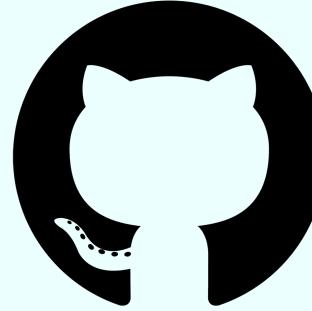




# Git : Part 3

# Collaborate on GitHub



Amine FERDJAOUI  
[amine.ferdjaoui@sogeti.com](mailto:amine.ferdjaoui@sogeti.com)

Année universitaire 202/2025

# What is Github ?

---

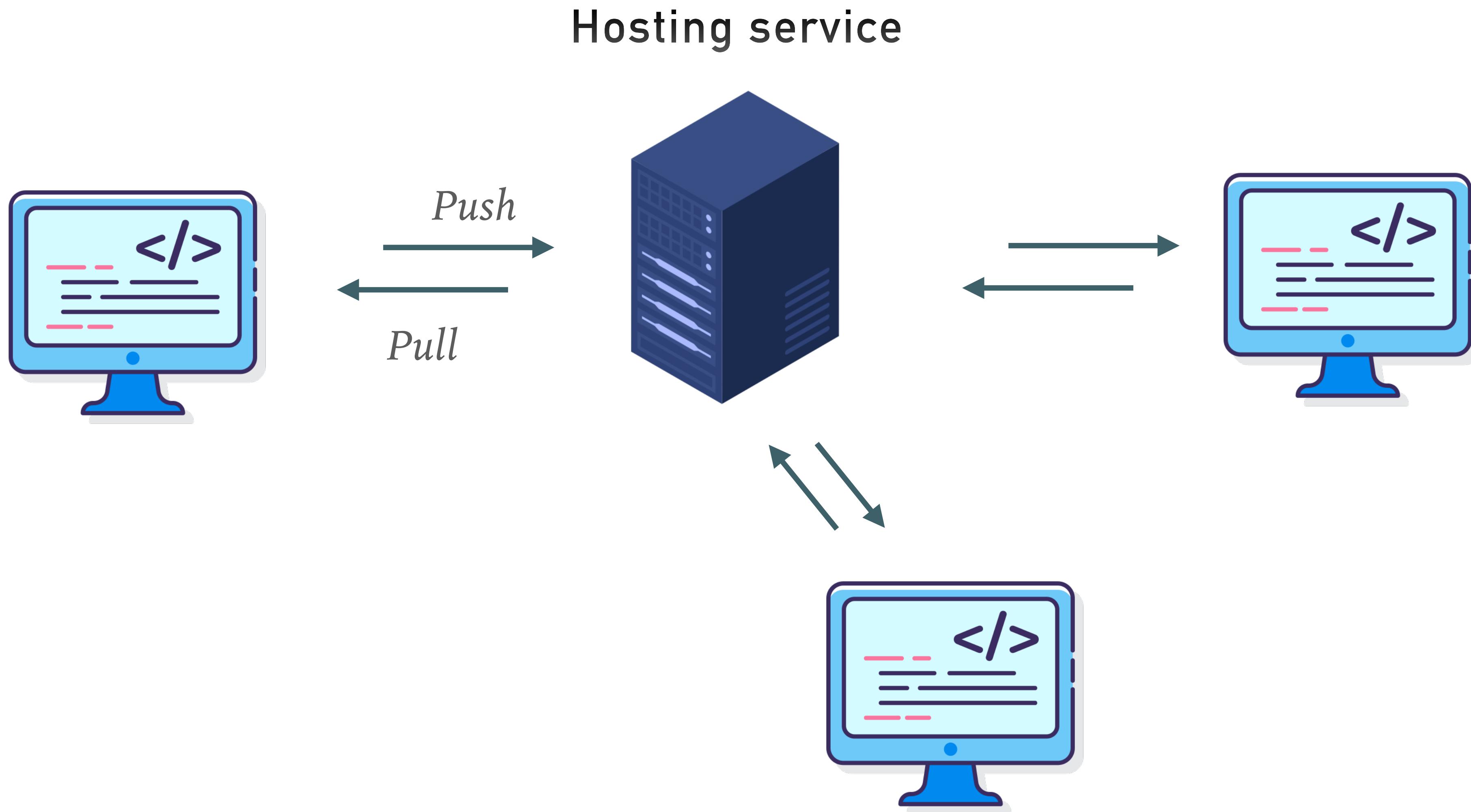


- ◆ Web hosting service that hosts git projects on the cloud
- ◆ A platform for hosting and collaborating on Git repositories
- ◆ The most used ones are : Github, bitbucket and gitlab.
- ◆ **Github** is the most popular one, it was bought by Microsoft in 2018



# What is Github ?

---



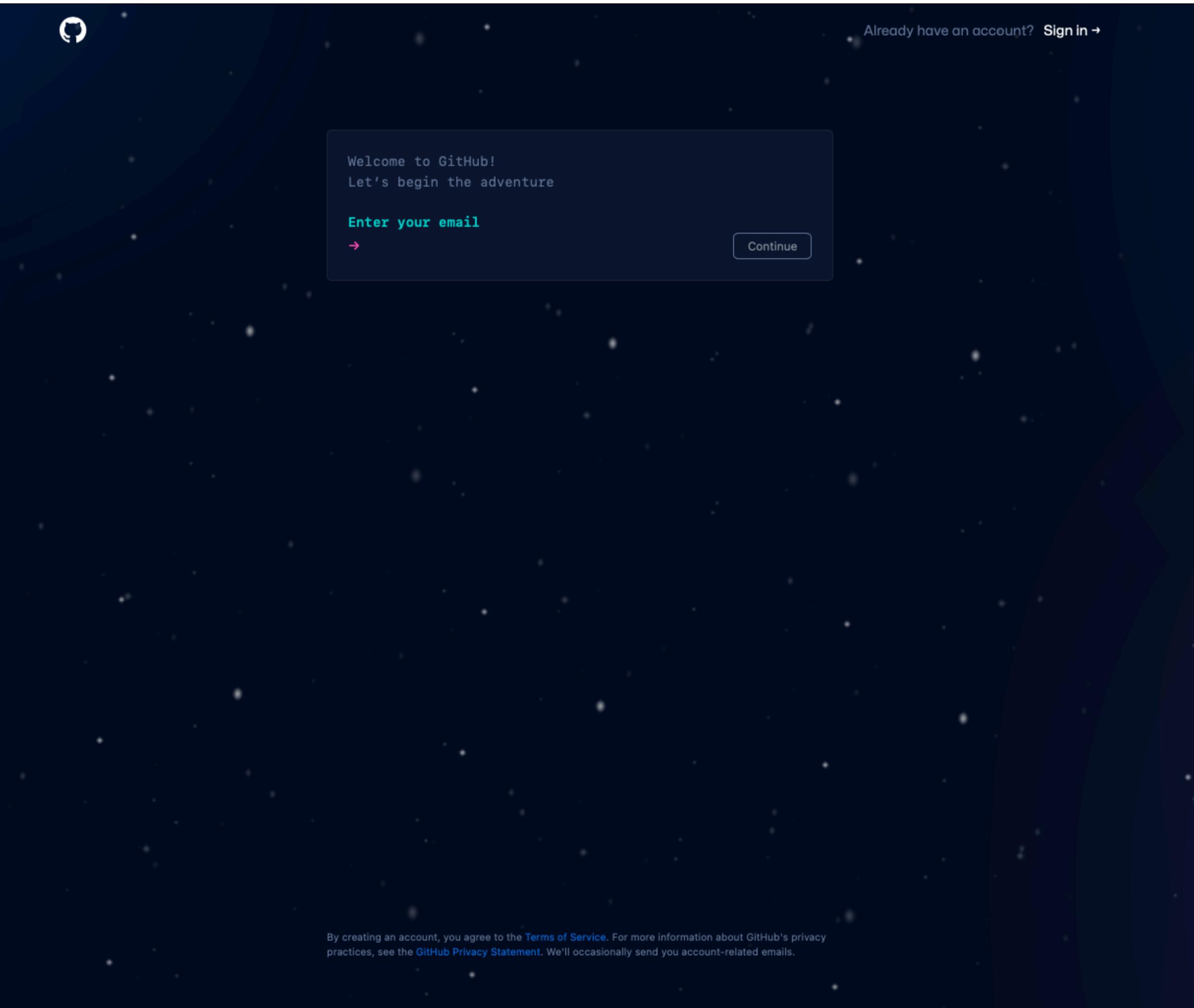
# GITHUB SETUP

---



# Sign up to GitHub

---



# Create a new repository

---

**Create a new repository**

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

---

**Owner \***      **Repository name \***

 AmineFrj  / my-repo 

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

**Description (optional)**

My first repo

---

 **Public**  
Anyone on the internet can see this repository. You choose who can commit.

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

---

**Initialize this repository with:**

Skip this step if you're importing an existing repository.

**Add a README file**  
This is where you can write a long description for your project. [Learn more](#).

**Add .gitignore**

Choose which files not to track from a list of templates. [Learn more](#).

.gitignore template: Python

**Choose a license**

A license tells others what they can and can't do with your code. [Learn more](#).

License: None

---

 You are creating a public repository in your personal account.

---

**Create repository**



© 2022 GitHub, Inc. [Terms](#) [Privacy](#) [Security](#) [Status](#) [Docs](#) [Contact GitHub](#) [Pricing](#) [API](#) [Training](#) [Blog](#) [About](#)

# Initial setup

---

**Quick setup — if you've done this kind of thing before**

[Set up in Desktop](#) or [HTTPS](#) [SSH](#) <https://github.com/AmineFrj/my-repo.git> [Copy](#)

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

**...or create a new repository on the command line**

```
echo "# my-repo" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/AmineFrj/my-repo.git
git push -u origin main
```

**...or push an existing repository from the command line**

```
git remote add origin https://github.com/AmineFrj/my-repo.git
git branch -M main
git push -u origin main
```

**...or import code from another repository**

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

[Import code](#)

💡 **ProTip!** Use the URL for this page when adding GitHub as a remote.



