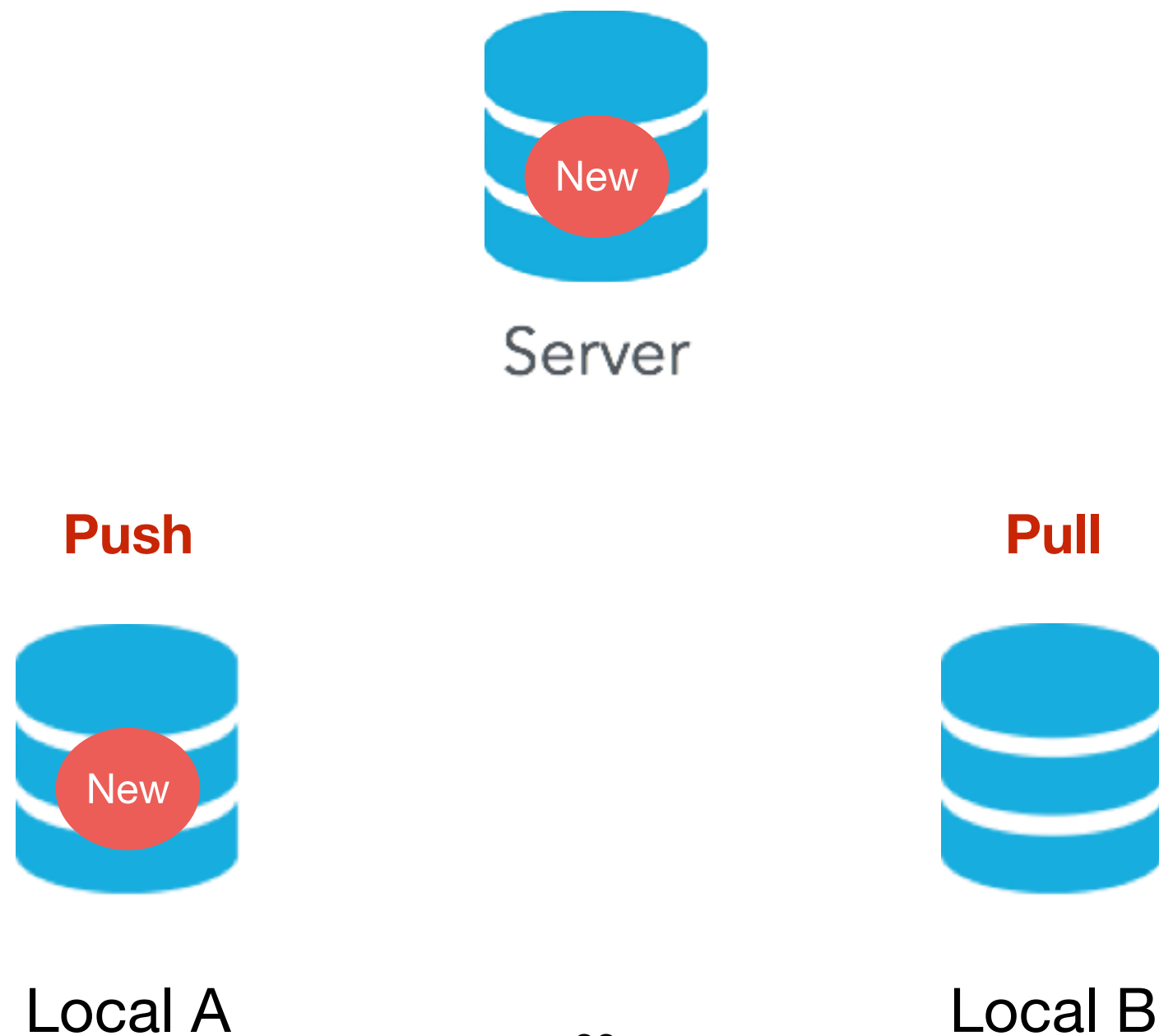
Collaboration workflow



Collaboration workflow



Collaboration workflow



Cloning & Pushing

- Cloning the remote repositories

```
git clone [Remote URL]
```

- The [Remote URL] is saved as **Origin**

- After committing a few versions, you can push the branch back to **Origin**

```
git push -u origin [Branch Name]
```

