

Git & GitHub

本日のアジェンダ

- Gitについて
- addとcommit
- branchについて
- mergeについて
- GitHubについて
- pushについて
- Bootstrapの使い方

インストール

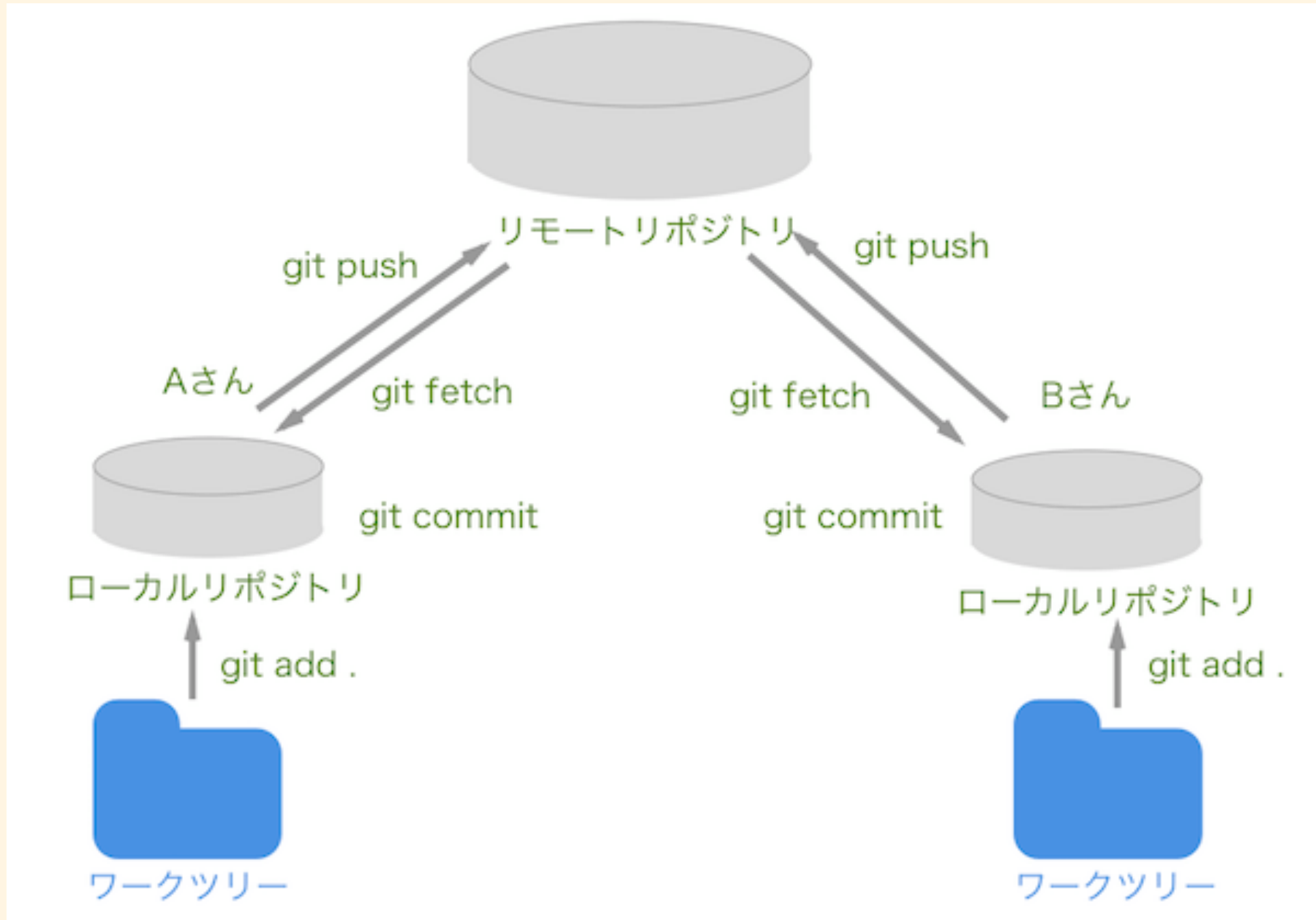
データベースの授業で使用

WindowsはXAMPP

MacはMAMP

Git

分散型バージョン管理システム



git config

globalな状態でユーザー名とメールアドレスを登録

ユーザー名登録

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

メールアドレス登録

```
git config --global user.email "name@example.co.jp"
```

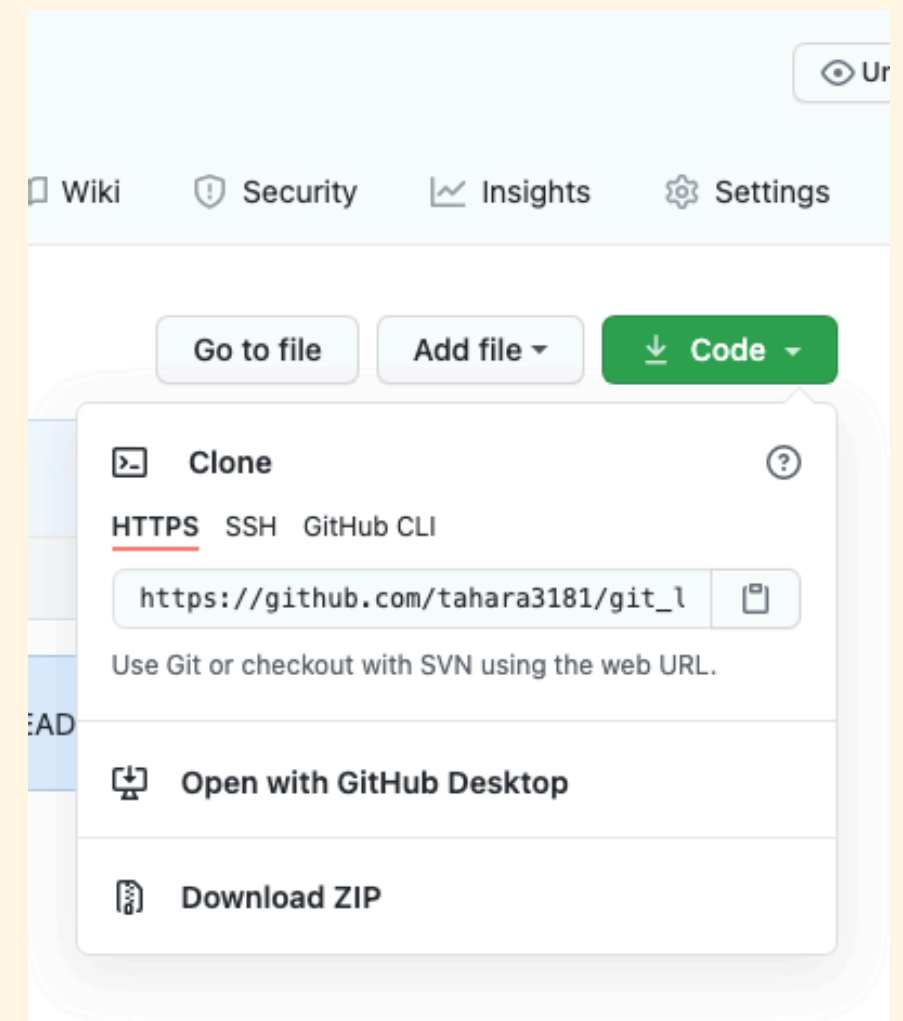
git clone

sample page

https://github.com/tahara3181/git_lesson

サンプルをclone

```
git clone https://github.com/tahara3181/git_lesson.git
```



今回はcloneしたフォルダ内の.gitフォルダを削除します。

git init

ローカルリポジトリ作成

```
git init
```

リポジトリ (repository) とは、ソースコードや変更履歴、コメントなどを一括して保管する場所です。

基本的な流れ

ステージに追加の準備

```
git add .
```

ステージに追加する

```
git commit
```


課題

前回作成フォームのファイルが入ったフォルダをgit_lessonという名前として、そのフォルダをGit管理するようにリポジトリを作成してください。

名前:

メールアドレス:

性別
☐ 男 ☐ 女

好きな食べ物は何ですか
☐ ラーメン ☐ カレー ☐ 寿司

感想を書いてください

vs-code
open in browser

Git手順

1. cdコマンドでgit_lessonフォルダに
カレントディレクトリを移動
2. ローカルリポジトリ作成
3. 対象ファイルをステージに移動準備
4. コミット

