

# WEB基礎演習

第3回

2020年10月9日(金)

- ・ソーシャルディスタンスを保とう(できるだけ1席ずつあける)
- ・必携PCを使いましょう
- ・もし忘れた場合→181教室でのログイン
  - 2限のユーザ名: WebPro4\_1kumi
  - 3限のユーザ名: WebPro4\_2kumi
  - パスワード: kudo

# Today's Agenda

本日の内容

- 
- Web技術の**基本用語2**
  - 課題1の**アップロードと公開**: GitHub Pages
-

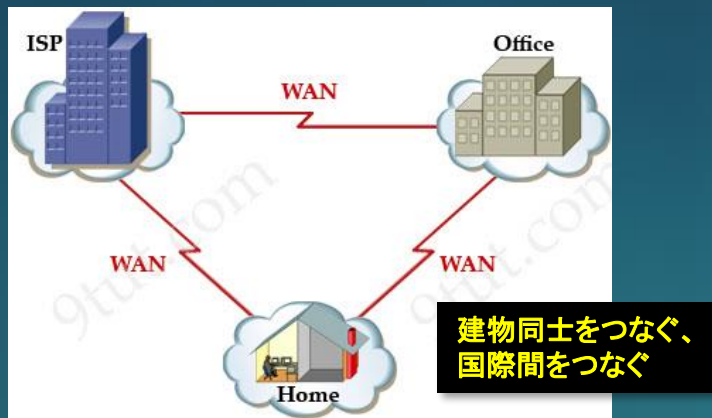
# Web技術の基本用語2

# LANとWAN

**LAN:** Local Area Networkの略。**室内や建物内**など、物理的に配線できる小規模な**範囲の機器**を接続しているネットワークのこと。

ルーターと呼ばれる機器を境にLANとWANが分かれる

**WAN:** Wide Area Networkの略。**建物同士や郊外、県外、国際**など、LAN同士を**広い範囲**で接続するネットワークのこと



# URL

- Uniform Resource Locatorの略
- インターネット上のHTMLや画像などのリソースの所在を表す文字列
- それぞれのURLは世界で1つのユニーク文字列であり、特定のリソースと1対1に結び付く

<例えば久留米工業大学HPのURL>  
<http://www.kurume-it.ac.jp/>



日本国内のある特定のサーバ(パソコン)の中に入っている、ある1つのHTMLファイルを指す

# Webサイト

- あるブランドの洋服、あるゲームの攻略など、ある特定のテーマに沿って公開されるWebページの集まり
- サイト内のWebページ同士はハイパーリンクによって結びつく



# ダウンロードとアップロード

**アップロード**： データやファイルをクライアントからサーバにコピーすること

**ダウンロード**： データやファイルをサーバからクライアントにコピーすること

Webクライアント



ファイル



アップロード



ダウンロード



ファイル



Webサーバ



# プロトコル

コンピュータ間のデータ通信を行うときの手順や約束事を定義したもの

(例えば: データはこの**圧縮形式**で...この**文字列**を付け足して...**このように分割**しながら送ろう)

## プロトコルの種類 (ほんの一部)

プロトコルの名前	用途などの説明
TCP/IP	インターネットで使われているプロトコル
FTP	<b>File Transfer</b> Protocolの略 インターネット上で <b>ファイル</b> を <b>転送</b> するときのプロトコル
HTTP	<b>Hyper Text Transfer</b> Protocolの略 <b>Web</b> で使われているプロトコル
SMTP, POP	メールに使われているプロトコル



提出課題1をアップロードしよう

# 提出課題1 : **kadai1.html**という名前で、以下のページを作ろう



タイトル

大見出し

画像(好きな画像でよい)