© 2022 GitHub, Inc.

[Terms](#)

[Privacy](#)

[Security](#)

[Status](#)

[Docs](#)

[Contact GitHub](#)

[Pricing](#)

[API](#)

[Training](#)

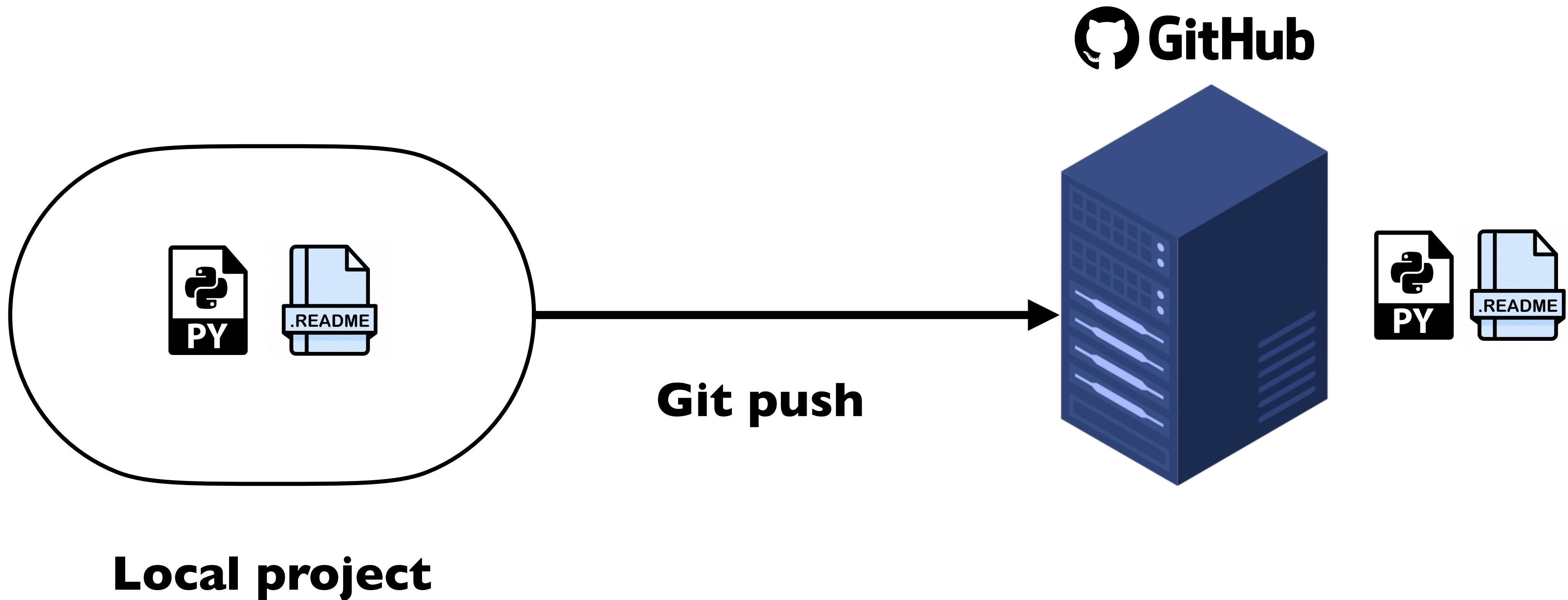
[Blog](#)

[About](#)

# What is a push

---

Git push sends projects into repository in the cloud



# Setup ssh key

---

- ◆ Since august 2021 GitHub password authentication has been removed and replaced by ssh keys, so you don't have to enter your password multiple times.

```
[macbook-pro-de-amine:hello git amine$ git push -u origin main
Username for 'https://github.com': AmineFrj
Password for 'https://AmineFrj@github.com':
remote: Support for password authentication was removed on August 13, 2021.
remote: Please see https://docs.github.com/en/get-started/getting-started-with-git/about-remote-repositories#cloning-with-https-urls for information on currently recommended modes of authentication.
fatal: Authentication failed for 'https://github.com/AmineFrj/my-repo.git/'
```

# Configure SSH keys

The screenshot shows a GitHub repository setup page for a public repository named "AmineFrj/my-repo". The user is signed in as "AmineFrj". A red arrow points from the bottom right towards the "Settings" link in the user menu on the right side of the page.

**Quick setup — if you've done this kind of thing before**

Set up in Desktop or HTTPS SSH <https://github.com/AmineFrj/my-repo.git>

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

**...or create a new repository on the command line**

```
echo "# my-repo" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/AmineFrj/my-repo.git
git push -u origin main
```

**...or push an existing repository from the command line**

```
git remote add origin https://github.com/AmineFrj/my-repo.git
git branch -M main
git push -u origin main
```

**...or import code from another repository**

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

[Import code](#)

💡 **ProTip!** Use the URL for this page when adding GitHub as a remote.

# Configure SSH keys

The screenshot shows the GitHub account settings page for a user named Amine Ferdjaoui. The left sidebar contains navigation links for Public profile, Account, Appearance, Accessibility, Notifications, Access, Billing and plans, Emails, Password and authentication, Sessions, SSH and GPG keys (which is selected and highlighted with a red arrow), Organizations, and Moderation. The main content area displays sections for SSH keys, GPG keys, and Vigilant mode. The SSH keys section indicates no keys are associated, with a link to generate one. The GPG keys section also indicates no keys are associated, with a link to generate one. The Vigilant mode section includes a checkbox for flagging unsigned commits as unverified, with explanatory text and a link to learn more. A red arrow points from the 'SSH and GPG keys' link in the sidebar to the 'New SSH key' button in the main content area.

Search or jump to... / Pull requests Issues Codespaces Marketplace Explore

Amine Ferdjaoui  
Your personal account

Public profile Account Appearance Accessibility Notifications

Billing and plans Emails Password and authentication Sessions

**SSH and GPG keys** (selected)

Organizations Moderation

Code, planning, and automation

Repositories Codespaces Packages Copilot Pages Saved replies

Security

Code security and analysis

Integrations

Applications Scheduled reminders

Archives

Security log Sponsorship log

SSH keys

New SSH key

There are no SSH keys associated with your account.

Check out our guide to [generating SSH keys](#) or troubleshoot [common SSH problems](#).

GPG keys

New GPG key

There are no GPG keys associated with your account.

Learn how to [generate a GPG key](#) and add it to your account.

Vigilant mode

**Flag unsigned commits as unverified**

This will include any commit attributed to your account but not signed with your GPG or S/MIME key.  
Note that this will include your existing unsigned commits.

[Learn about vigilant mode.](#)

Go to your personal profile

# Setup ssh key

---

- ◆ First check if you have no SSH keys on your computer

```
$ ls -al ~/.ssh
```

- ◆ SSH keys supported by GitHub are :
  - ◆ id\_rsa.pub
  - ◆ id\_ecdsa.pub
  - ◆ id\_ed25519.pub
- ◆ If you already have one just copy it into your GitHub settings, otherwise follow the instructions : <https://docs.github.com/en/authentication/connecting-to-github-with-ssh/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent>

# Setup multiple ssh keys

---

- ◆ If you have multiple accounts, make sure to add specific key for each one in `~/.ssh/config`  
then add ssh private keys to your agent with : `ssh-add ~/.ssh/<key-name>`
- ◆ Cf. <https://gist.github.com/oanhnn/80a89405ab9023894df7>

# Initial push

---

```
$ git remote add origin git@<repo_url>
$ git branch -M main
$ git push -u origin main
```

Nb. Use the **ssh url** in *git remote add origin*

```
$ echo "This repo is a python test on GitHub" > README.md
$ git add README.md
$ git commit -m 'Add a README.'
$ git push
```

# Initial push

A screenshot of a GitHub repository page for 'AmineFrj/my-repo'. The repository is public and contains 1 branch and 0 tags. The 'Code' tab is selected. A red arrow points to the '7 commits' link in the commit list, which shows four commits: 'Add README.' (4 minutes ago), 'Add README.' (minutes ago), 'add new comment 2' (4 hours ago), and 'Add new print' (9 hours ago). The 'About' section indicates no description, website, or topics provided. The 'Releases' section shows no releases published. The 'Languages' section shows Python at 100.0%.

Search or jump to... /

Pull requests Issues Codespaces Marketplace Explore

AmineFrj / my-repo Public

Pin Unwatch 1 Fork 0 Star 0

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main 1 branch 0 tags Go to file Add file Code

AmineFrj Add README. 912ffa1 4 minutes ago 7 commits

README.md Add README.

main2.py add new comment 2

requirements.txt Add new print

README.md My readme

About

No description, website, or topics provided.

Readme 0 stars 1 watching 0 forks

Releases

No releases published Create a new release

Packages

No packages published Publish your first package

Languages

Python 100.0%

# Initial push

The screenshot shows a GitHub repository page for 'AmineFrj / my-repo'. The 'Code' tab is selected. The commit history is displayed under the 'main' branch. There are two main sections: 'Commits on Nov 28, 2022' and 'Commits on Nov 27, 2022'. Each commit includes the author (AmineFrj), date, message, copy icon, commit hash, and a diff icon.

- Commits on Nov 28, 2022:**
  - Add README.  
AmineFrj committed 4 minutes ago
  - add new comment 2  
AmineFrj committed 4 hours ago
  - added comment  
AmineFrj committed 4 hours ago
  - Rename main.py  
AmineFrj committed 9 hours ago
  - add readme  
AmineFrj committed 9 hours ago
  - Add new print  
AmineFrj committed 9 hours ago
- Commits on Nov 27, 2022:**
  - Initial commit.  
AmineFrj committed yesterday

At the bottom, there are 'Newer' and 'Older' buttons.

