

1. 環境構築編

ポートフォリオサイトを作ろうハンズオン
2024-12-16 / @newt239

このセクションのゴール

- 環境構築を完了する
- GitHub Pages で Web サイトを公開する
- どんなサイトを作るか考える

目次

1. 最低限必要な知識
2. 準備
3. GitHub Pages での公開
4. ウェブサイトのデザインを考える

01-1. 最低限必要な知識

心構え

書き方を覚えようとしないでください。

量が膨大というのもありますが、特に Web 技術は日々進化しているため、数年前の常識が通用しないことがあります。個人的には写経（サンプルコードを 1 文字ずつ書き写すこと）にも意味がないと思っているので、どんどんコピペしてください。

そもそもこれはテストではないので、「勉強」という意識を持たないほうがラクです。

一方でもしエンジニアを目指しているのであれば、サンプルコードを読んで「どのコードがどこに影響を与えているか」を理解しながら進めることをおすすめします。

心構え

書き方を覚えるのではなく、調べ方を覚えてください。

詰まったときは信頼できるドキュメントを見るのが一番ですが、英語であることが多いので、ハードルを感じる場合は以下に挙げるサイトがおすすめです。

- [MDN Web Docs](#)
- [HTML&CSS 入門：イチから Web デザインを習得する講座](#)

ChatGPT や GitHub Copilot などの生成 AI を活用するのも良いでしょう。

各言語の役割

- HTML: 文書の構造を記述
 - 「ここが見出し」「ここが段落」など
- CSS: デザインを記述
 - 「見出しが赤色」「段落のフォントサイズは 16px」など
- JavaScript: 動的な挙動を記述
 - 「ボタンをクリックしたらローディングアニメーションを表示」など