Git手順

cdコマンドでフォームのフォルダに
カレントディレクトリを移動

```
cd git_lesson
```

Git手順

ローカルリポジトリ作成

```
git init
```

Git手順

対象ファイルをステージに
移動準備

```
git add .
```

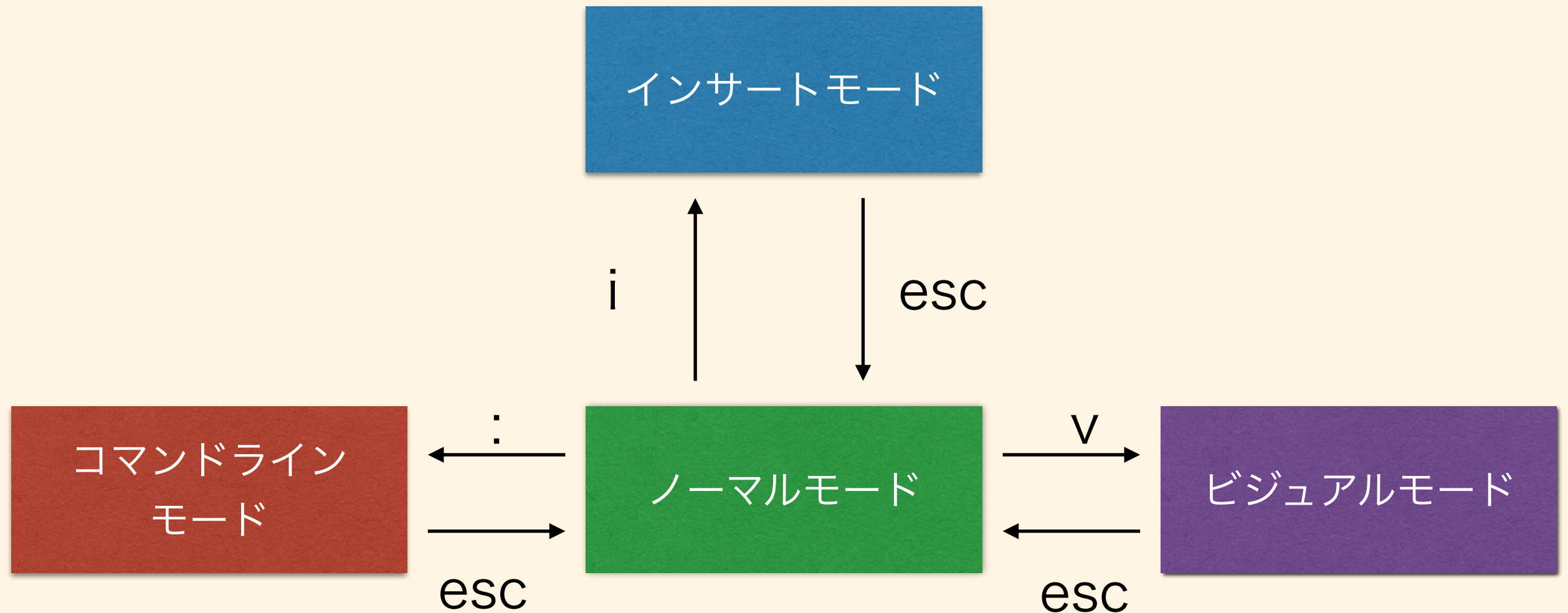