# Clone project

---

- Once you pushed the initial commit, you can clone it using this clone command

```
$ git clone <repo_url>
```

- You can see that git added the remote branches under origin

```
$ git branch -a
```

# Git visualization

---

<https://git-school.github.io/visualizing-git/>

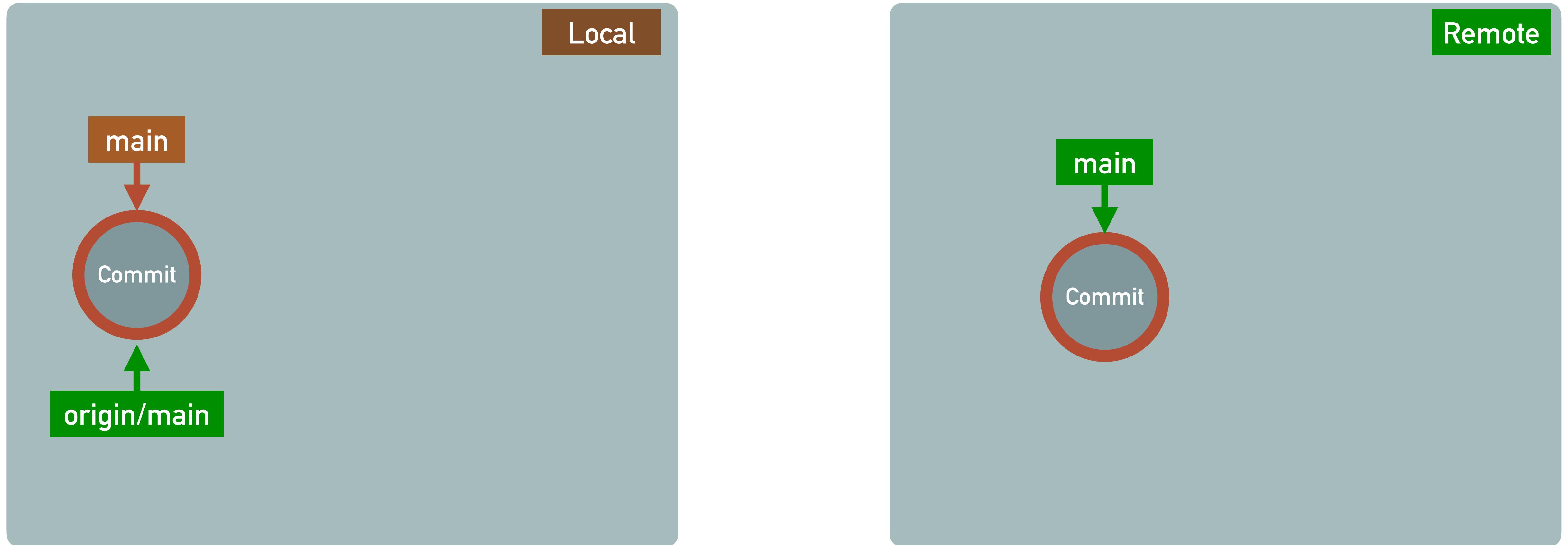
# GIT FETCH & GIT PULL

---

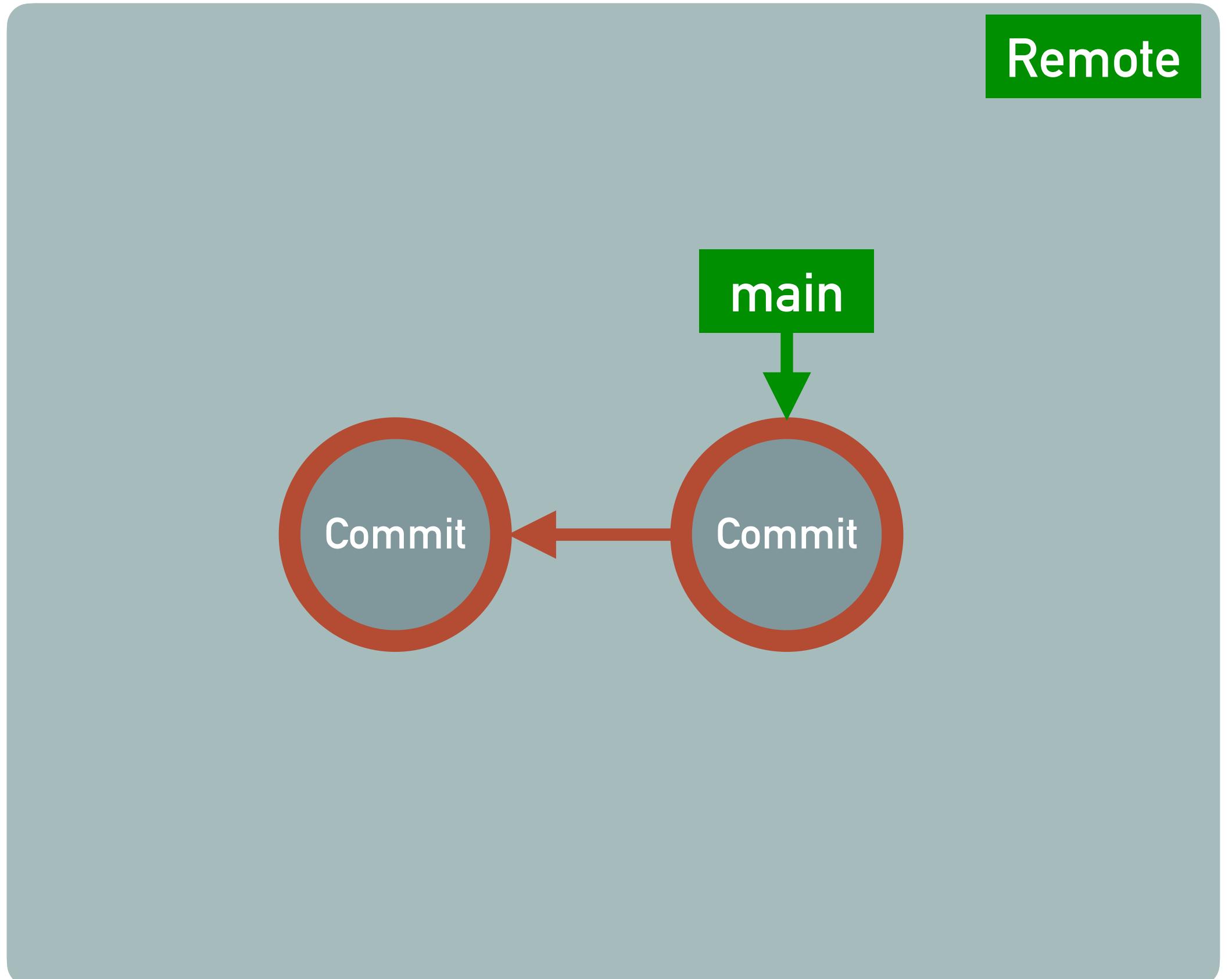
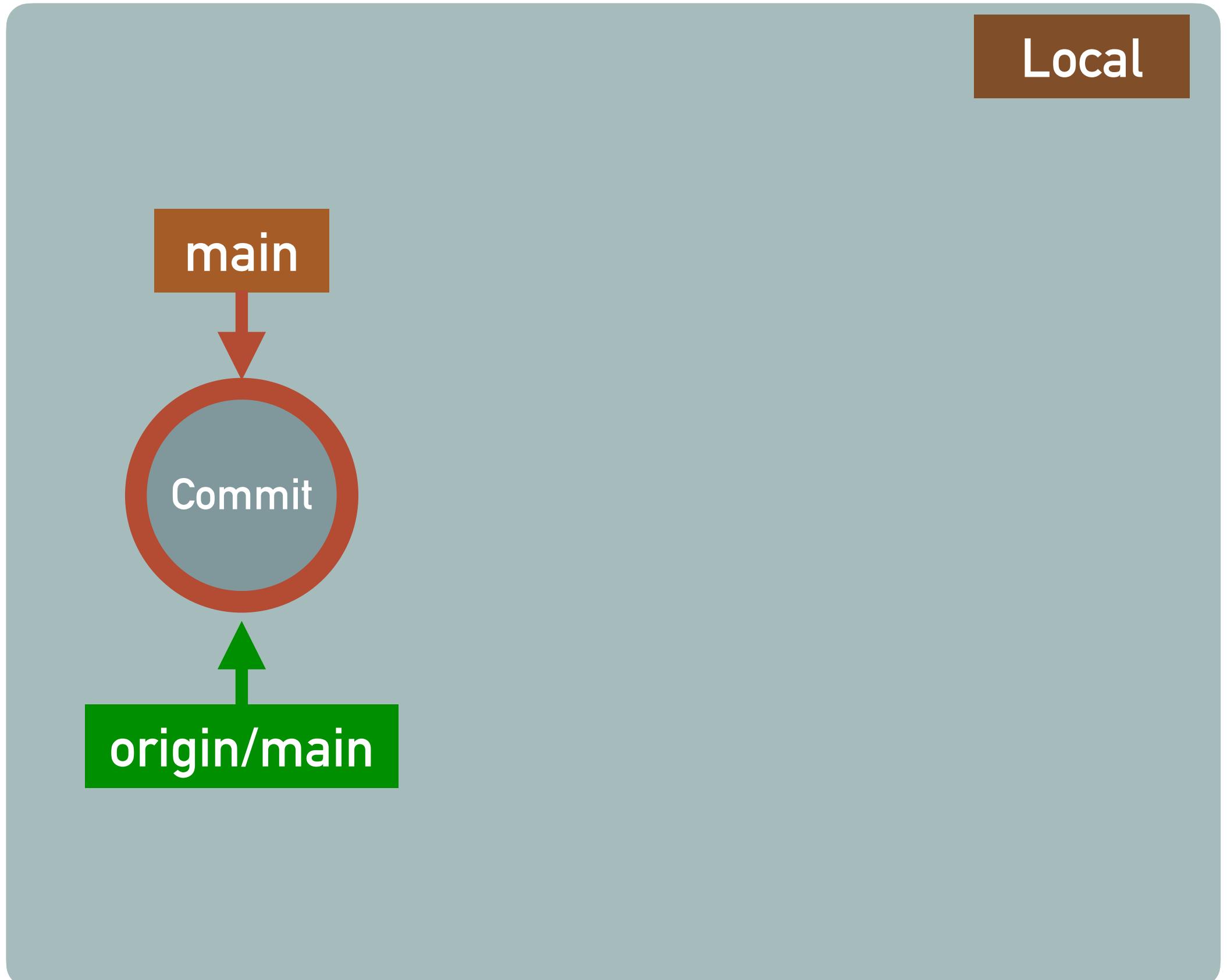