Fetch & Pull

- Updating a branch from the remote repository

- Fetching the remote repository to local

```
git fetch origin
```

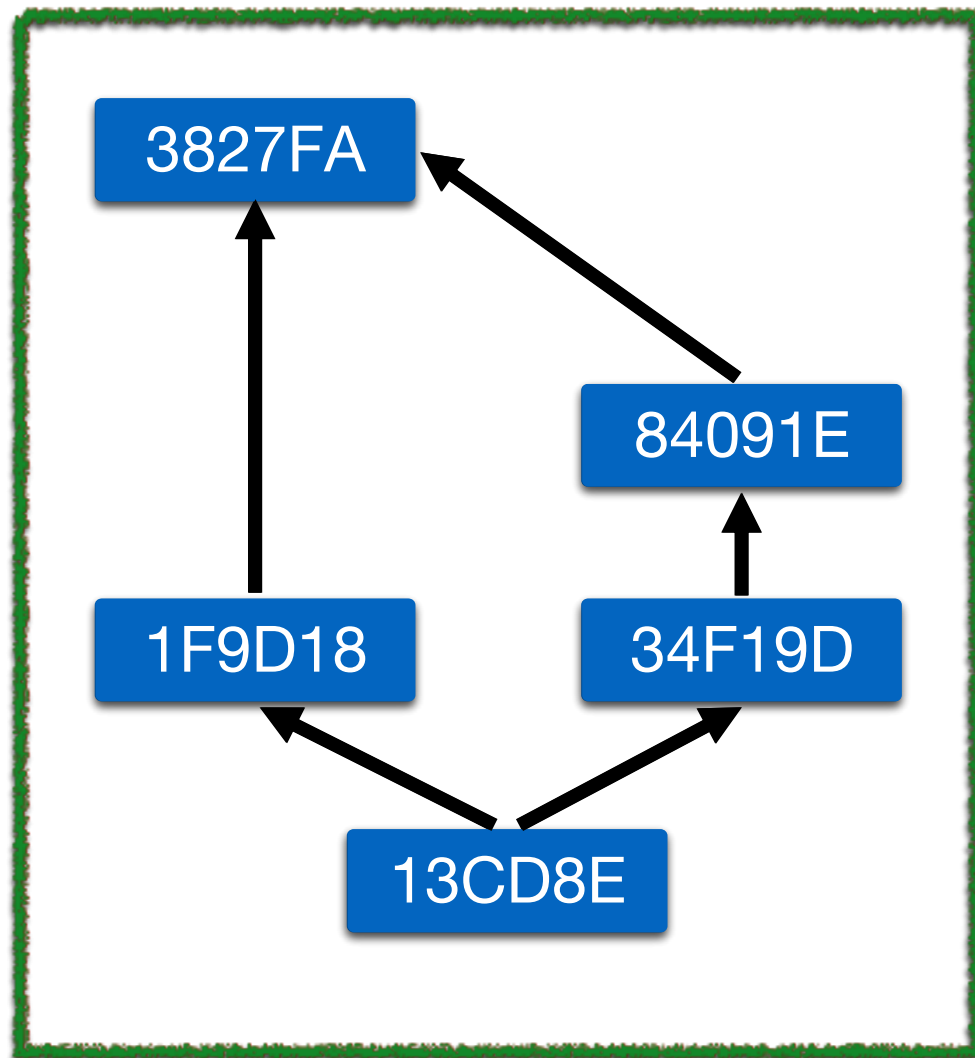
- Merging the remote branch

```
git merge origin/[Branch Name]
```