小見出し

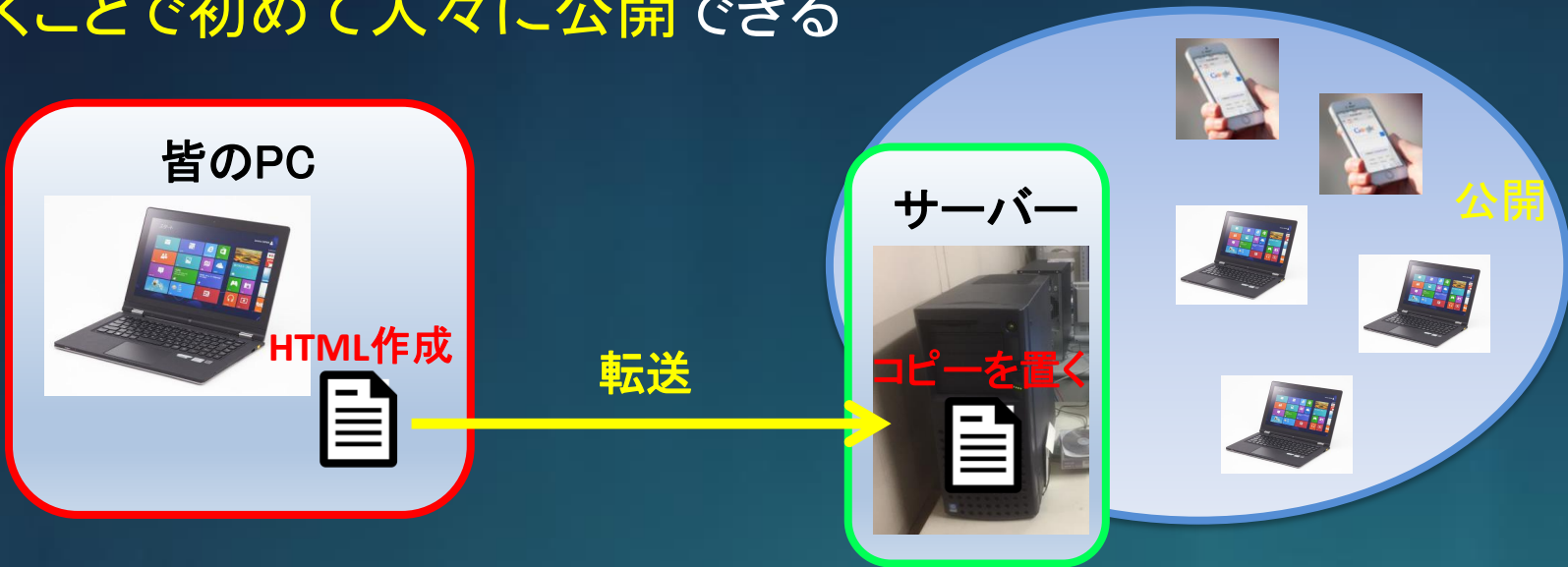
中見出し

本文はそれぞれ段落<p>タグで囲むことに注意

「こちら」に久留米工業大学HPへのリンク

# ファイルを作るだけでは足りない(再)

- Webページは、HTMLファイル等の作成の後に**サーバーに置く**ことで初めて人々に**公開**できる

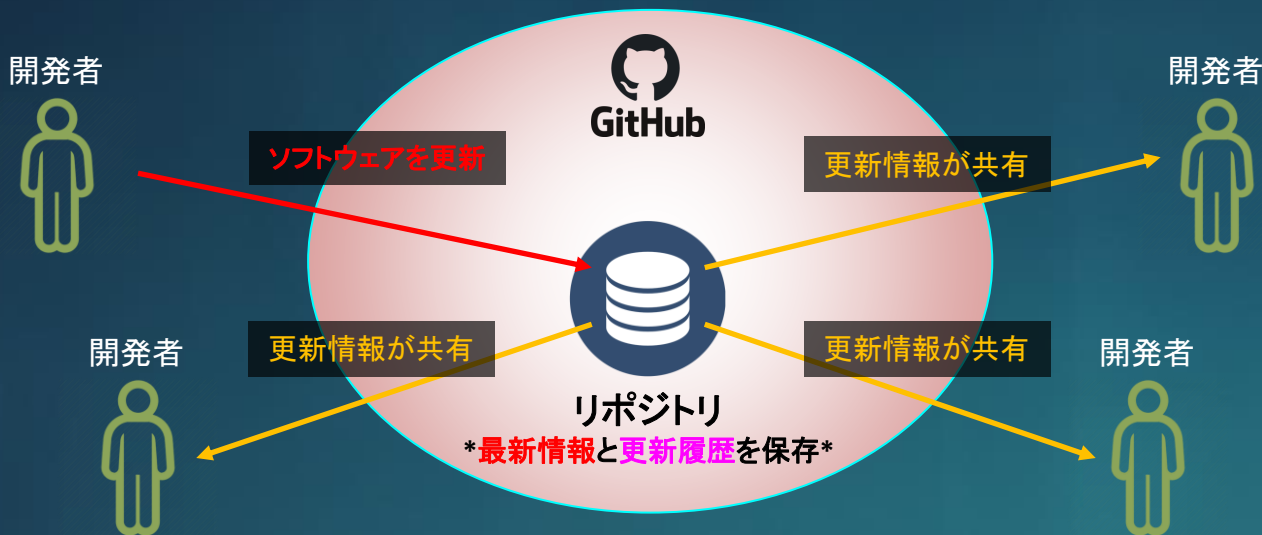


サーバーは皆のPCとハードウェア的には変わらない  
だが、自分の持つHTMLファイル等を外部に公開する設定がされている

公開サーバーとして、  
GitHub Pagesの機能を使います

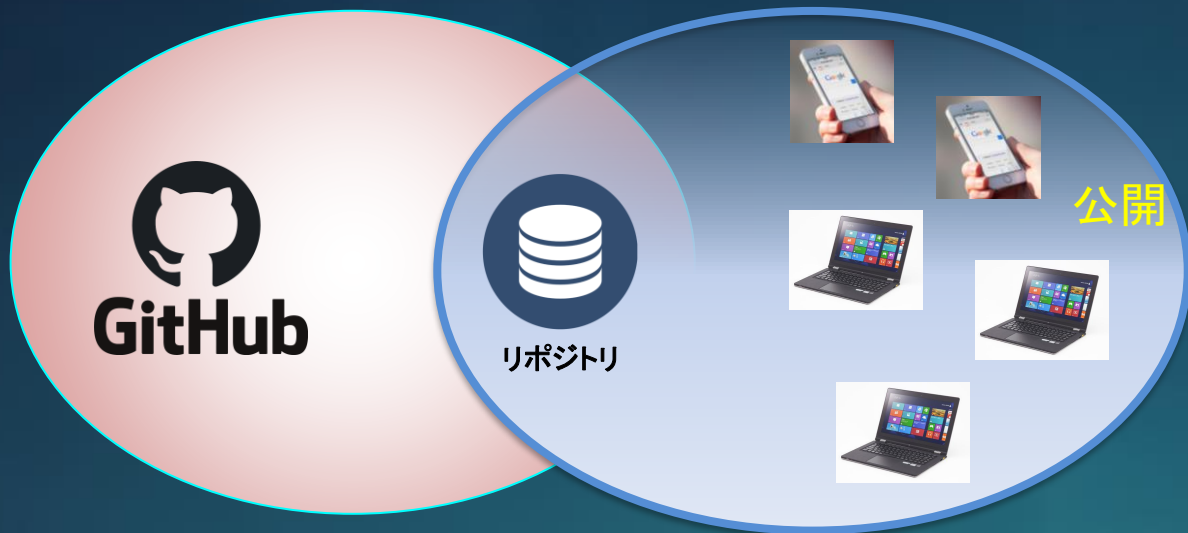
# GitHubとは

- ・ 本来は、1つのソフトウェアを多人数で開発していく時に使われる超便利なWebツール
- ・ リポジトリ(WEB上の情報格納スペース)を参照することで、複数人でも常に最新のファイルを共有しながら開発を進めていくことができる



# GitHub Pagesとは

- GitHubの機能の1つに、リポジトリをWebサーバー化できるものがある  
→ GitHub Pages



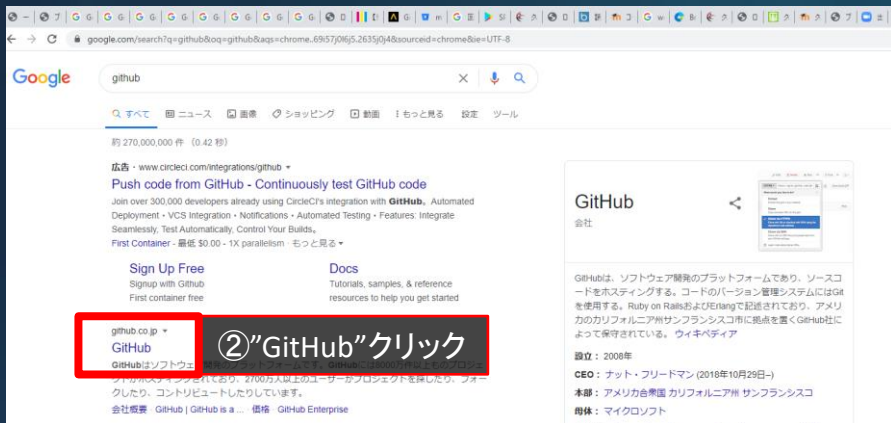
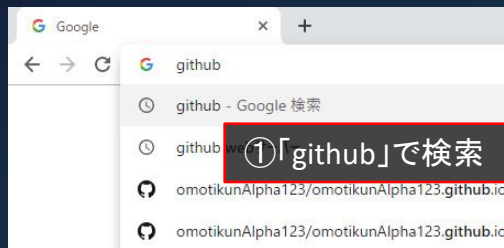
授業では、この機能を使って皆が作ったWebページを公開していきます

# 手順概要

- 1) GitHubで学校の[学籍番号文字列]のアカウントを取得 → ログイン
- 2) [学籍番号文字列].github.io という名前でリポジトリを作る
- 3) 作ったリポジトリに課題のファイルをアップロード(コミット)
- 4) Github PagesのSourceとテーマを設定
- 5) [https://\[学籍番号文字列\].github.io/kadai1.html](https://[学籍番号文字列].github.io/kadai1.html) の表示を確認

# 1) GitHubでアカウント取得 → ログイン

a) まず、ChromeでGitHubのサイトへ行こう



③GitHubのサイトが表示される





# 1) GitHubでアカウント取得 → ログイン

b) Usernameを[学籍番号の文字列]としたアカウントを作成

Username \*

e204199kt

Email address \*

e204199kt@kurume-it.ac.jp

この部分が  
一致してるのを確認

Create your account

Username \*

e204199kt

Email address \*

e204199kt@kurume-it.ac.jp

Password \*

.....

Make sure it's at least 15 characters, OR at least 8 characters including a number and a lowercase letter.  
[Learn more.](#)

Email preferences

☐ Send me occasional product updates, announcements, and offers.

Verify your account

?