# Fetching

Your local repository is not automatically synchronised with the remote, you have to update the remote changes locally to have the latest version

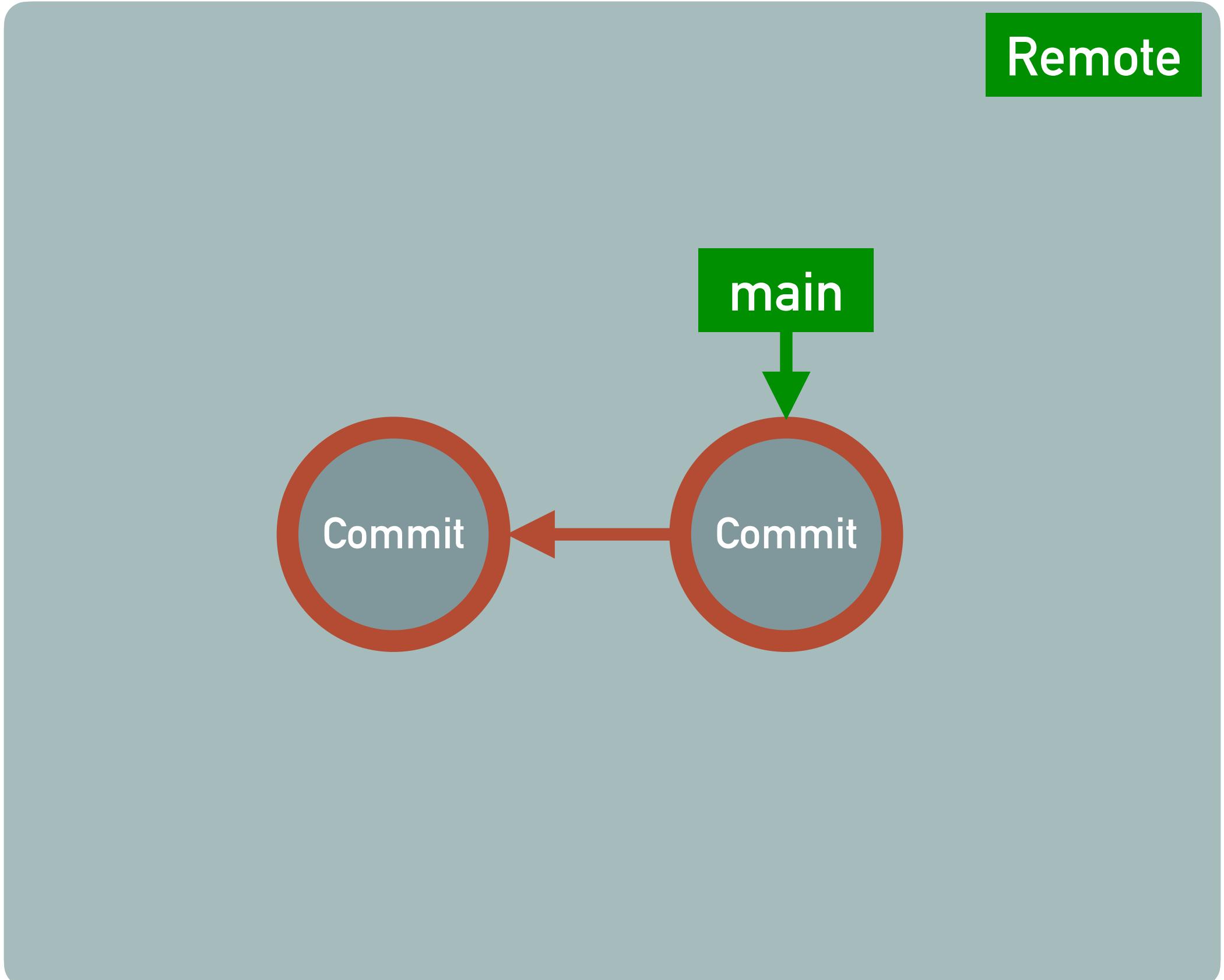
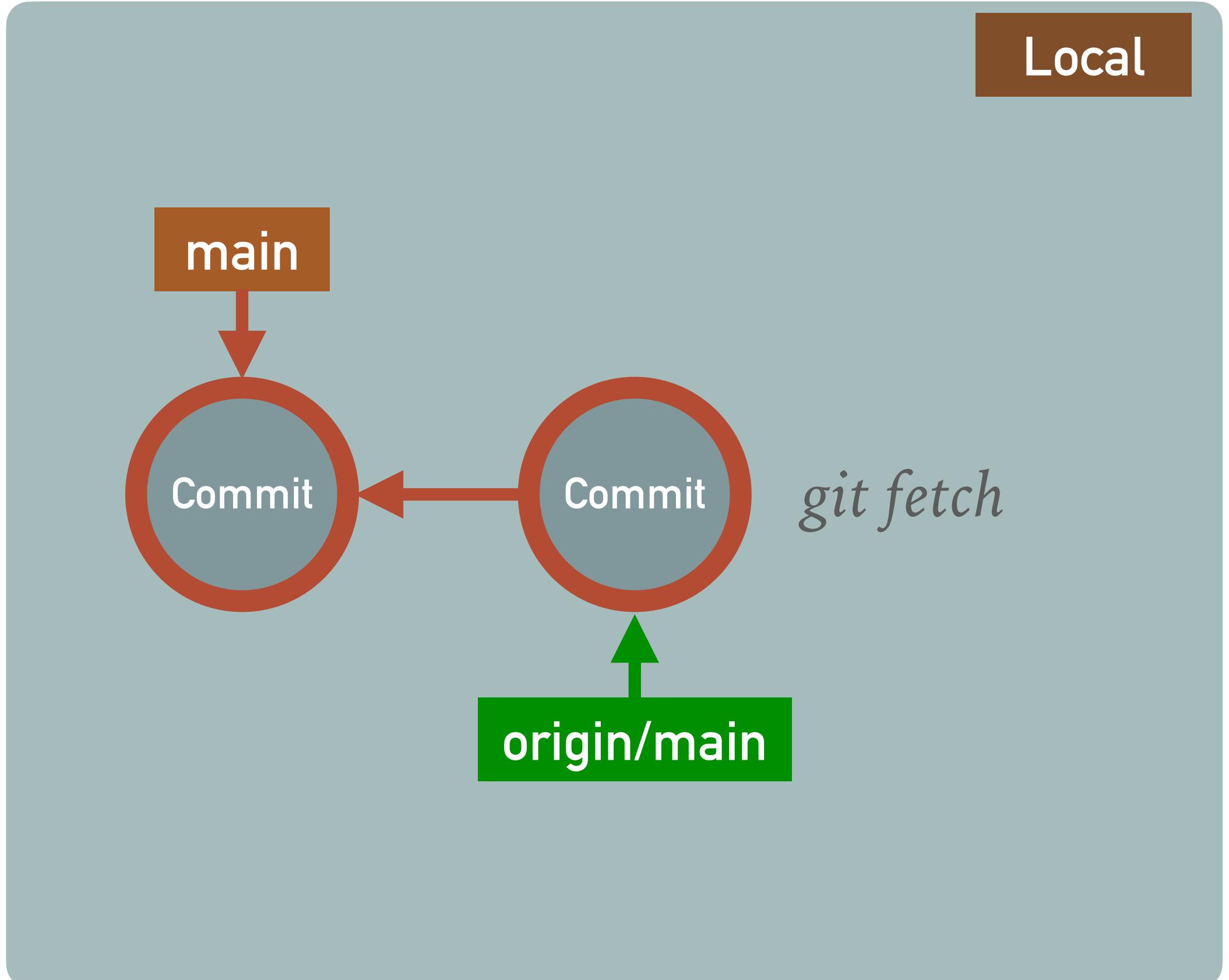