- Doing above commands in one command

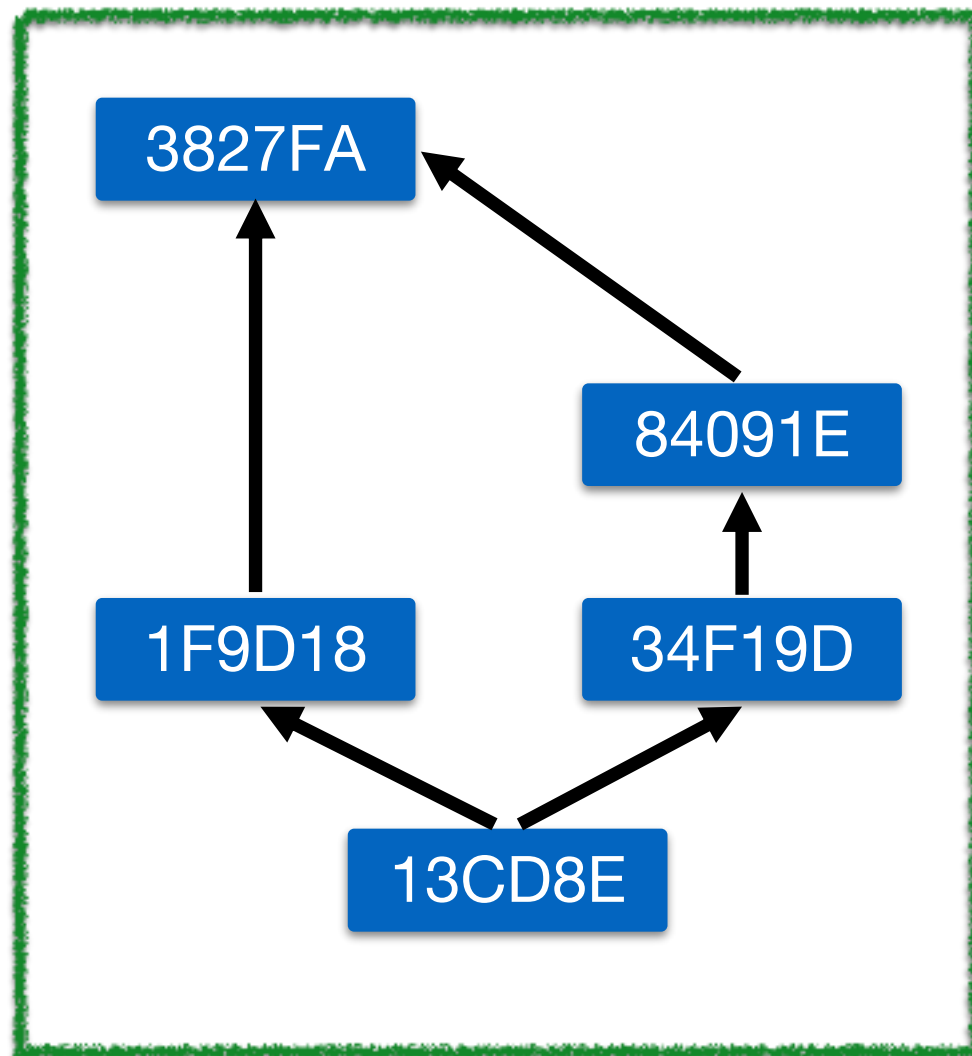
```
git pull [Branch Name]
```

Fork

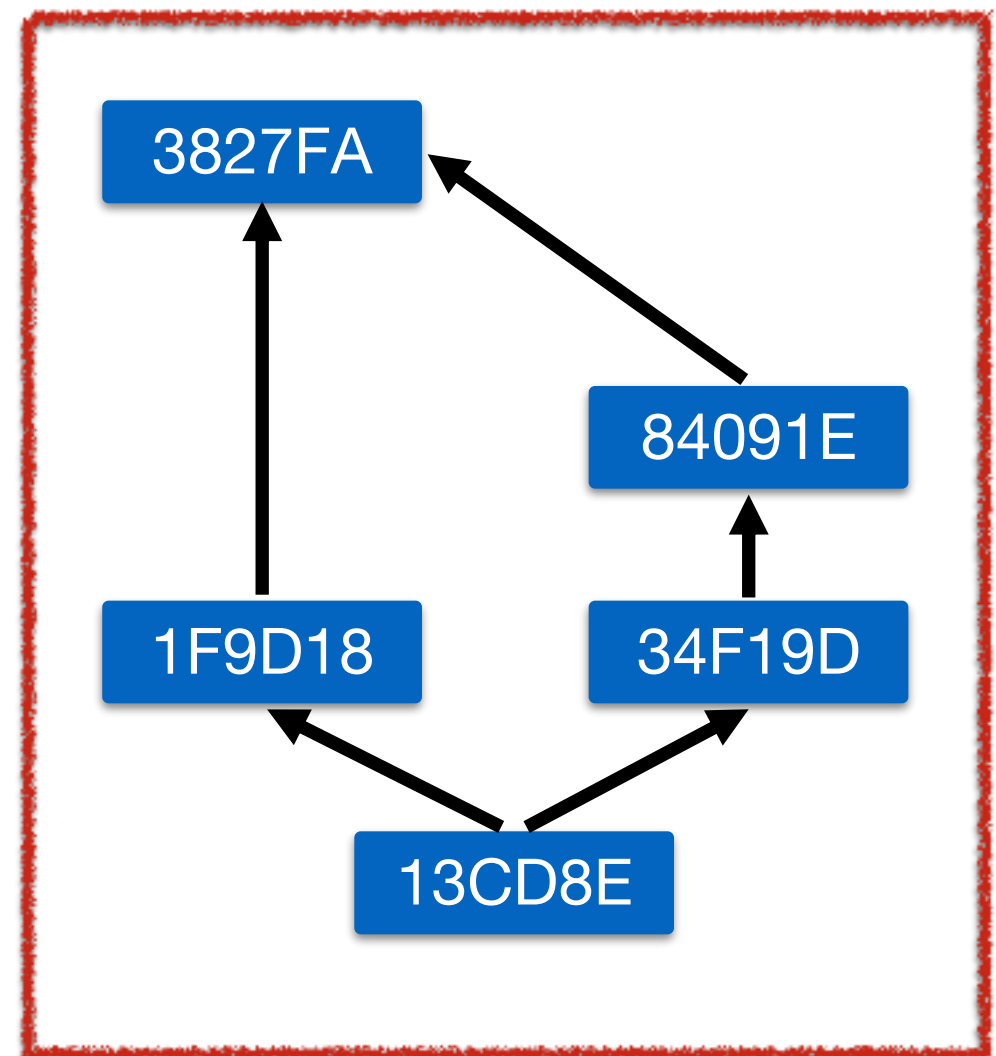


The Repo. Under TA's
Account

Fork

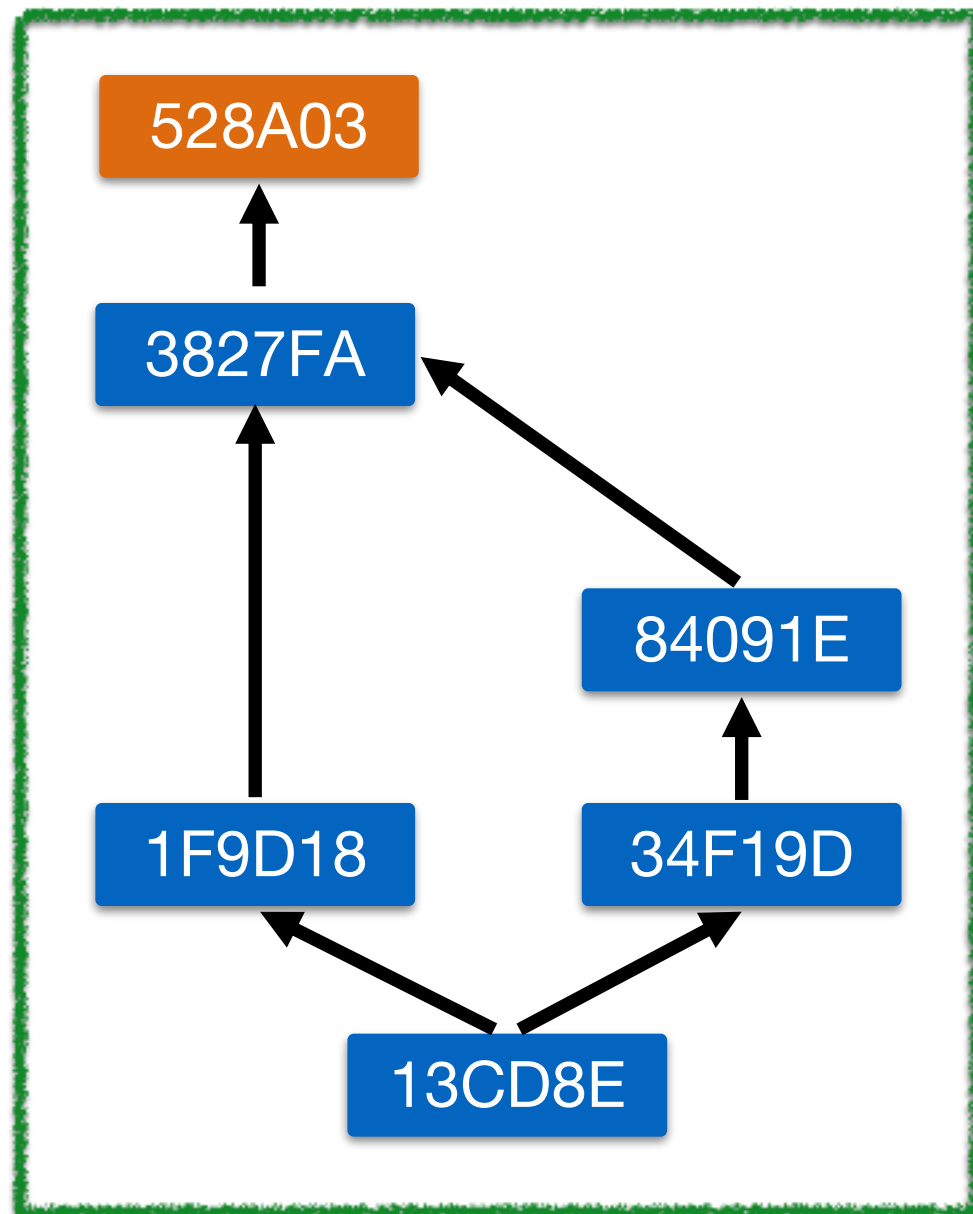


The Repo. Under TA's
Account



The Repo. Under Your
Account

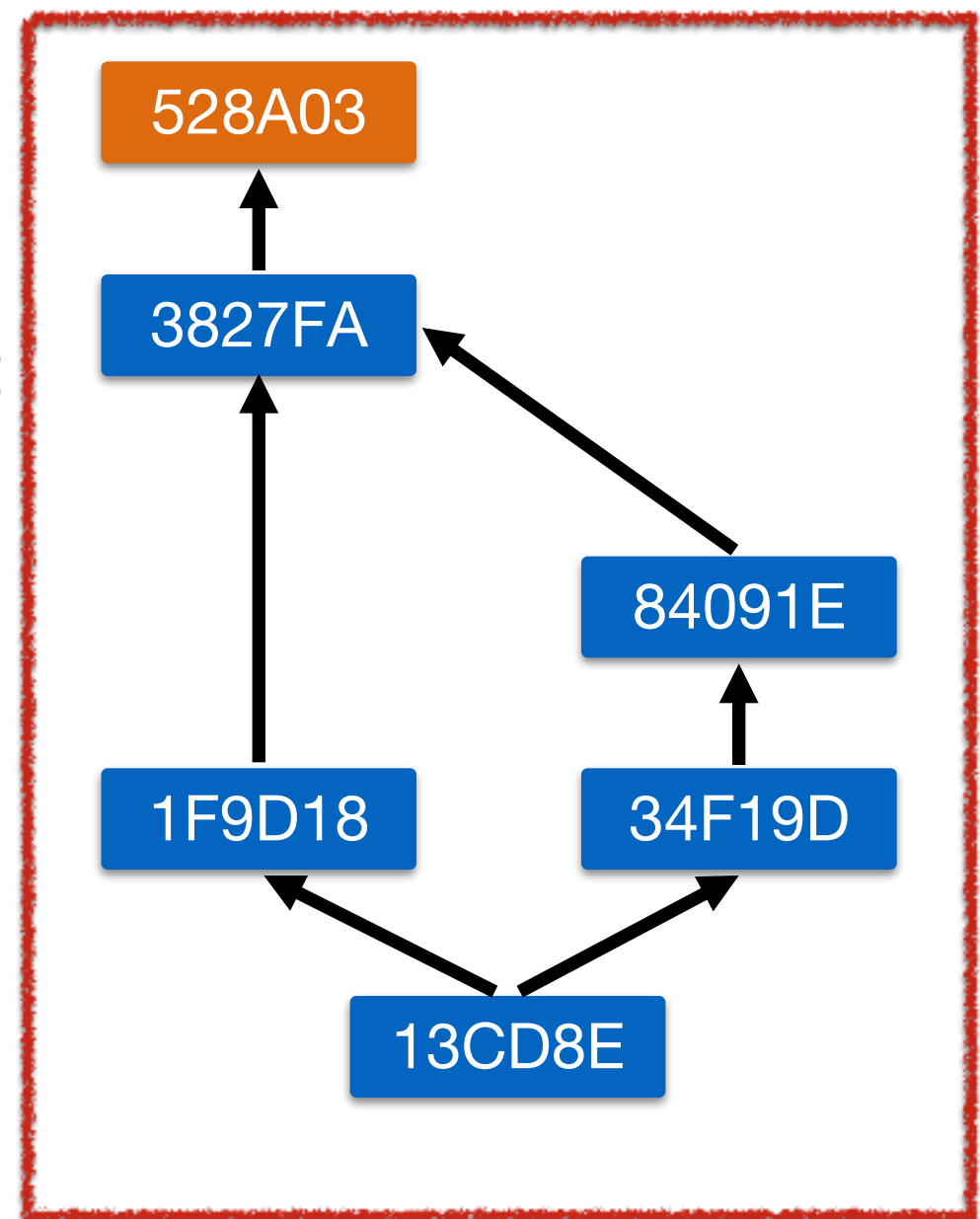
Pull (Merge) Request



The Repo. Under TA's Account

Pull Request

Accept



The Repo. Under Your Account

.gitignore File

- You can ignore some files that you don't want them to be tracked by editing the .gitignore file
- Remember to track and commit your .gitignore file
- Don't know what should be in .gitignore ?
 - <https://github.com/github/gitignore>
 - <https://www.gitignore.io/>

How to Submit Your Code to Gitlab

Workflow

- For each lab, you should follow the workflow below
 1. Fork our template repository on Gitlab
 2. Clone the forked repository to your computer
 3. Finish your lab
 4. Commit in your computer
 5. Push to Gitlab
 6. Send merge request of **your branch** to our template repository

Workflow

- For each lab, you should follow the workflow below
 1. Fork our template repository on Gitlab
 2. Clone the forked repository to your computer
 3. Finish your lab
 4. Commit in your computer
 5. Push to Gitlab
 6. Send merge request of your branch to our template repository

Projects ▾

Groups ▾

Activity

Milestones

Snippets

+

Search or jump to...

🔍

📄

🔗

📧

?