「.」は全てのファイルという意味

Git手順

コミット

```
git commit
```

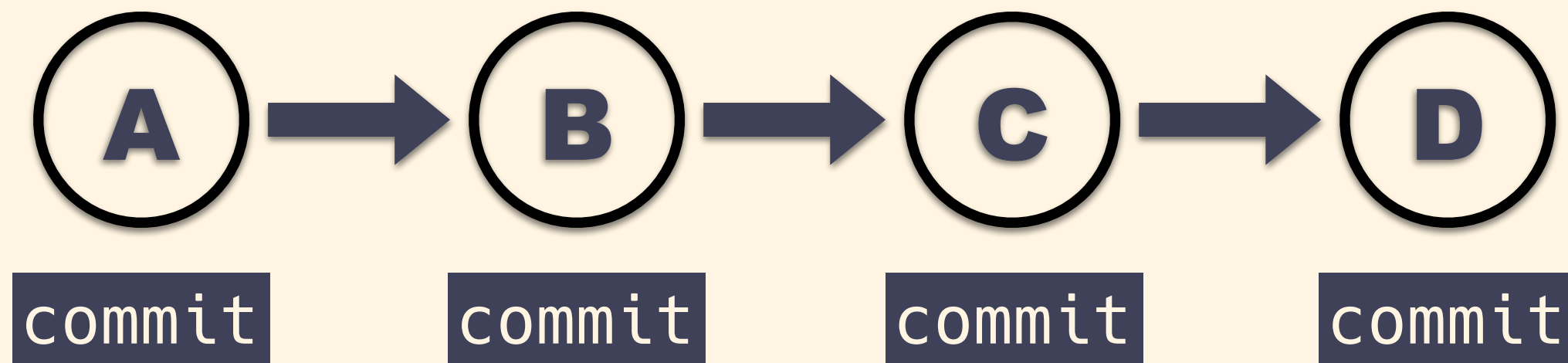
vim



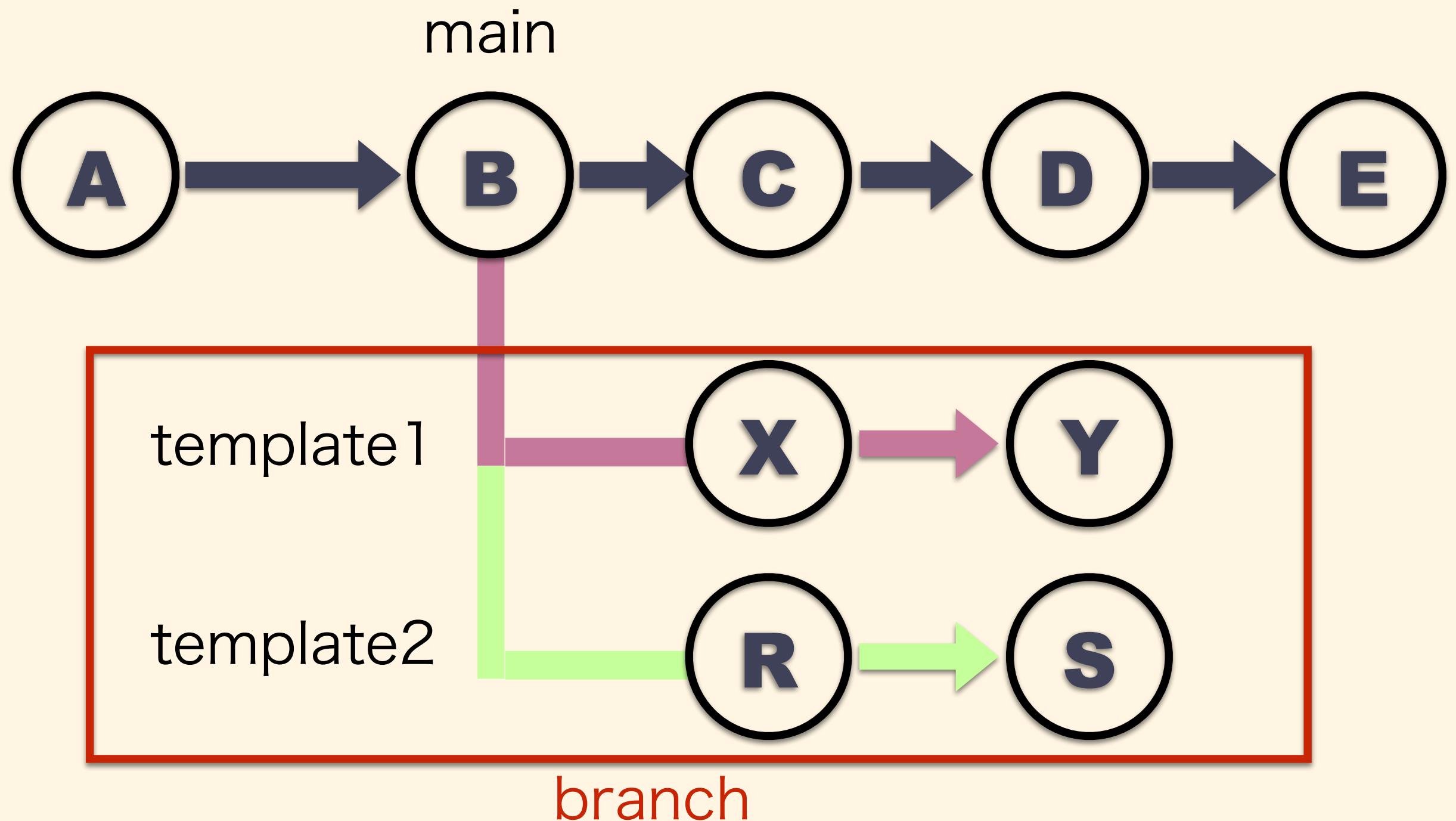
編集 : iキー
保存 : escキー
保存は : wq

main(master)