# Fetching



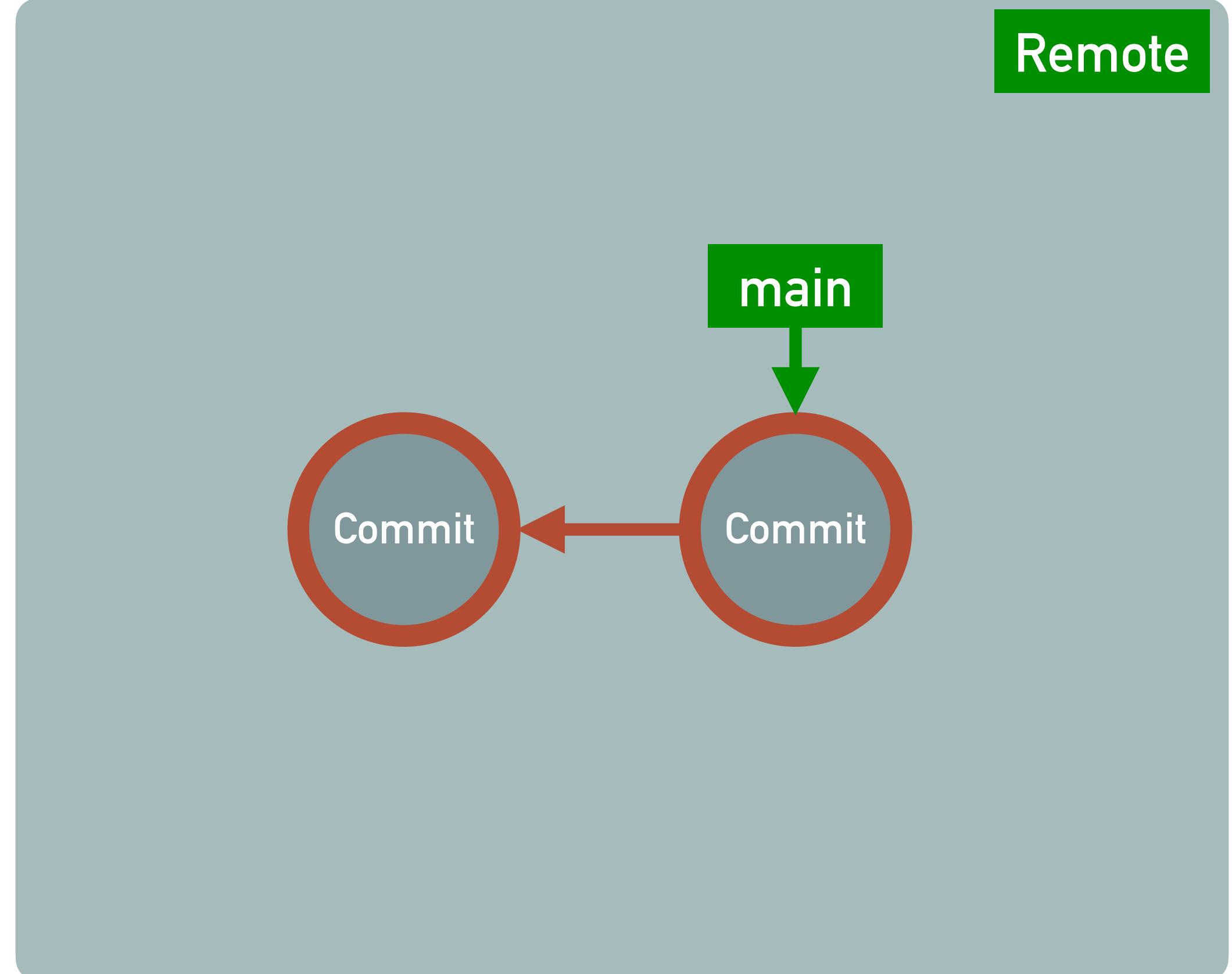
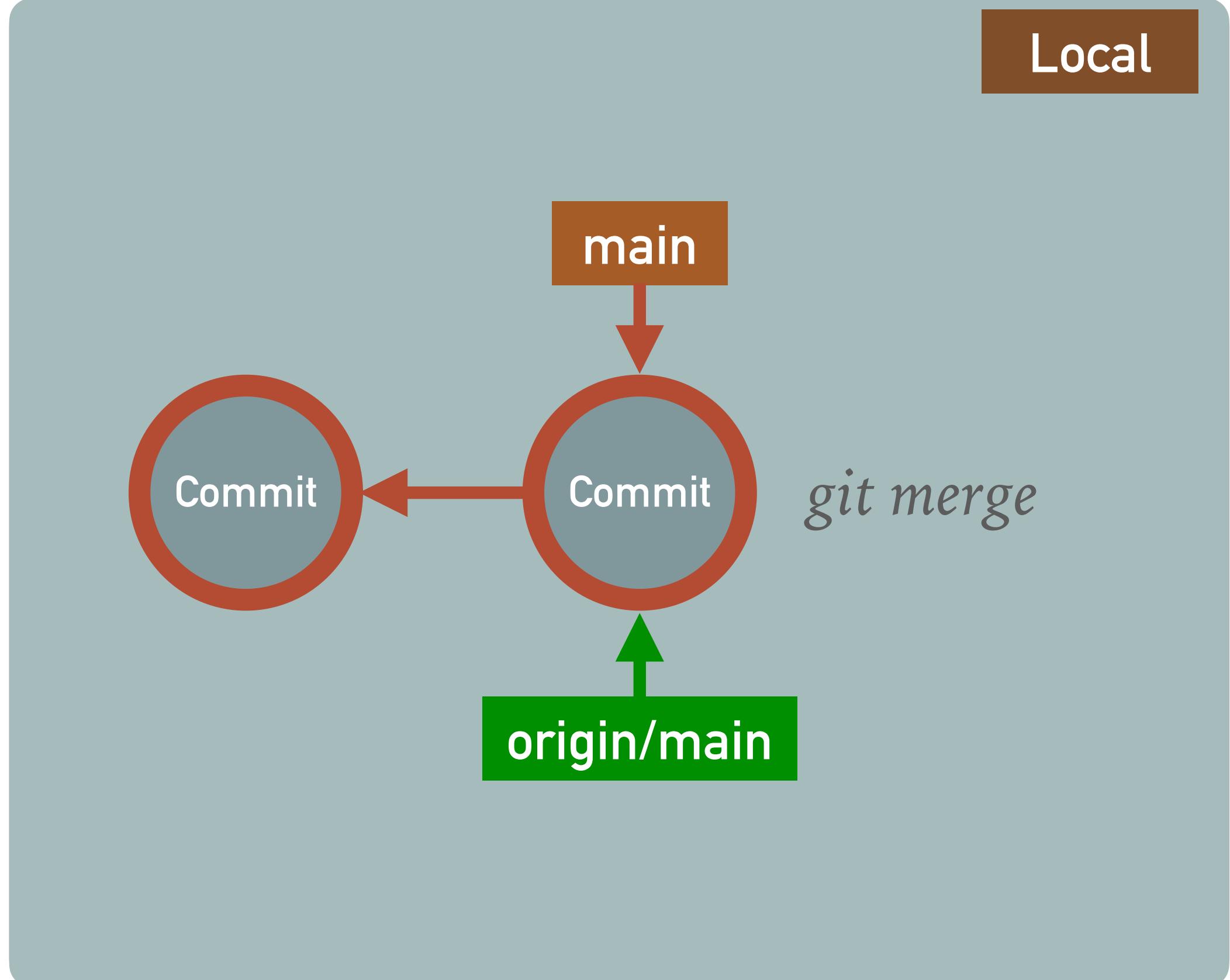
# Fetching

```
$ git fetch <branch-name> #get the remote changes locally
```



# Fetching

```
$ git merge origin/main
```



# Git pull

---

We can do both in a single command :  
git pull = git fetch + get merge + create a new commit

```
$ git pull
```

# Git pull

- Simuler un push dans github en modifiant sur l'interface github comme suit :
- Faire un pull ensuite en local

The screenshot shows the GitHub interface for the repository "AmineFrj / my-repo". The "Code" tab is selected. At the top, it shows "main" branch, 1 branch, 0 tags. Below is a list of commits:

- AmineFrj Add README. (912ffa1, 11 minutes ago)
- README.md (Add README, 11 minutes ago)
- main2. (add new comment 2, 4 hours ago)
- requirements.txt (Add new print, 9 hours ago)

Below the commits, there is a file editor for "README.md" with the content "My readme". A red arrow points to the "main2." commit in the list.

The screenshot shows the GitHub interface for the repository "AmineFrj / my-repo". The "Code" tab is selected. It shows the "my-repo / README.md" file with the content "My readme". A red arrow points to the "Blame" button in the toolbar below the file content.

At the bottom right, a "Commit changes" dialog is open:

- Input field: "Update README"
- Text area: "Add an optional extended description..."
- Radio buttons:
  - Commit directly to the `main` branch.
  - Create a new branch for this commit and start a pull request. [Learn more about pull requests.](#)
- Buttons: "Commit changes" (green) and "Cancel" (red)

A red arrow also points to the "Commit changes" button.