Create account

この手順は特に間違えないように  
また全て半角です

① Usernameに[学籍番号の文字列]  
(メールアドレスの@の前の文字)を入れる

② Emailに学校の自分のメールアドレス入力

③ Passwordはとりあえず  
webbasic2020としよう(後で変えてもかまわない)

④ パズルを解く

⑤ クリック

# 1) GitHubでアカウント取得 → ログイン

## c) アンケートに答える

Issues Marketplace Explore

## Welcome to GitHub

Woohoo! You've joined millions of developers who are doing their best work on GitHub. Tell us what you're interested in. We'll help you get there.

①どんな仕事をしていますか？  
で学生 (Student) を選択

Software Engineer I write code	<b>Student I go to school</b>
Product Manager I write specs	UX & Design I draw interfaces
Data & Analytics I write queries	Marketing & Sales I look at charts
Teacher I educate people	Other I do my own thing

How much programming experience do you have?

None I don't program at all	A little I'm new to programming
A moderate amount I'm somewhat experienced	A lot I'm very experienced

How much programming experience do you have?

None I don't program at all	A little I'm new to programming
A moderate amount I'm somewhat experienced	A lot I'm very experienced

②プログラミングのスキルレベル、  
4つのどれか選ぶ

Learn to code	Learn Git and GitHub	Host a project (repository)
<b>Create a website with GitHub Pages</b>	Find and contribute to open source	School work and student projects

③GitHubで何をするか、  
Webサイト作ります選択でよい

I am interested in:

We'll connect you with communities and projects. For example, JavaScript.

④クリック

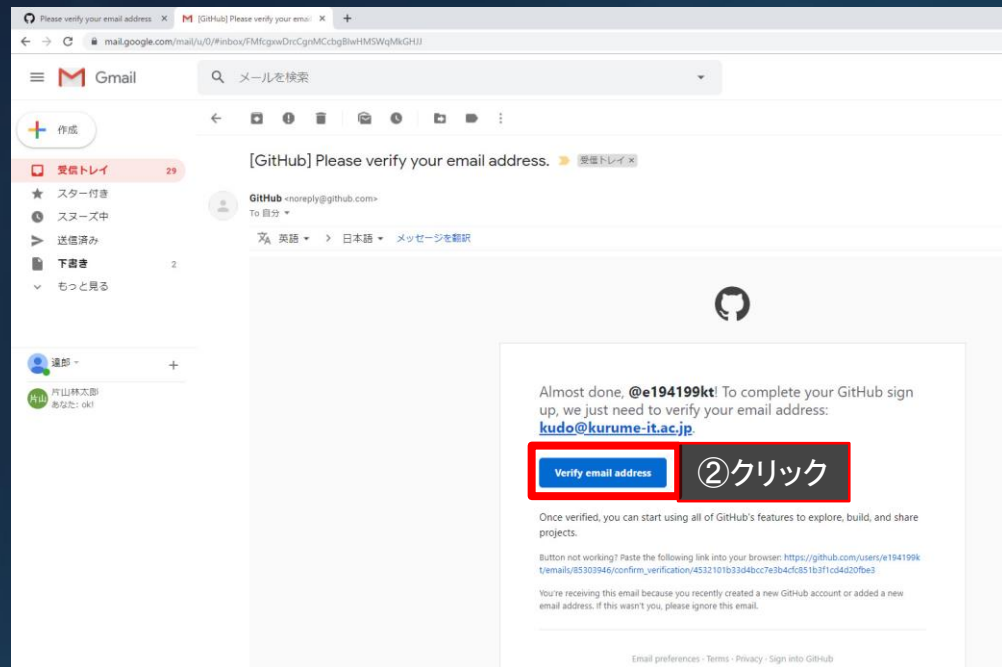
**Complete setup**

Skip this step

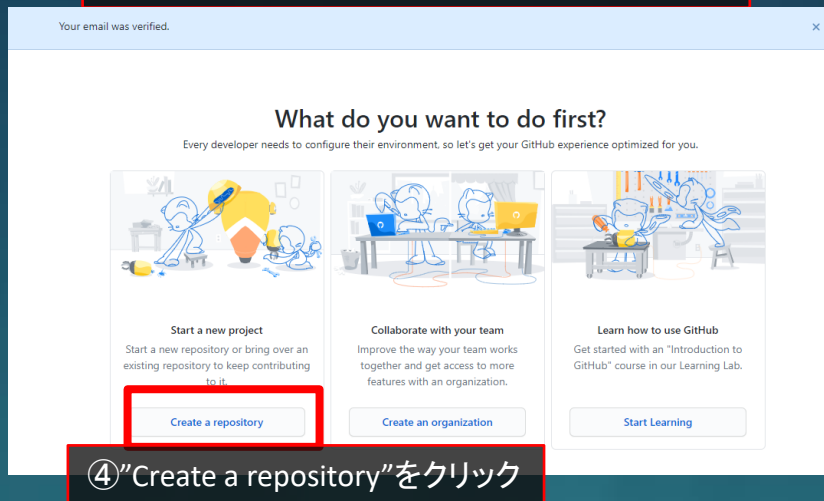
# 1) GitHubでアカウント取得 → ログイン

d) メールで確定 → アカウント作成終了 → Create a repository

① Gmail(学校のアドレス)に以下のようなメールが来ている



③このような画面が出ればアカウント作成終了



## 2) [学籍番号文字列].github.io という名前でリポジトリを作る

### Create a new repository

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

Owner

e194199kt

Repository name \*

e194199kt.github.io

Great repository names are short and memorable. Need inspiration? How about bookish-winner?