mainブランチ



branch



branch

- ・ブランチ作成

```
git branch <ブランチ名>
```

- ・ブランチ一覧

```
git branch
```

- ・ブランチ切り替え

```
git checkout <既存ブランチ名>
```

ブランチを新規作成して切り替える場合

```
git checkout -b <ブランチ名>
```

課題

新しいbranchを作成してcss1としてください。
その後、mainブランチからcss1ブランチに
チェックアウトしてください。

手順

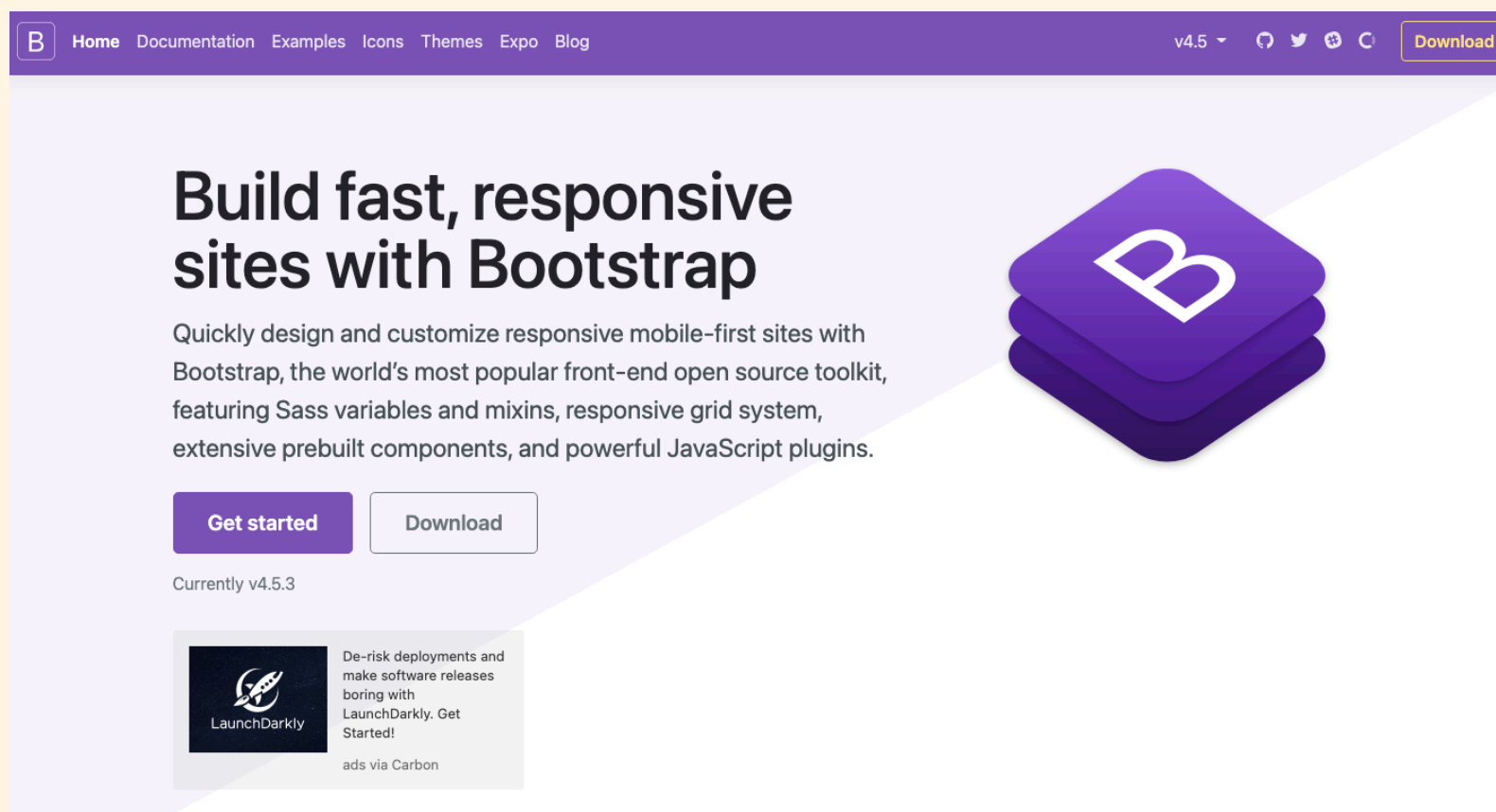
新しいbranchを作成してcss1
として、css1branchにチェッ
クアウト

```
git checkout -b css1
```

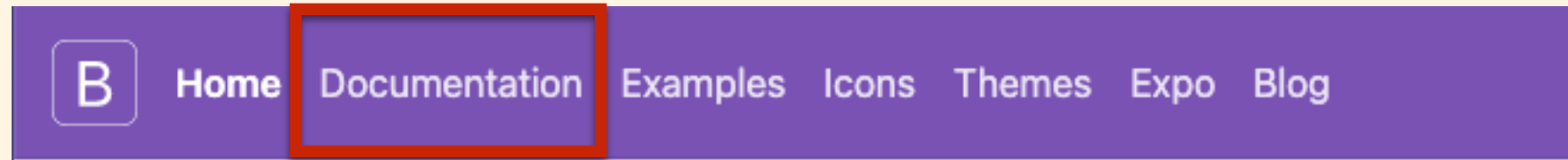
Bootstrap

CSSフレームワーク

<https://getbootstrap.com/>



Bootstrapリンク設定



- ・ グローバルナビの「Documentation」を選択
- ・ CSSコードをHTMLのhead領域にコピー＆ペースト
- ・ BundleコードをHTMLの</body>に直前コピー＆ペースト

CSS

Copy-paste the stylesheet `<link>` into your `<head>` before all other stylesheets to load our CSS.

```
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.5.3/dist/css/bootstrap.min.css">
```

Copy

Bundle

Include everything you need in one script with our bundle. Our `bootstrap.bundle.js` and `bootstrap.bundle.min.js` include `Popper`, but not `jQuery`. For more information about what's included in Bootstrap, please see our [contents](#) section.

```
<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js" integrity="sha384-DfXdz2htPH0lsSSs5nC">  
<script src="https://cdn.jsdelivr.net/npm/bootstrap@4.5.3/dist/js/bootstrap.bundle.min.js" integrity="
```

Copy

Jumbotron

B

Home

Documentation

Examples

Icons

Themes

Expo

Blog

v4.5

Search...