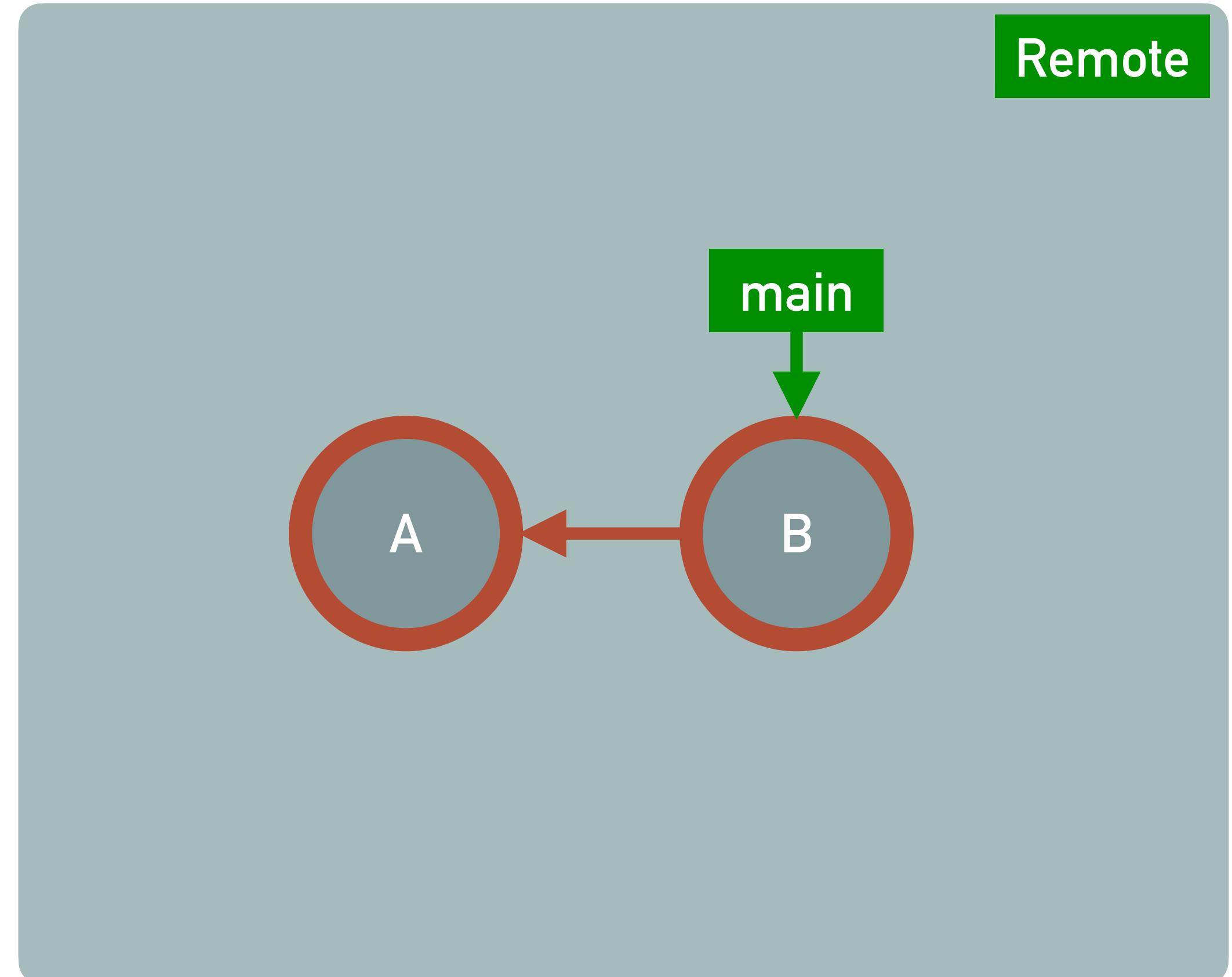
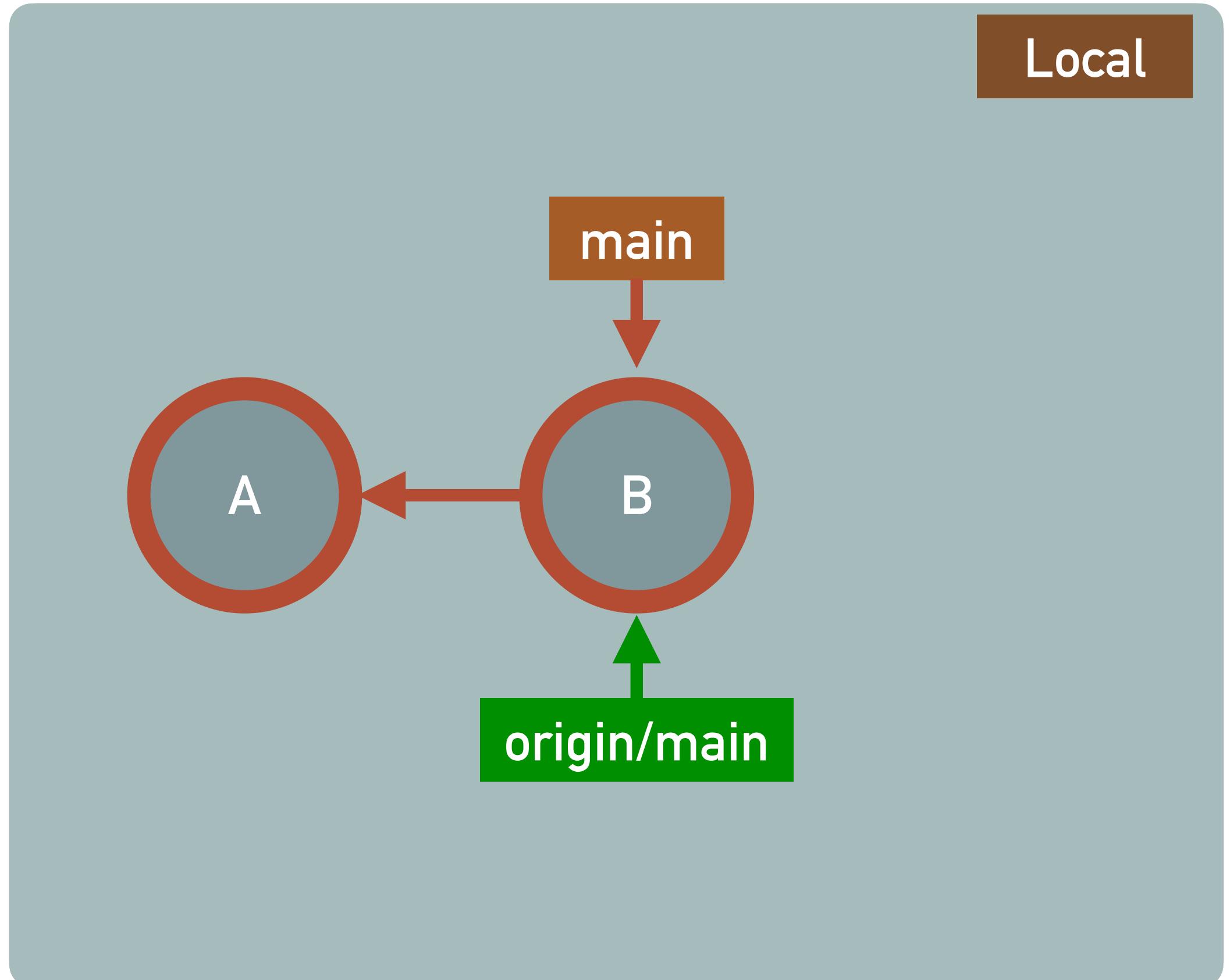
# GIT PUSH

---

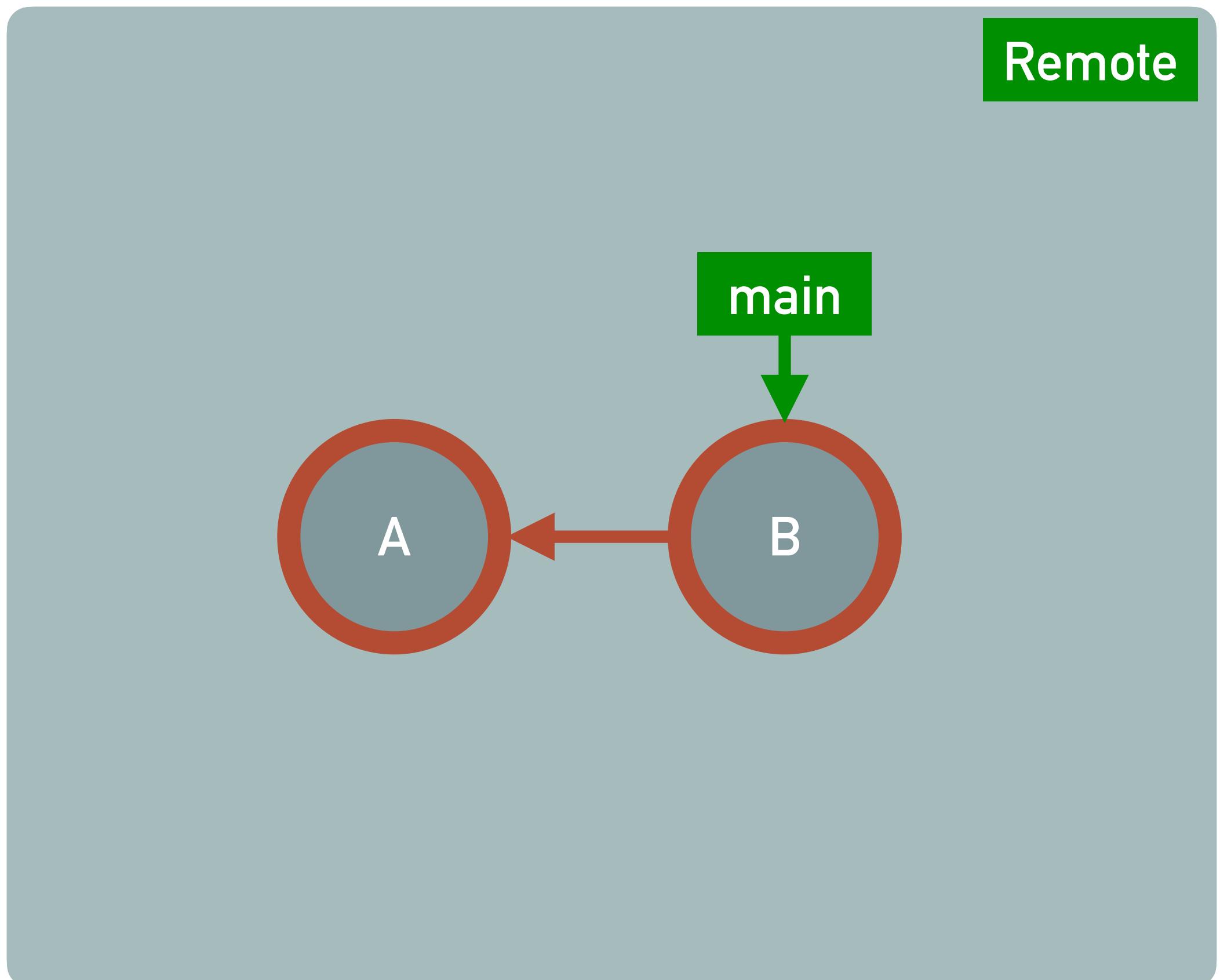
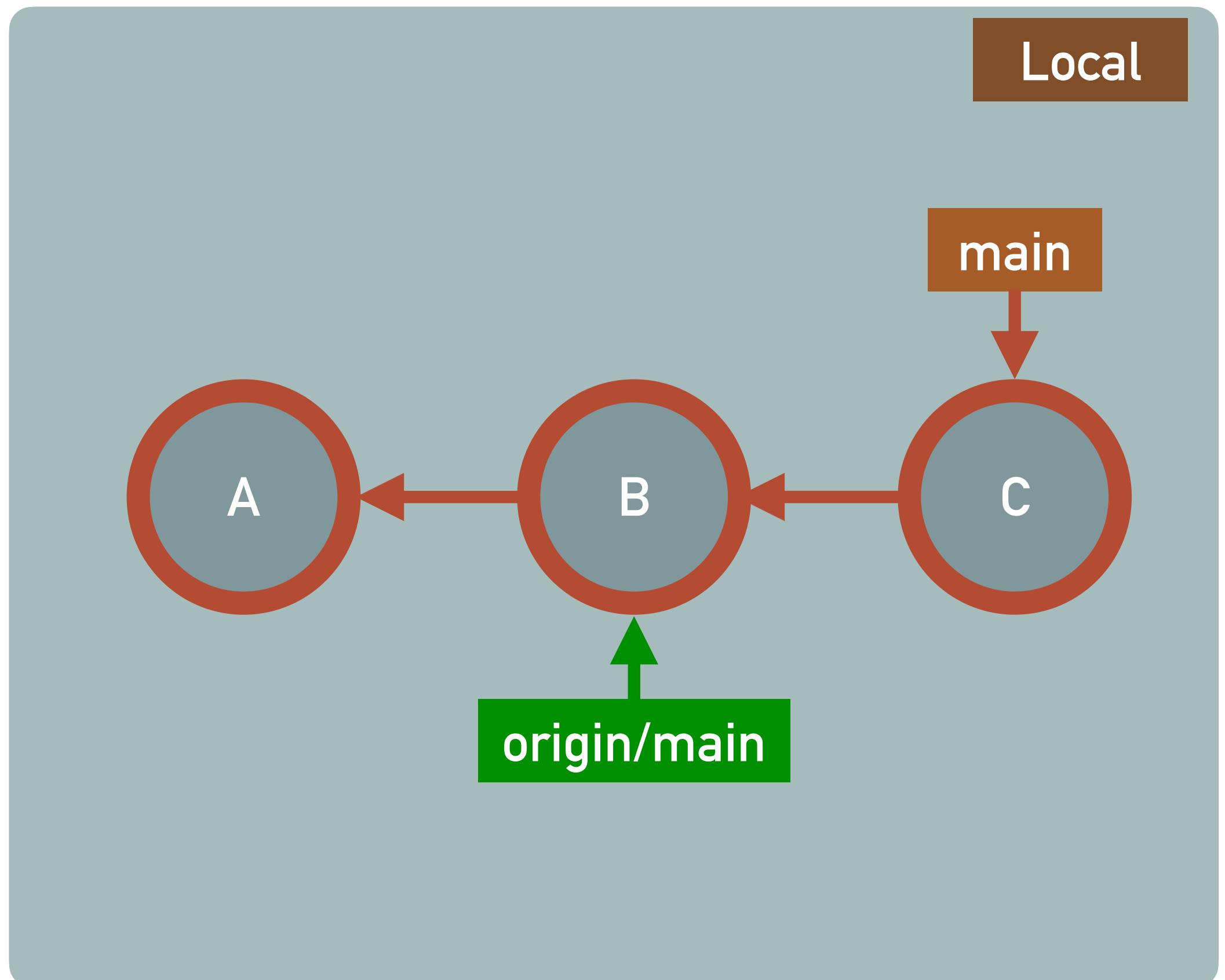