De

この部分が  
一致してるのを確認

☒ Public

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

☐ Private

You choose who can see and commit to this repository.

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

☐ Initialize this repository with a README

This will let you immediately clone the repository to your computer.

Add .gitignore: None

Add a license: None



Create repository

この手順も間違えると  
面倒なので気をつけて

① [学籍番号の文字列].github.io  
と入力

(本来リポジトリ名は自由だが、  
上記の名前にするとWebサイト構築用のリポジトリになる)

② Public(公開)が選ばれていることを確認

③ クリック

### 3) 作ったリポジトリに課題のファイルをアップロード(コミット)

#### a) リポジトリへのファイルアップロード画面へ

Learn Git and GitHub without any code!

Using the Hello World guide, you'll start a branch, write comments, and open a pull request.

Read the guide

①先ほど作ったリポジトリ名になっている

e194199kt e194199kt.github.io

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Security Insights Settings

②Codeをクリック

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

Set up in Desktop or SSH or HTTPS `https://github.com/e194199kt/e194199kt.github.io.git`

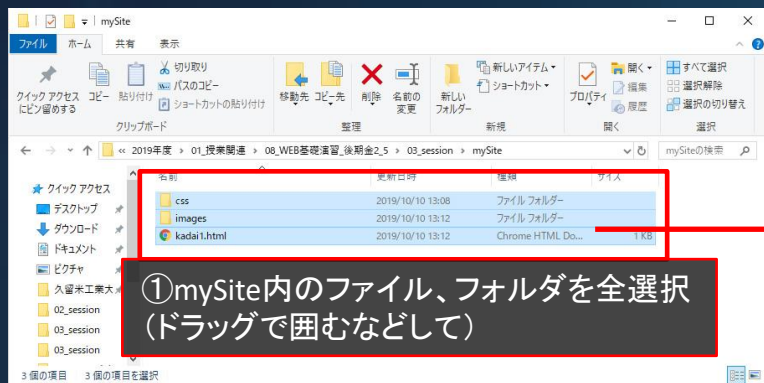
Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a README, LICENSE, and .gitignore.

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

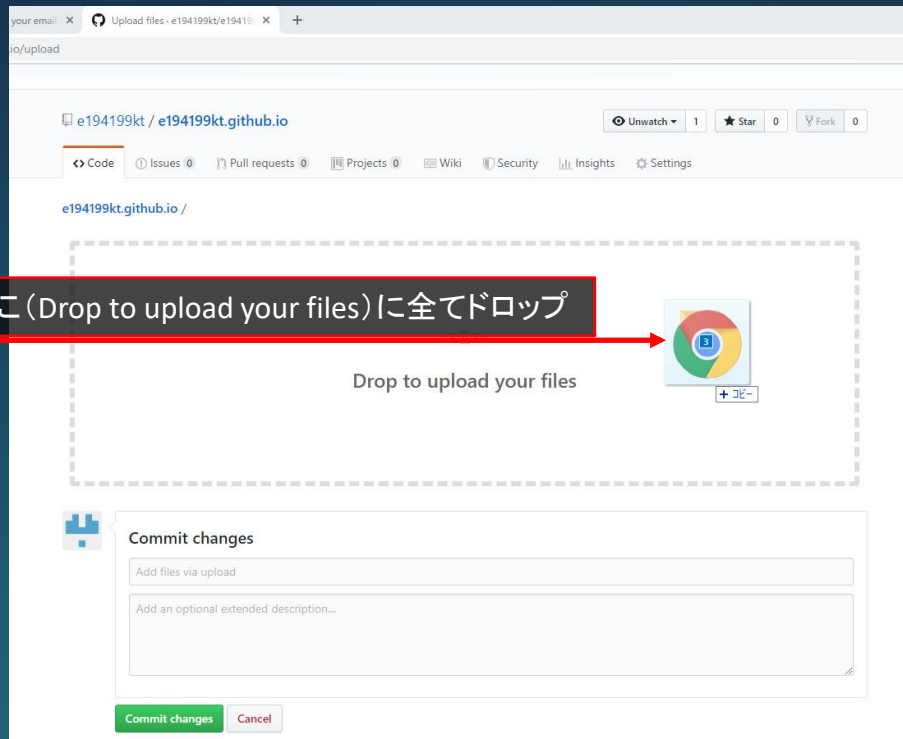
```
echo "# e194199kt.github.io" >> README.md
git init
```