Getting started

Layout

Content

Components

Alerts

Badge

Breadcrumb

Buttons

Button group

Card

Carousel

Collapse

Dropdowns

Forms

Input group

Jumbotron

List group

Media object

Modal

Navs

Navbar

Pagination

A lightweight, flexible component that can optionally extend the entire viewport to showcase key marketing messages on your site.

Hello, world!

This is a simple hero unit, a simple jumbotron-style component for calling extra attention to featured content or information.

It uses utility classes for typography and spacing to space content out within the larger container.

[Learn more](#)

```
<div class="jumbotron">
  <h1 class="display-4">Hello, world!</h1>
  <p class="lead">This is a simple hero unit, a simple jumbotron-style component for calling extra a
  <hr class="my-4">
  <p>It uses utility classes for typography and spacing to space content out within the larger conta
  <a class="btn btn-primary btn-lg" href="#" role="button">Learn more</a>
</div>
```

Copy

Forms



[Home](#) [Documentation](#) [Examples](#) [Icons](#) [Themes](#) [Expo](#) [Blog](#)

v4.5 ▾



Getting started

Layout

Content

Components

Alerts

Badge

Breadcrumb

Buttons

Button group

Card

Carousel

Collapse

Dropdowns

Forms

Input group

Jumbotron

List group

Media object

Modal

Navs

Navbar

Pagination

Poppers

Email address

We'll never share your email with anyone else.

Password

☐ Check me out

Submit

Copy

```
<form>
  <div class="form-group">
    <label for="exampleInputEmail1">Email address</label>
    <input type="email" class="form-control" id="exampleInputEmail1" aria-describedby="emailHelp">
    <small id="emailHelp" class="form-text text-muted">We'll never share your email with anyone else.
  </div>
  <div class="form-group">
    <label for="exampleInputPassword1">Password</label>
    <input type="password" class="form-control" id="exampleInputPassword1">
  </div>
  <div class="form-group form-check">
    <input type="checkbox" class="form-check-input" id="exampleCheck1">
    <label class="form-check-label" for="exampleCheck1">Check me out</label>
  </div>
  <button type="submit" class="btn btn-primary">Submit</button>
</form>
```


課題

1. branch css1の状態、Bootstrapを使ってformの見栄えをよくしてください。
2. Junbotronを使ってヘッダー領域を作成してください。
3. headの中にある<html lang="en">のlang="en"をlang="ja"に変更してください。
4. css1ブランチでaddとcommitを行ってください。
5. mainにチェックアウトしてください。
6. headの中にある<html lang="en">のlang="en"をlang="jp"に変更してください。
7. mainブランチでaddとcommitを行ってください。

merge

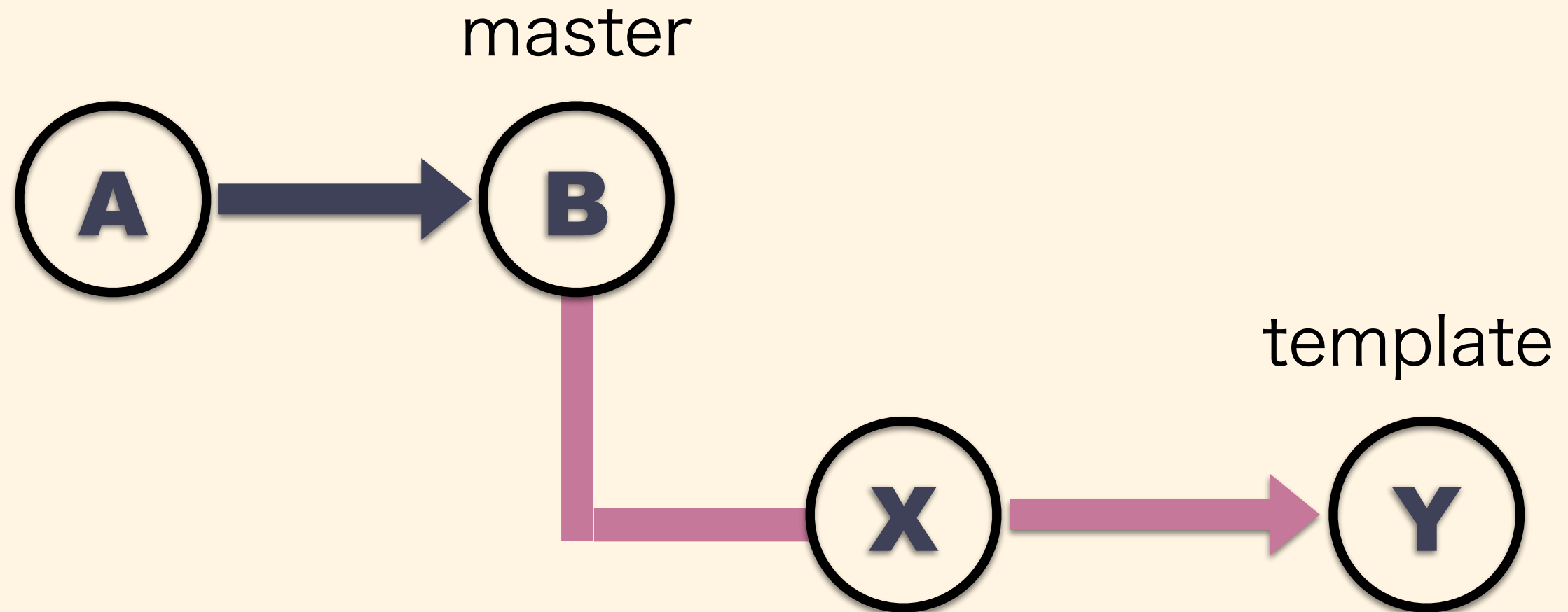
- ・ローカルリポジトリのブランチをマージ

```
git merge <ブランチ名>
```