🌐

P

practice-submission

🏠 Project

Details

Activity

Releases

Cycle Analytics

📁 Repository

🔖 Issues0

🔗 Merge Requests0

📖 Wiki

✂ Snippets

👤 Members

⏪ Collapse sidebar

🔗 courses > ... > 2020-spring > practice-submission > Details

P

practice-submission

Project ID: 9897

🔔 ▾

☆ Star0

🍴 Fork0

📄 Clone ▾

🔒 No license. All rights reserved

🔗 1 Commit

🌿 1 Branch

🏷 0 Tags

📄 41 KB Files

master ▾

practice-submission /

+

▾

History

🔍 Find file

Web IDE

🔄 ▾

🌐

Initial commit

Yi-Chun Chen authored 6 hours ago

9af13851

🔗

📄 README

Auto DevOps enabled

Name	Last commit	Last update
📄 README.md	Initial commit	6 hours ago

📄 README.md

Practice Submission

This repository is built for practicing submissions for assignments and projects. You can follow the instructions below in order to know the whole workflow for submitting an assignment or a project.

Try It !!

GitLab

Projects ▾Groups ▾ActivityMilestonesSnippets

Search or jump to... 🔍

🔗📄📁🔍👤

P practice-submission

Project

Details

Activity

Releases

Cycle Analytics

Repository

Issues 0

Merge Requests 0

Wiki

Snippets

Settings

⏪ Collapse sidebar

Pin-Yu Wang > practice-submission > Details

P practice-submission 🔒

Project ID: 9902

🔗 Add license 🔒 0 Commits 🌿 1 Branch 🔗 0 Tags 📄 0 Bytes Files

Forked from [courses / databases / 2020-spring / practice-submission](#)

🔔 ▾ ⭐ Star 0 🍴 Fork 0 Clone ▾

📄 master ▾ practice-submission / + ▾