### 3) 作ったリポジトリに課題のファイルをアップロード(コミット)

b) mySite内にあるファイルをドロップ



②ここ(Drop to upload your files)に全てドロップ



### 3) 作ったリポジトリに課題のファイルをアップロード(コミット)

c) コメントを書いて Commit changesでアップロード完了

①アップロード予定のファイルが表示

②コメント(任意)

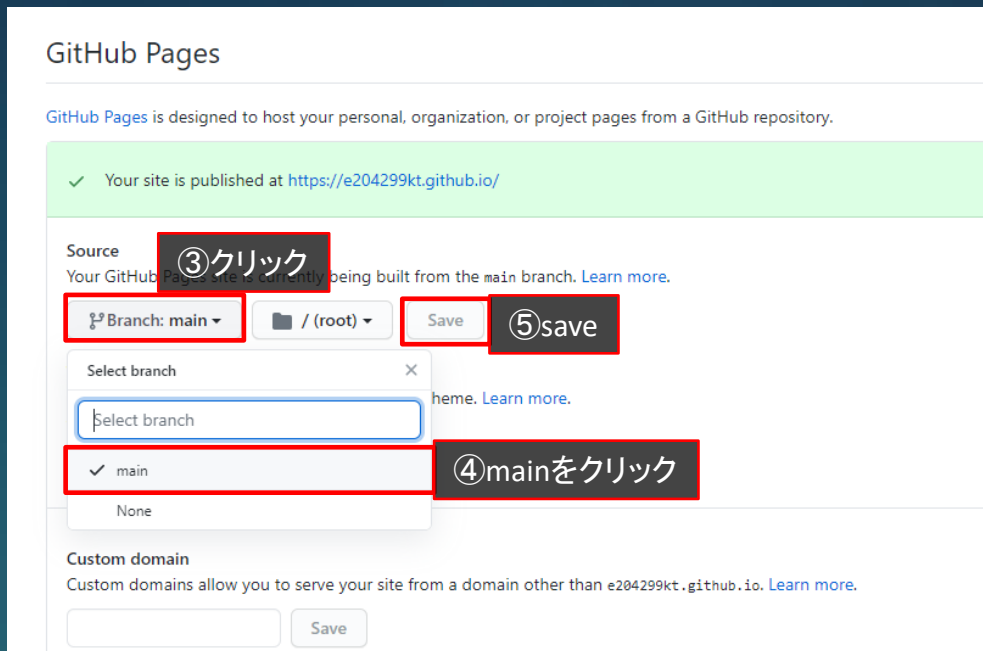
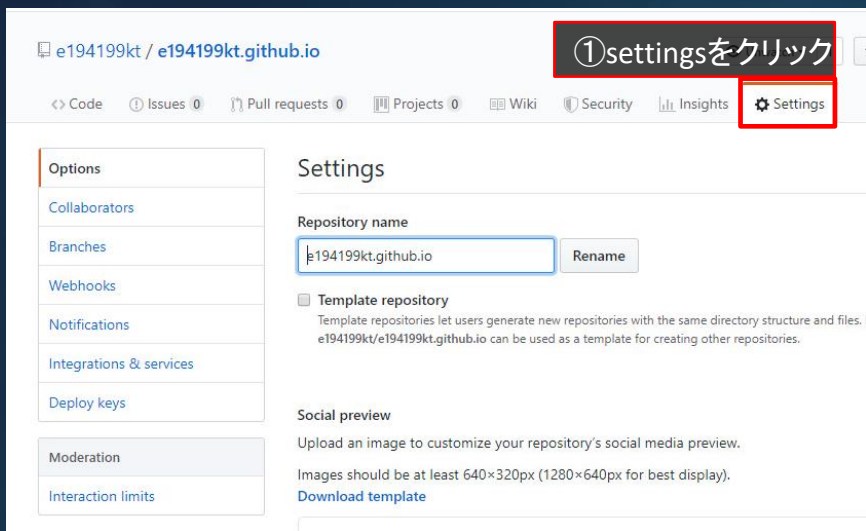
③クリック

④アップロード完

## 4) GitHub PagesのSourceとテーマを設定

a) Settingボタンを押し、Sourceを”main”に変更

②下へスクロールしていくとGitHub Pagesの項目がある





# 4) GitHub PagesのSourceとテーマを設定

## b) テーマ(Theme)も選択

☒ Automatically delete head branches  
Deleted branches will still be able to be restored.

### GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

✓ Your site is published at <https://e194199kt.github.io/>

**Source**  
Your GitHub Pages site is currently being built from the `master` branch. [Learn more.](#)  
User pages must be built from the `master` branch.