ウェブページを開いたときに表示されているのはほとんどの場合 HTML です。

基本的に CSS と JavaScript は HTML から呼び出されます。

今回のハンズオンでは HTML と CSS のみを扱います。

01-2. 準備

事前準備

必要なアカウント

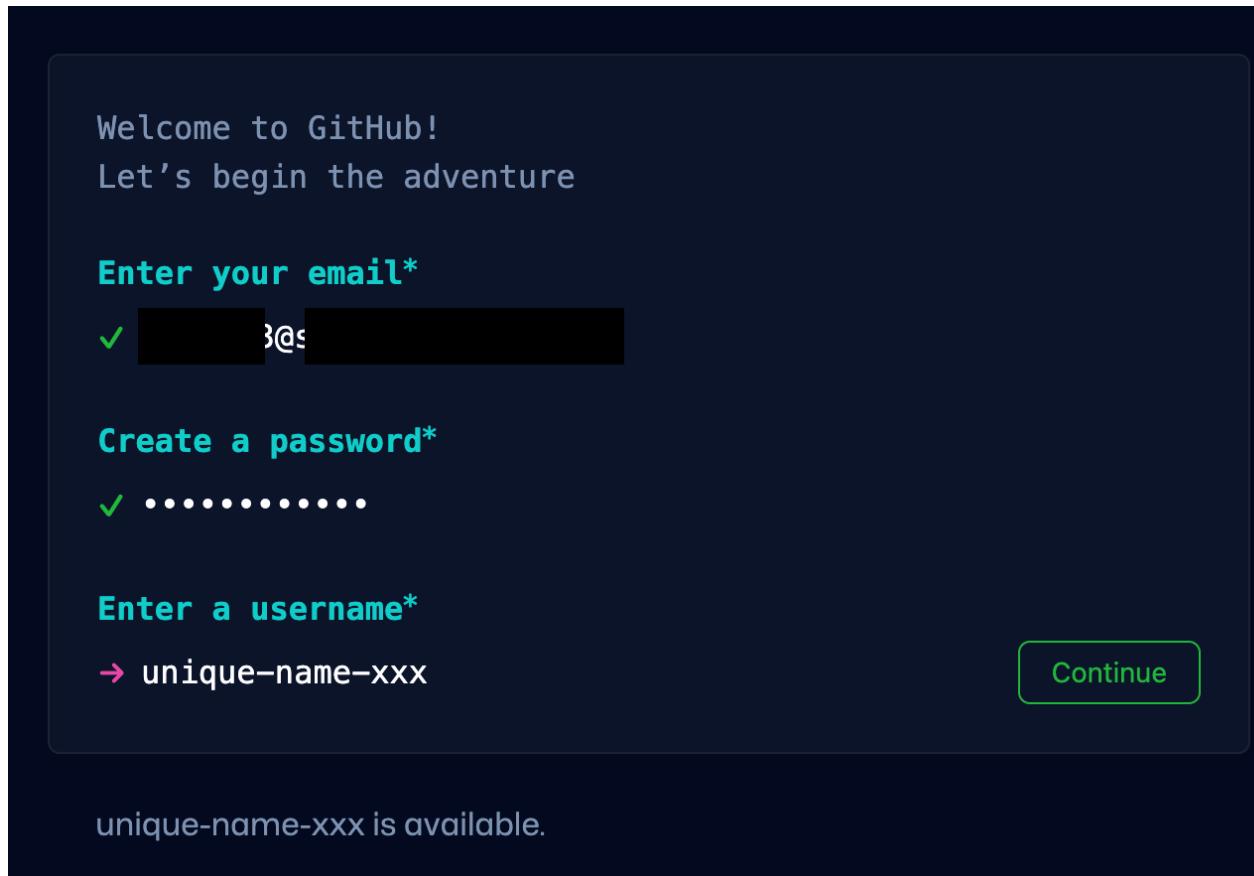
- GitHub
 - リモートリポジトリのホスティングサービス

必要なソフトウェア

- Git
 - バージョン管理ツール
- Visual Studio Code
 - 多機能なコードエディタ

GitHub アカウントの作成

1. GitHub にアクセス
2. 右上の `Sign up` をクリック
3. メールアドレス、パスワード、ユーザ名を設定
4. bot テストのミニゲームをクリア
5. メールで送られた認証コードを入力
6. プランは Free を選択
 - メールアドレスは個人のものを推奨
 - ユーザ名は他人と被っている場合利用できません



Git のダウンロード

Windows の場合

1. [Git for Windows](#) にアクセス
2. [Click here to download](#) をクリック

Mac の場合

1. [XCode](#)にアクセス
2. 右上の Download をクリック
3. App Store が開くので、 XCode をインストール

The screenshot shows the official Git website at git-scm.com. The top navigation bar includes links for About, Documentation, Downloads, and Community. The Downloads section is highlighted with a red box around the 'Click here to download' button, which links to the latest 64-bit version (2.46.2). Below this, there are links for other Windows downloads like Standalone Installer, 32-bit and 64-bit Setup, and Portable versions. A sidebar on the left promotes the 'Pro Git book' by Scott Chacon and Ben Straub, available online or on Amazon. The 'Now What?' section provides instructions for starting to use Git.

git --distributed-is-the-new-centralized

Type / to search entire site...

[About](#)

[Documentation](#)

[Downloads](#)

GUI Clients
Logos

[Community](#)

[Click here to download](#) the latest (2.46.2) 64-bit version of **Git for Windows**. This is the most recent [maintained build](#). It was released **2 days ago**, on 2024-09-24.

Other Git for Windows downloads

Standalone Installer
[32-bit Git for Windows Setup](#).
[64-bit Git for Windows Setup](#).

Portable ("thumbdrive edition")
[32-bit Git for Windows Portable](#).
[64-bit Git for Windows Portable](#).

Using winget tool
Install [winget tool](#) if you don't already have it, then type this command in command prompt or Powershell.
`winget install --id Git.Git -e --source winget`

The current source code release is version **2.46.2**. If you want the newer version, you can build it from [the source code](#).

Now What?

Now that you have downloaded Git, it's time to start using it.

[Read the Book](#)
Dive into the Pro Git book and learn at your own pace.