History 🔍 Find file Web IDE 🔗 ▾

🌐 Initial commit

Yi-Chun Chen authored 6 hours ago

9af13851 🔗

📄 README

Add CHANGELOG

Add CONTRIBUTING

🔗 Auto DevOps enabled

Name	Last commit	Last update
📄 README.md	Initial commit	6 hours ago

📄 README.md

Practice Submission

This repository is built for practicing submissions for assignments and projects. You can follow the instructions below in order to know the whole workflow for submitting an assignment or a project.

1. Click to fork

GitLab

Projects

Groups

Activity

Milestones

Snippets

+

Search or jump to...

Q

P

practice-submission

Project

Details

Activity

Releases

Cycle Analytics

Repository

Issues0

Merge Requests0

Wiki

Snippets

Settings

<<

Collapse sidebar

Pin-Yu Wang > practice-submission > Details

2. Check if this repository is under your account

0 Clone

Add license

0 Commits

1 Branch

0 Tags

0 Bytes Files

Forked from [courses / databases / 2020-spring / practice-submission](#)

master

practice-submission / +

History

Find file

Web IDE

Initial commit

Yi-Chun Chen authored 6 hours ago

9af13851

README

Add CHANGELOG

Add CONTRIBUTING

Auto DevOps enabled

Name	Last commit	Last update
README.md	Initial commit	6 hours ago

README.md

Practice Submission

This repository is built for practicing submissions for assignments and projects. You can follow the instructions below in order to know the whole workflow for submitting an assignment or a project.