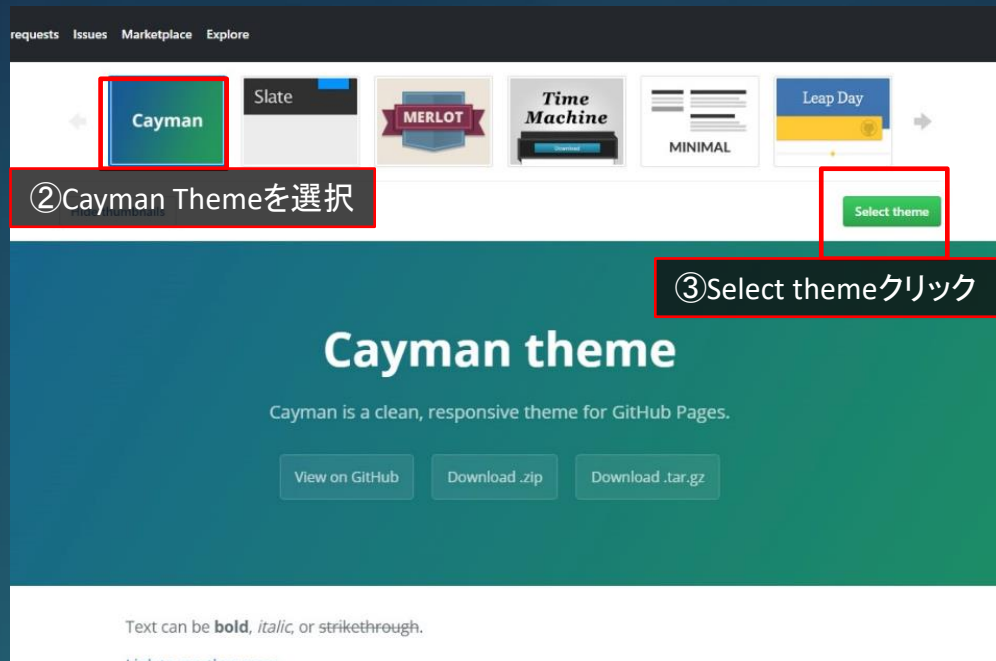
**Theme Chooser**  
Select a theme to publish your site with a Jekyll theme. [Learn more.](#)

**①Choose a Themeをクリック**

☒ **Enforce HTTPS**  
— Required for your site because you are using the default domain (e194199kt.github.io).  
HTTPS provides a layer of encryption that prevents others from snooping on or tampering with traffic to your site. When HTTPS is enforced, your site will only be served over HTTPS. [Learn more.](#)

### Danger Zone

Make this repository private  
Hide this repository from the public.



## 5) 表示の確認 ( [https://\[学籍番号文字列\].github.io/kadai1.html](https://[学籍番号文字列].github.io/kadai1.html) )

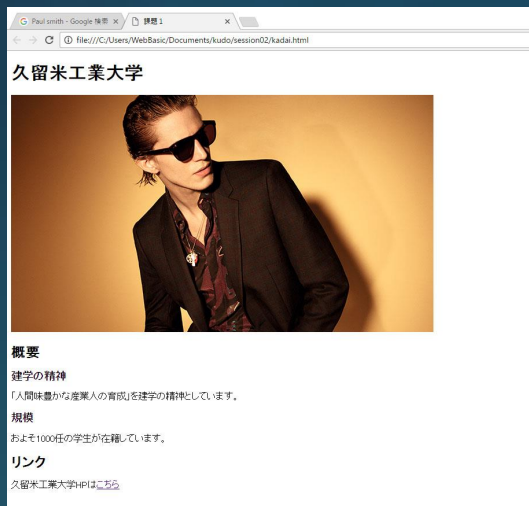
- ・ アップロードしたファイルは、全世界に公開される
- ・ ブラウザを使ってどこからでも（皆の携帯からでも）見ることができる

課題 1 のURL

<https://e〇〇〇.github.io/kadai1.html>



自分のGitHubアカウント名  
(学籍番号+2文字)