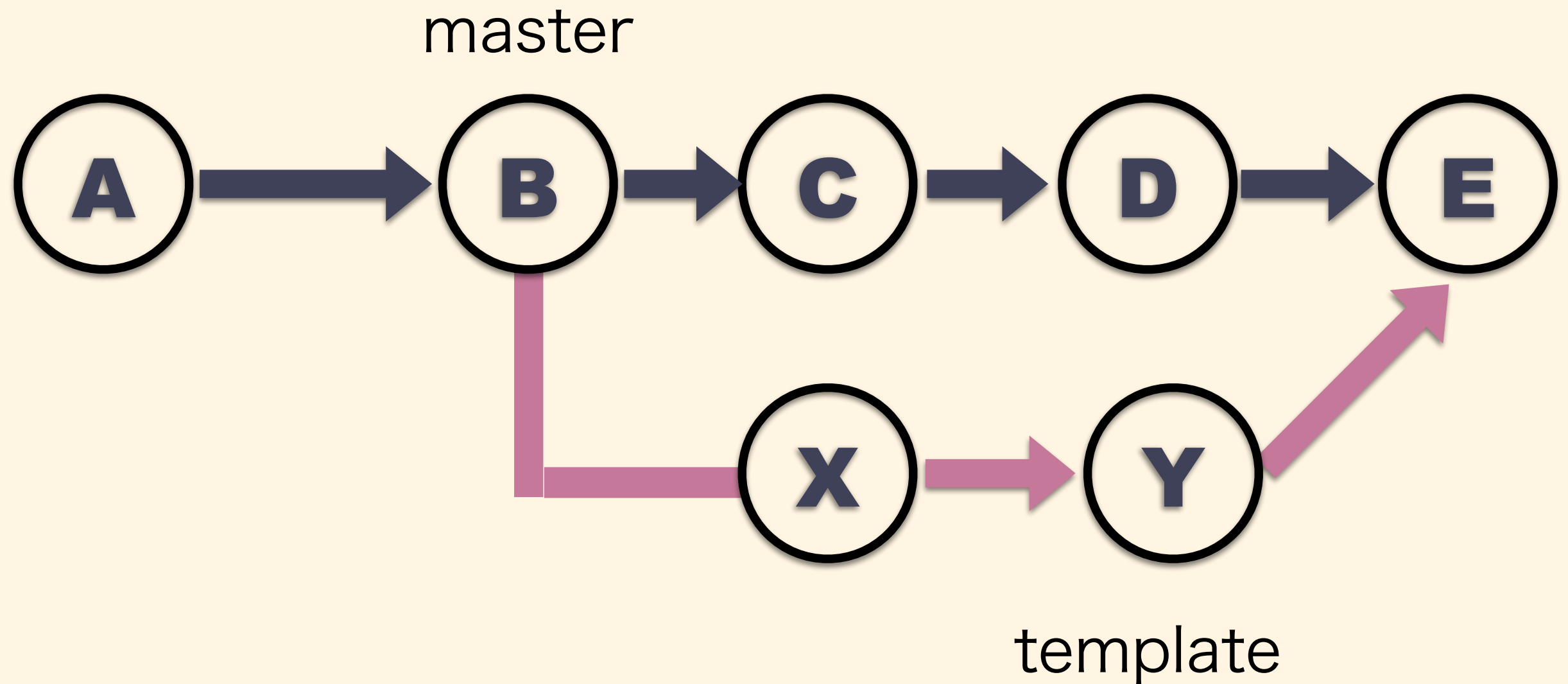
- ・リモートリポジトリのブランチとマージ

```
git merge <リモート名/ブランチ名>
```

fast-forward



Auto Merge



conflict

同じファイルに異なる編集を行ったら起こる
コンフリクト関連の事故を減らすルール

- ・複数人で同じファイルを変更しない
- ・pullやmergeする前に変更中の状態を無くしておく
(commitしておく)
- ・pullするときは、pullするブランチに移動してから
pullする
- ・コンフリクトしても慌てない