3. Go to settings

GitLab

ProjectsGroupsActivityMilestonesSnippets

Search or jump to...

Ppractice-submission

ProjectRepositoryIssues0Merge Requests0WikiSnippetsSettingsGeneralMembersIntegrationsRepository

<< Collapse sidebar

Pin-Yu Wang > practice-submission > General Settings

General project

Update your project name, description, avatar, and other general settings.

Expand

Permissions

Enable or disable certain permissions for this project.

Collapse

Project visibility

Private

The project is accessible only by members of the project. Access must be granted explicitly to each user.

Issues

Lightweight issue tracking system for this project

Only Project Members

Repository

View and edit files in this project

Only Project Members

Merge requests

Submit changes to be merged upstream

Only Project Members

4. Set project to private

GitLab

ProjectsGroupsActivityMilestonesSnippets

P

practice-submission

Project

Repository

Issues0

Merge Requests0

Wiki

Snippets

Settings

General

Members

Integrations

Repository

<< Collapse sidebar

5. Scroll down and save changes

Save changes

Merge request

Customize your merge request restrictions.

Expand

Badges

Customize your project badges. [Learn more about badges.](#)

Expand

Export project

Export this project with all its related data in order to move your project to a new GitLab instance. Once the export is finished, you can import the file from the "New Project" page.

Expand

Advanced

Perform advanced options such as housekeeping, archiving, renaming, transferring, or removing your project.

Expand

Workflow

- For each lab, you should follow the workflow below
 1. Fork our template repository on Gitlab
 2. Clone the forked repository to your computer
 3. Finish your lab
 4. Commit in your computer
 5. Push to Gitlab
 6. Send merge request of your branch to our template repository

Projects

Groups

Activity

Milestones

Snippets

Search or jump to...

+

🔍

📄

🔗

📧

?

⚙️

P

practice-submission

Project

Details

Activity

Releases

Cycle Analytics

Repository

Issues

Merge Requests

Wiki

Snippets

Settings

Pin-Yu Wang > practice-submission > Details

P

practice-submission

Project ID: 9902

Add license

0 Commits

1 Branch

0 Tags

0 Bytes Files

Forked from [courses / databases / 2020-spring / practice-submission](#)

master

practice-submission

+

Initial commit

Yi-Chun Chen authored 6 hours ago

9af13851

🔗

README

Add CHANGELOG

Add CONTRIBUTING

Auto DevOps enabled

Name	Last commit	Last update
README.md	Initial commit	6 hours ago

README.md

Practice Submission

This repository is built for practicing submissions for assignments and projects. You can follow the instructions below in order to know the whole