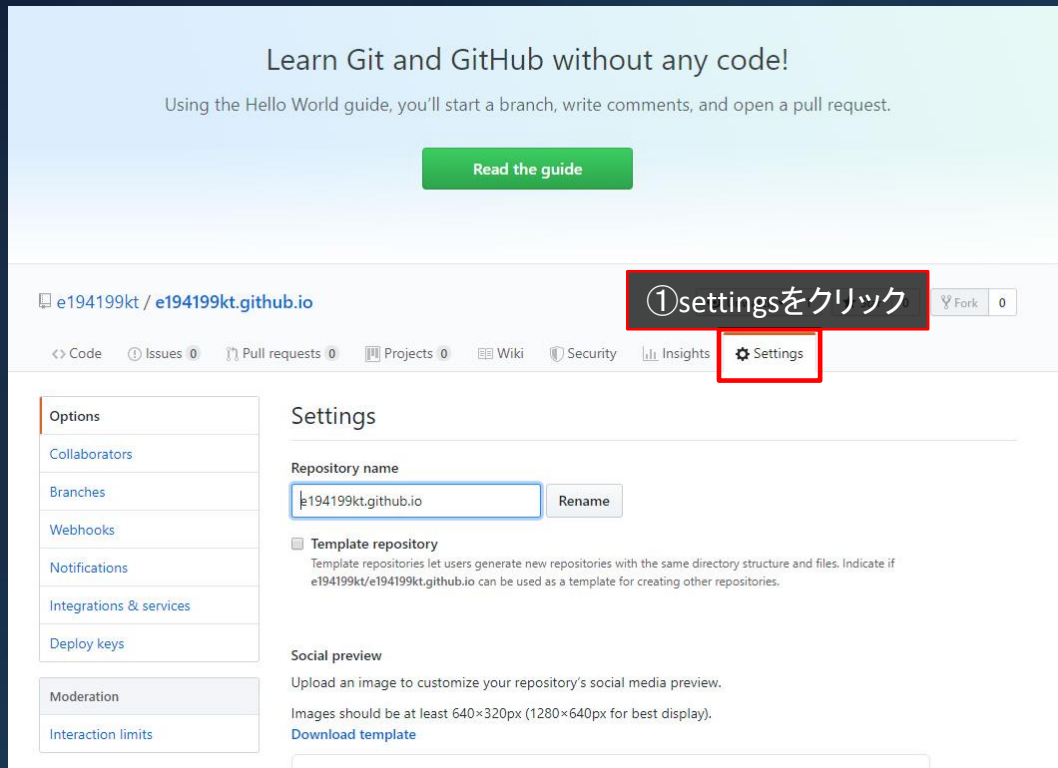


- 1) ちゃんとページが出るか
- 2) リンクは正しく動いているか

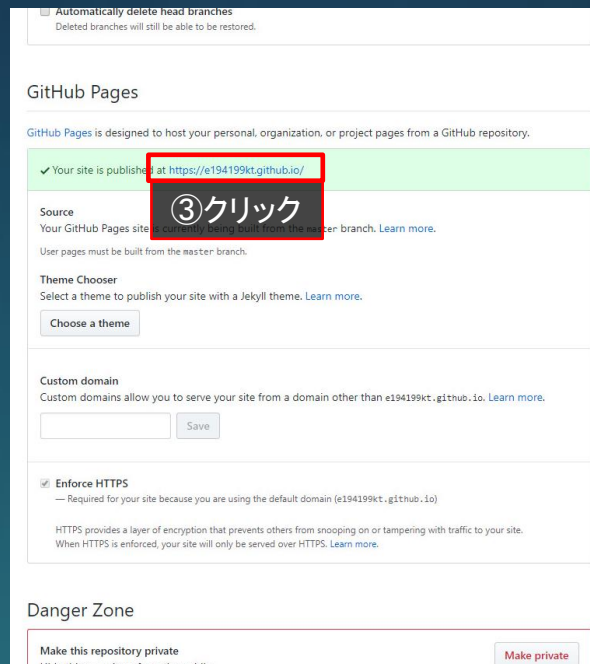
確認

## 5) 表示の確認 ( [https://\[学籍番号文字列\].github.io/kadai1.html](https://[学籍番号文字列].github.io/kadai1.html) )

a) [https://\[学籍番号文字列\].github.io](https://[学籍番号文字列].github.io) までは以下の手順でも行ける



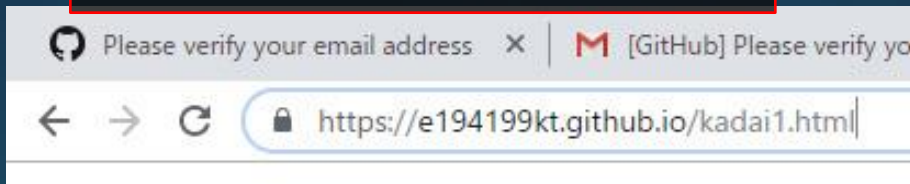
② 下へスクロールしていくとGitHub Pagesの項目がある



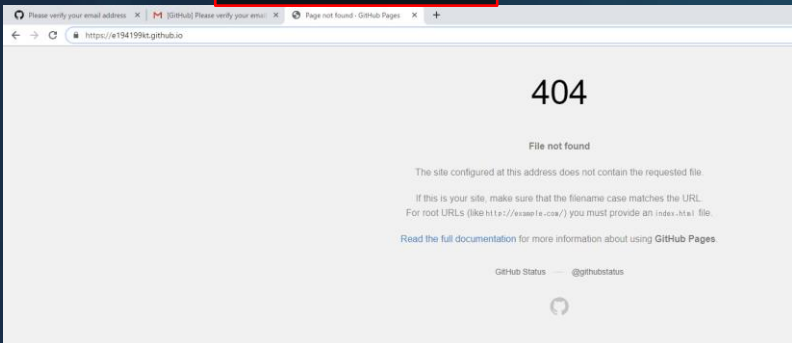
## 5) 表示の確認 ([https://\[学籍番号文字列\].github.io/kadai1.html](https://[学籍番号文字列].github.io/kadai1.html))

b) URLの最後に課題のhtmlファイル名 : kadai1.html を加えればよい

②URLに **/kadai1.html** を追加 → エンターキー



①404画面が表示



③ページが表示される



# 次回の講義内容

- 小テスト2(今回の内容から)
- body内に書くコンテンツタグの続き
- 課題2