課題

1. `css1` ブランチを `main` にマージしてください。
2. コンフリクトが起こったら解決してください。

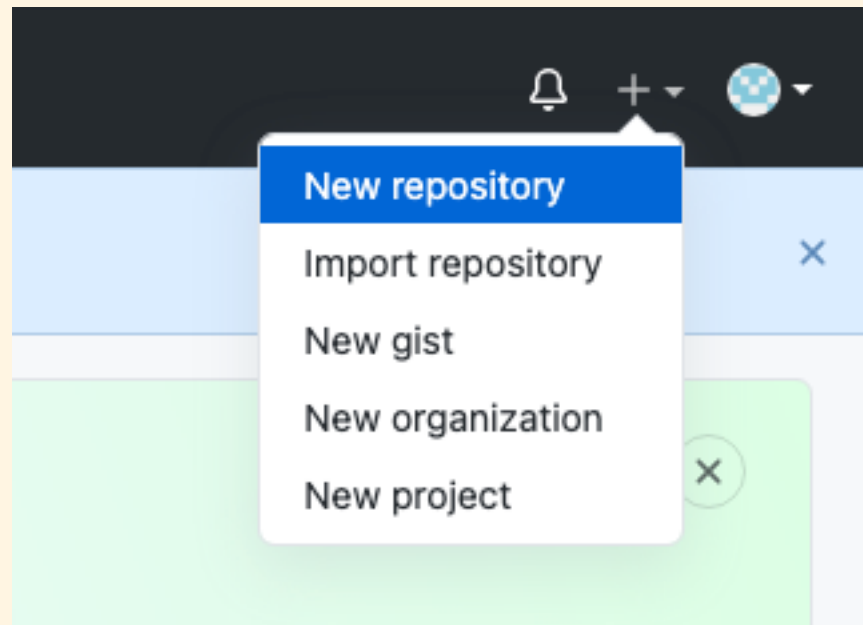
GitHub

- ・GitHubとは、ソフトウェア開発プロジェクトのためのソースコード管理サービスです。
- ・リモートリポジトリの役目を果たします。

1. GitHubでリポジトリを作成
2. ローカルリポジトリをプッシュして
リモートリポジトリへ反映

GitHub

新規リポジトリ作成



GitHub

新規リポジトリ作成

Create a new repository

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

Owner *

 tahara3181 ▾

Repository name *

git_lesson ✓

Great repository names are **git_lesson** is available. Need inspiration? How about **friendly-winner**?

Description (optional)



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.](#)




Choose a license

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

Create repository

GitHub

新規リポジトリ作成

 **tahara3181 / git_lesson**


Unwatch 1

Star 0

Fork

<> Code | ! Issues | 🔄 Pull requests | ▶ Actions | 📁 Projects | 📖 Wiki | 🛡 Security | 📈 Insights | ⚙ Settings

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


 Set up in Desktop

 or

HTTPS

SSH


https://github.com/tahara3181/git_lesson.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 "# git_lesson" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/tahara3181/git_lesson.git
git push -u origin main
```



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

```
git remote add origin https://github.com/tahara3181/git_lesson.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

push

master のpush

リモートに接続する設定

```
git remote add origin https://github.com/  
[アカウント]/[リポジトリ].git
```

pushは以下コマンド

```
git push -u origin master
```

便利なコマンド

```
git push -u origin HEAD
```

push

main のpush

リモートに接続する設定

```
git remote add origin https://github.com/  
[アカウント]/[リポジトリ].git
```

チェックアウト中のカレントブランチをmainにする



```
git branch -M main
```


pushは以下コマンド


```
git push -u origin main
```


GitHub


新規リポジトリ作成


 **tahara3181 / git_lesson** 


 **Code**


 Issues


 Pull requests


 Actions


 Projects


 Wiki


 Security

 Insights

 Settings


 main ▾



 1 branch


 0 tags

Go to file

Add file ▾

 **Code** ▾

 **haidoro3181** Initial commit 090fdc6 2 hours ago  1 commits

 form.html

Initial commit

2 hours ago

Help people interested in this repository understand your project by adding a README.

Add a README

fetch

・fetchしてmergeするのがPoint

```
git fetch origin
```



```
git diff HEAD origin/main
```



```
git merge origin/main
```

pull

・fetchしてmergeまで行ってくれる

```
git pull origin main
```