Clone with SSH

git@shwu10.cs.nthu.edu.tw:pywa

Clone with HTTPS

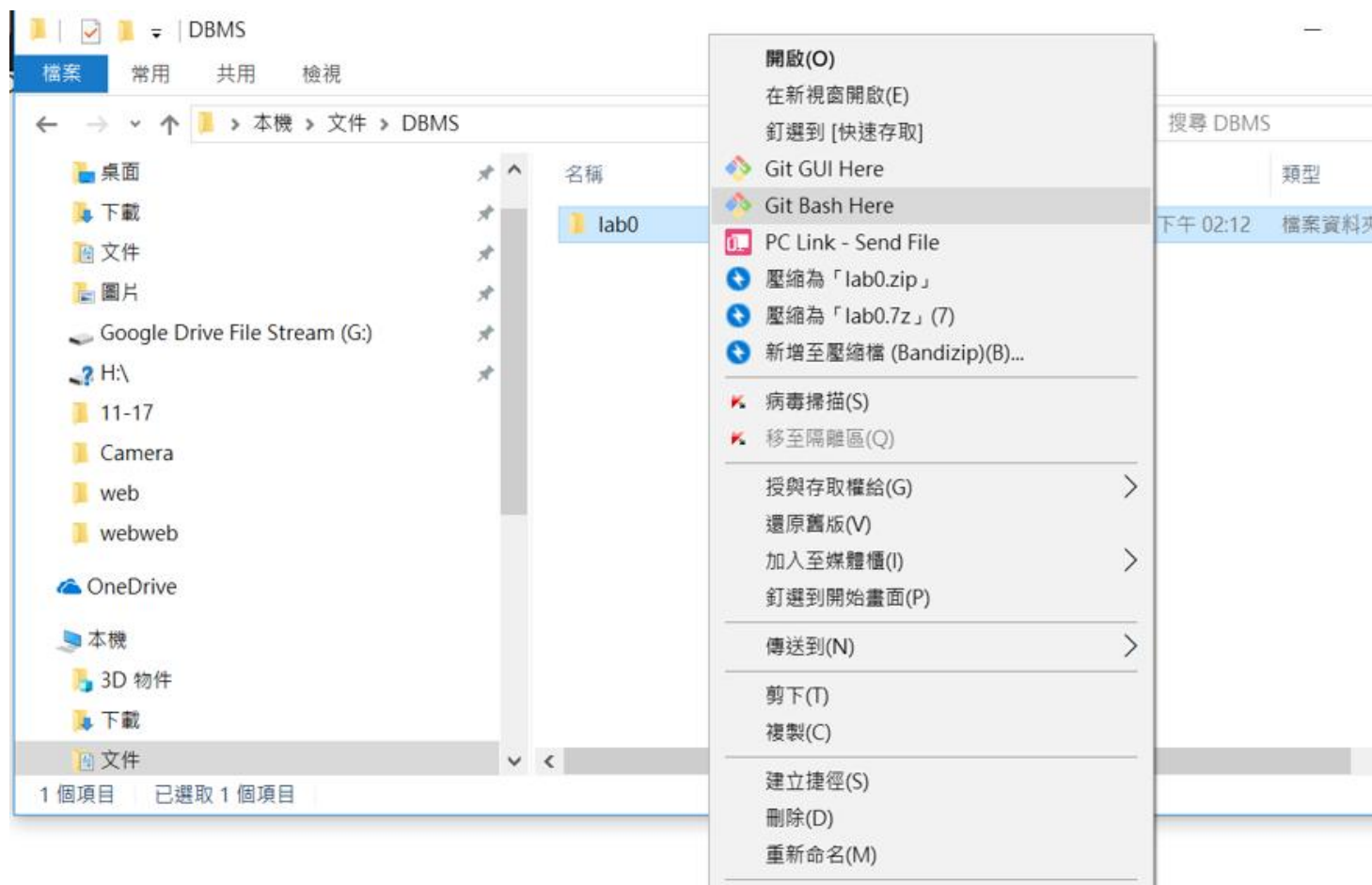
https://shwu10.cs.nthu.edu.tw/p

🔗

2. Copy the link

1. Choose HTTPS

If You use Windows



```
yicchen@LAPTOP-V7AFE0V7 MINGW64 ~/Documents/DBMS/lab0
$ git clone https://shwu10.cs.nthu.edu.tw/ycchen/practice-submission.git
Cloning into 'practice-submission'...
remote: Counting objects: 3, done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 3 (delta 0)
Unpacking objects: 100% (3/3), done.

yicchen@LAPTOP-V7AFE0V7 MINGW64 ~/Documents/DBMS/lab0
$ ls
practice-submission/

yicchen@LAPTOP-V7AFE0V7 MINGW64 ~/Documents/DBMS/lab0
$ ..
```

3. Create a folder to put your repos

```
yicchen@LAPTOP-V7AFE0V7 MINGW64 ~/Documents/DBMS/lab0
$ git clone https://shwu10.cs.nthu.edu.tw/ycchen/practice-submission.git
Cloning into 'practice-submission'...
remote: Counting objects: 3, done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 3 (delta 0)
Unpacking objects: 100% (3/3), done.
```

4. Type "git clone {URL}"

```
yicchen@LAPTOP-V7AFE0V7 MINGW64 ~/Documents/DBMS/lab0
$ ls
practice-submission/
```

```
yicchen@LAPTOP-V7AFE0V7 MINGW64 ~/Documents/DBMS/lab0
$
```

5. The repo has been successfully cloned

Workflow

- For each lab, you should follow the workflow below
 1. Fork our template repository on Gitlab
 2. Clone the forked repository to your computer
 - 3. Finish your lab**
 4. Commit in your computer
 5. Push to Gitlab
 6. Send merge request of your branch to our template repository

Workflow

- For each lab, you should follow the workflow below
 1. Fork our template repository on Gitlab
 2. Clone the forked repository to your computer
 3. Finish your lab
 4. **Commit in your computer**
 5. Push to Gitlab
 6. Send merge request of your branch to our template repository

```
yicchen@LAPTOP-V7AFEOV7 MINGW64 ~/Documents/DBMS/lab0/practice-submission (master)
```

```
$ git add -A
```

1. -A means all files

```
yicchen@LAPTOP-V7AFEOV7 MINGW64 ~/Documents/DBMS/lab0/practice-submission (master)
```

```
$ git status
```

```
On branch master
```

```
Your branch is up-to-date with the upstream branch.
```

2. Check if your file is added to git

```
Changes to be committed:
```

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

```
new file:   practice.txt
```

```
yicchen@LAPTOP-V7AFEOV7 MINGW64 ~/Documents/DBMS/lab0/practice-submission (master)
```

```
$ git commit -m "Finish"
```

```
[master 93a03d5] Finish
```

```
1 file changed
```

```
create mode 100644 practice.txt
```

3. Commit your changes

```
yicchen@LAPTOP-V7AFE0V7 MINGW64 ~/Documents/DBMS/lab0/practice-submission (master)
```

```
git commit -m "Finish"
```

```
** Please tell me who you are.
```

```
un
```

```
git config --global user.email "you@example.com"
```

```
git config --global user.name "Your Name"
```

```
o set your account's default identity.
```

```
mit --global to set the identity only in this repository
```

```
atal: unable to auto-detect email address (got 'yicchen@LAPTOP-V7AFE0V7.(none)')
```

If you see these message, type
git config --global user.name
"{name}"
git config --global user.email "{email}"
{email} is the email you use on gitlab