# Push

- Origin/main is considered as a separate branch
- Since it shows the remote state, it can only be modified with a push

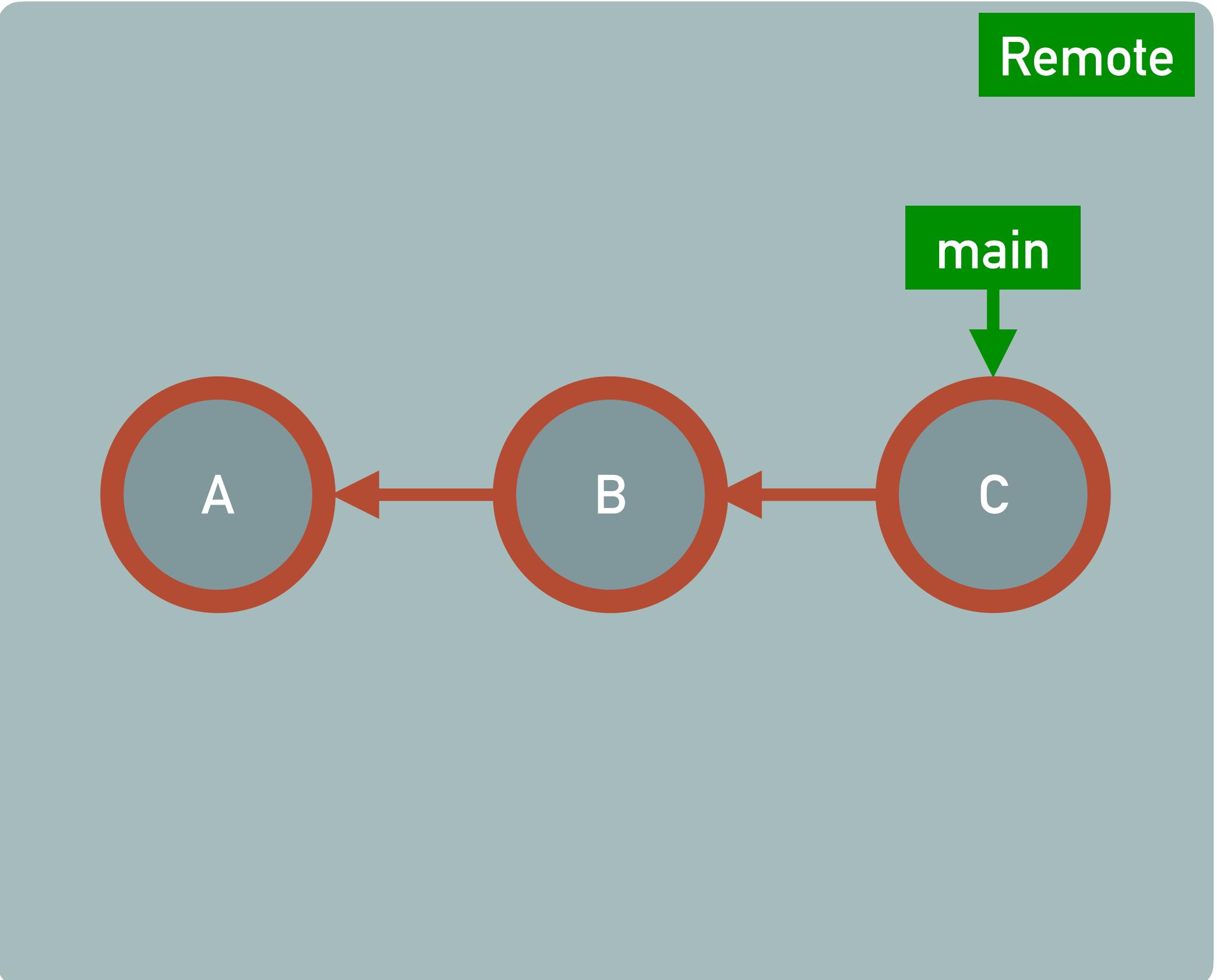
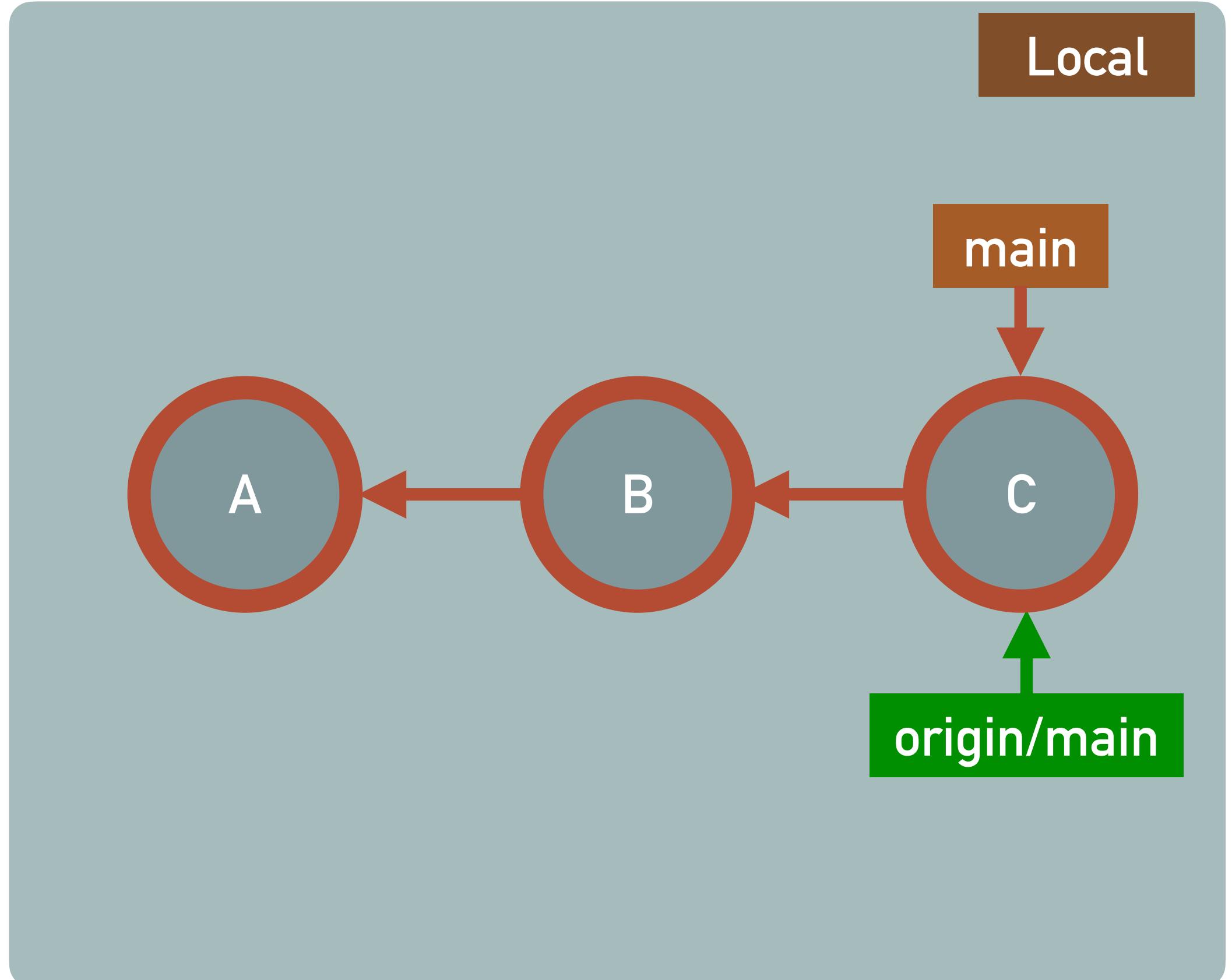