Workflow

- For each lab, you should follow the workflow below
 1. Fork our template repository on Gitlab
 2. Clone the forked repository to your computer
 3. Finish your lab
 4. Commit in your computer
 - 5. Push to Gitlab**
 6. Send merge request of your branch to our template repository


```
ycchen@LAPTOP-V7AFE0V7 MINGW64 ~/Documents/DBMS/lab0/practice-submission (master)
```

```
$ git push origin master
```

```
Counting objects: 3, done
```

```
Delta compression
```

```
Compressing object
```

```
Writing objects: 100% (3/3), 284 bytes | 284.00 KiB/s, done.
```

```
Total 3 (delta 0), reused 0 (delta 0)
```

```
To https://shwu10.cs.nthu.edu.tw/ycchen/practice-submission.git
```

```
21e2fda..93a03d5 master -> master
```

Type "git push -u origin master"

Workflow

- For each lab, you should follow the workflow below
 1. Fork our template repository on Gitlab
 2. Clone the forked repository to your computer
 3. Finish your lab
 4. Commit in your computer
 5. Push to Gitlab
 6. Send merge request of **your branch** to our template repository

GitLab

Projects

Groups

Activity

Milestones

Snippets

+

Search or jump to...

🔍

📄

🔗

📧

?

🌐

P

practice-submission

🏠 Project

Details

Activity

Releases

Cycle Analytics

📁 Repository

🔖 Issues0

🔗 Merge Requests

📖 Wiki

📄 Snippets

👤 Members

⏪ Collapse sidebar

🌐 courses

>

...

>

2020-spring

>

practice-submission

>

Details

P

practice-submission

Project ID: 9897

🔔

☆ Star0

🍴 Fork0

📄 Clone

📄 No license. All rights reserved

🔗 1 Commit

🌿 1 Branch

🏷️ 0 Tags

📄 41 KB Files

master

practice-submission /

+

📄

History

🔍 Find file

Web IDE

🔗

🌐

Initial commit

Yi-Chun Chen authored 6 hours ago

9af13851

📄

📄

README.md

Name

Last commit

Last update

📄 README.md

Initial commit

6 hours ago



📄

README.md

Practice Submission

This repository is built for practicing submissions for assignments and projects. You can follow the instructions below in order to know the whole workflow for submitting an assignment or a project.

1. Click Merge Requests

 **GitLab** Projects ▾ Groups ▾ Activity Milestones Snippets 

P practice-submission

Project

Repository

Issues 0


Merge Requests 0

Wiki

Snippets

Members

courses > ... > 2020-spring > practice-submission > Merge Requests



Merge requests are a place to propose changes you've made to a project and discuss those changes with others

Interested parties can even contribute by pushing commits if they want to.

New merge request

2. New merge request

60

GitLab

Projects Groups Activity Milestones Snippets

practice-submission

Project

Repository

Issues 0

Merge Requests 0

Wiki

Snippets

Settings

New Merge Request

Source branch

pywang/practice-submission

first commit
Pin-Yu Wang authored 10 minutes ago
6846e605

Compare branches and continue

Target branch

courses/databases/2020-spring/practic...

Initial commit
Yi-Chun Chen authored 7 hours ago
9af13851

3. Choose the branch you pushed in your repo

4. Choose the branch named after your ID

5. Compare branches

Projects ▾

Groups ▾

Activity

Milestones

Snippets

+

This project

Search

🔍

📄

🔖

📧

🌐

P

practice-submission

🏠 Project

📁 Repository

🔖 Issues 0

🔗 Merge Requests 0

📖 Wiki

✂ Snippets

⚙ Settings

New Merge Request

From ycchen/practice-submission

Change branches

Title

std107000000 Submission

Start the title with **WIP:** to prevent a **Work In Progress** merge request from being merged before it's ready.

Add [description templates](#) to help your contributors communicate effectively!

Description

WritePreview

B I

finish|

Markdown and [quick actions](#) are supported

[Attach a file](#)

Assignee

Assignee ▾

[Assign to me](#)

Milestone

Milestone ▾

Labels

Labels ▾

Source branch

master ▾

Target branch

107000000 ▾

[Change branches](#)

☐ Squash commits when merge request is accepted. [About this feature](#)

6. Set title to "{ID} Submission"

GitLab Projects Groups Activity Milestones Snippets

practice-submission

Project Repository Issues 0 Merge Requests 0 Wiki Snippets Settings

Markdown and quick actions are supported [Attach a file](#)

Assignee Assignee [Assign to me](#)

Milestone Milestone

Labels Labels

Source branch master

Target branch 107000000 [Change branches](#)


☐ Squash commits when merge request is accepted. [About this feature](#)


Contribution ☐ Allow commits from members who can merge to the target branch. [About this feature](#)

[Submit merge request](#) [Cancel](#)

Commits 1 Changes 1

24 Feb, 2019 1 commit

 **Finish**
Your Name authored 55 minutes ago

93a03d56 

<< Collapse sidebar

7. If everything is OK,
submit your merge request

Reference

- Learn Git branching (interactive)
 - <http://pcottle.github.io/learnGitBranching/>
- Pro Git
 - <http://git-scm.com/book/>
- 寫給大家的 Git 教學
 - <http://www.slideshare.net/littlebtc/git-5528339>