# Push



# Push

```
$ git push origin <branch> #By default push into origin and your current branch
```



# PUSH BRANCH TO REMOTE

---



# Push a branch to remote

---

When creating a branch locally you have to push it to remote

```
$ git branch -C <branch-name>
$ git push -u origin <branch-name> #push the branch to remote
$ git branch -vv #check if local branches are synchronised with remote
```

Delete a branch (locally and remotely)

```
$ git push -d origin <branch-name> #delete the remote branch
$ git switch main
$ git branch -d <branch-name>
```

# FETCH A REMOTE BRANCH LOCALLY

---



# Setup a track remote branch

---

- ◆ In order to work on a new remote branch you have to fetch it and then create a new one locally and make it track the remote one

```
$ git fetch  
$ git switch -C <branch_name> origin/<branch_name>
```

# MERGE PULL REQUESTS

---

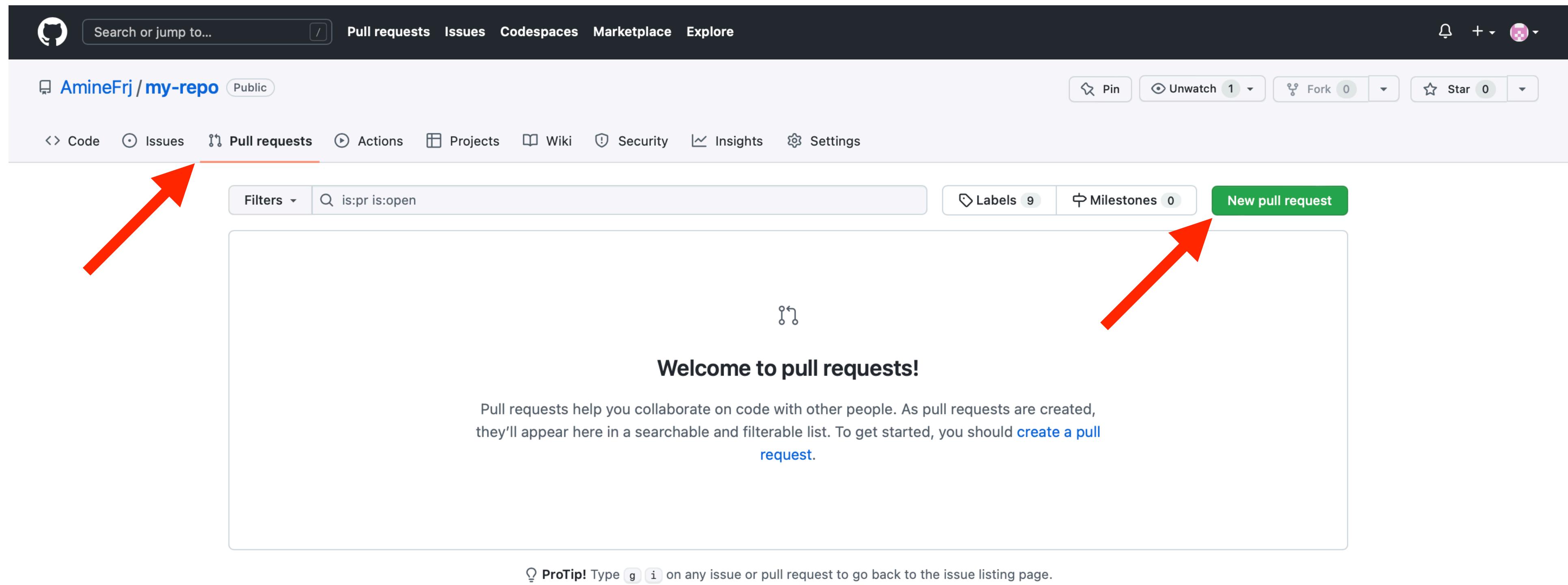


# Why do we need merge pull requests

---

- ◆ In real world projects developers don't merge their branches directly to the main branch, otherwise everybody overwrites each other work !
- ◆ We do what we call a ***pull request***. It simply means that you ask a ***reviewer*** to review your changes and make sure that your changes passed the ***tests*** and doesn't cause problems to the main project and then ***approves*** your merge.

# Git merge pull request



© 2022 GitHub, Inc.

Terms

Privacy

Security

Status

Docs

Contact GitHub

Pricing

API

Training

Blog

About

# Git merge pull request

The screenshot shows the GitHub interface for comparing branches. At the top, the repository 'AmineFrj / my-repo' is selected. The 'Code' tab is active. In the center, a comparison between 'base: main' and 'compare: pre-processing' is shown. A green checkmark indicates 'Able to merge'. Red arrows point to the 'base: main' dropdown and the 'compare: pre-processing' dropdown. Below this, a message encourages discussion and review. A green 'Create pull request' button is highlighted with a red arrow. Summary statistics show 1 commit, 1 file changed, and 1 contributor. A commit log for 'Add preprocessing' by 'AmineFrj' is displayed. At the bottom, a diff view shows the addition of 'import numpy' to 'pre\_processing.py'. Red arrows also point to the 'Split' and 'Unified' buttons at the bottom right of the diff view.

Comparing changes

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also [compare across forks](#).

base: main ▾ ← compare: pre-processing ▾ ✓ Able to merge. These branches can be automatically merged.

Discuss and review the changes in this comparison with others. [Learn about pull requests](#)

Create pull request

-o 1 commit

1 file changed

1 contributor

Commits on Nov 29, 2022

Add preprocessing

AmineFrj committed 11 hours ago

b1f9c26

Show 1 changed file with 1 addition and 0 deletions.

1 pre\_processing.py

... @@ -0,0 +1 @@ 1 + import numpy

Split Unified

# Git merge pull request

The screenshot shows a GitHub pull request page for a repository named "AmineFrj / my-repo". The pull request is titled "Add preprocessing #1" and is currently open, merging 1 commit from the "pre-processing" branch into the "main" branch. The commit message is "Add preprocessing". The pull request has 0 reviews, 0 assignees, and no labels or projects assigned. It is still in progress. A red arrow points to the "Merge pull request" button at the bottom of the main content area. Another red arrow points to the "Reviewers" section on the right, which is currently empty. A red box highlights the "Continuous integration has not been set up" section.

Add preprocessing #1

Open AmineFrj wants to merge 1 commit into `main` from `pre-processing`

Conversation 0 Commits 1 Checks 0 Files changed 1 +1 -0

AmineFrj commented 1 minute ago  
No description provided.

b1f9c26 Add preprocessing

Add more commits by pushing to the `pre-processing` branch on [AmineFrj/my-repo](#).

Continuous integration has not been set up GitHub Actions and several other apps can be used to automatically catch bugs and enforce style.

This branch has no conflicts with the base branch Merging can be performed automatically.

Merge pull request You can also [open this in GitHub Desktop](#) or view [command line instructions](#).

Reviewers No reviews Still in progress? Convert to draft

Assignees No one—assign yourself

Labels None yet

Projects None yet

Milestone No milestone

Development Successfully merging this pull request may close these issues.  
None yet

Notifications Customize Unsubscribe

# Git merge pull request

The screenshot shows a GitHub pull request page for a repository named "Add preprocessing #1". The pull request has been merged by "AmineFrj" into the "main" branch. The merge commit hash is "b1f9c26". The right sidebar contains merge options:

- Reviewers: No reviews
- Assignees: No one—assign yourself
- Labels: None yet
- Projects: None yet
- Milestone: No milestone
- Development: Successfully merging this pull request may close these issues.
- Notifications: Customize, Unsubscribe

A red arrow points from the "Delete branch" button in the merge options to the "Delete branch" button in the main pull request interface. A tooltip on the "Delete branch" button states: "The branch will be removed from the server only".

**\$ git remote prune origin**

The branch will be removed  
from the server only

To update the remote branches  
on local use

# TAGS

---



# Tags

Add a tag to the last commit

```
$ git tag <tag name>
$ git push origin <tag name>
$ git tag --delete <tag name>
```

The screenshot shows a GitHub repository page for 'AmineFrj / my-repo'. The 'Code' tab is selected. In the top navigation bar, there are links for Pull requests, Issues, Codespaces, Marketplace, Explore, and a user icon. Below the navigation bar, there are buttons for Pin, Unwatch (1), Fork (0), and Star (0). The main content area shows a 'Releases' tab followed by a 'Tags' tab, which is highlighted with a blue background. A single tag named 'v0' is listed, with details: '1 hour ago', a commit hash 'a1dd4f3', and file formats 'zip' and 'tar.gz'. A red arrow points to the 'v0' tag. At the bottom of the page, there is a footer with links for GitHub, Inc., Terms, Privacy, Security, Status, Docs, Contact GitHub, Pricing, API, Training, Blog, and About.

# Best practices

---

- ◆ Use pull request instead of merge
- ◆ Pull latest changes from remote main and create a branch