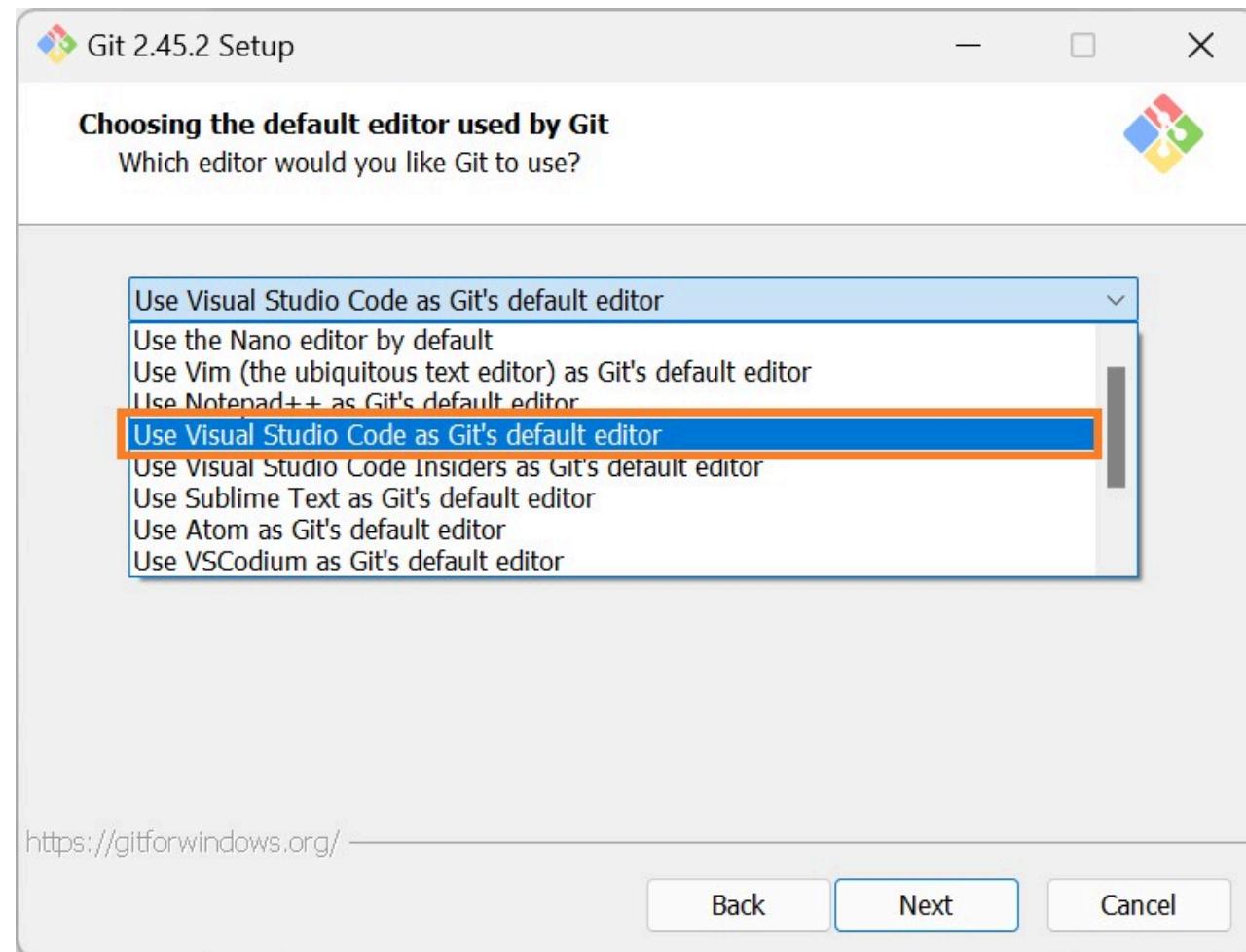
[Download a GUI](#)
Several free and commercial GUI tools are available for the Windows platform.

[Get Involved](#)
A knowledgeable Git community is available to answer your questions.

Git のインストール

Windows の場合

1. ダウンロードフォルダにある exe ファイルをダブルクリック
2. ウィザードが立ち上るので、基本的に右下の「Next」をクリックし続ける
3. 「Choosing the default editor used by Git」画面で、Use Visual Studio Code as Git's default editor を指定する



Git の設定

- スタートメニュー(左下 Windows アイコン)から Git Bash を起動
 - Mac の場合は「アプリケーション」の「ユーティリティ」フォルダにあるターミナルを起動
- 下記のコマンドを入力して、ユーザ名とメールアドレスを設定

```
git config --global user.name "ここにGitHubのユーザ名"
```

```
git config --global user.email "ここにGitHubのメールアドレス"
```

- 特にエラーが出ていなければ設定完了

Visual Studio Code のダウンロード

1. [Visual Studio Code](#) にアクセス
2. OS に合わせてインストーラをダウンロード
3. インストール

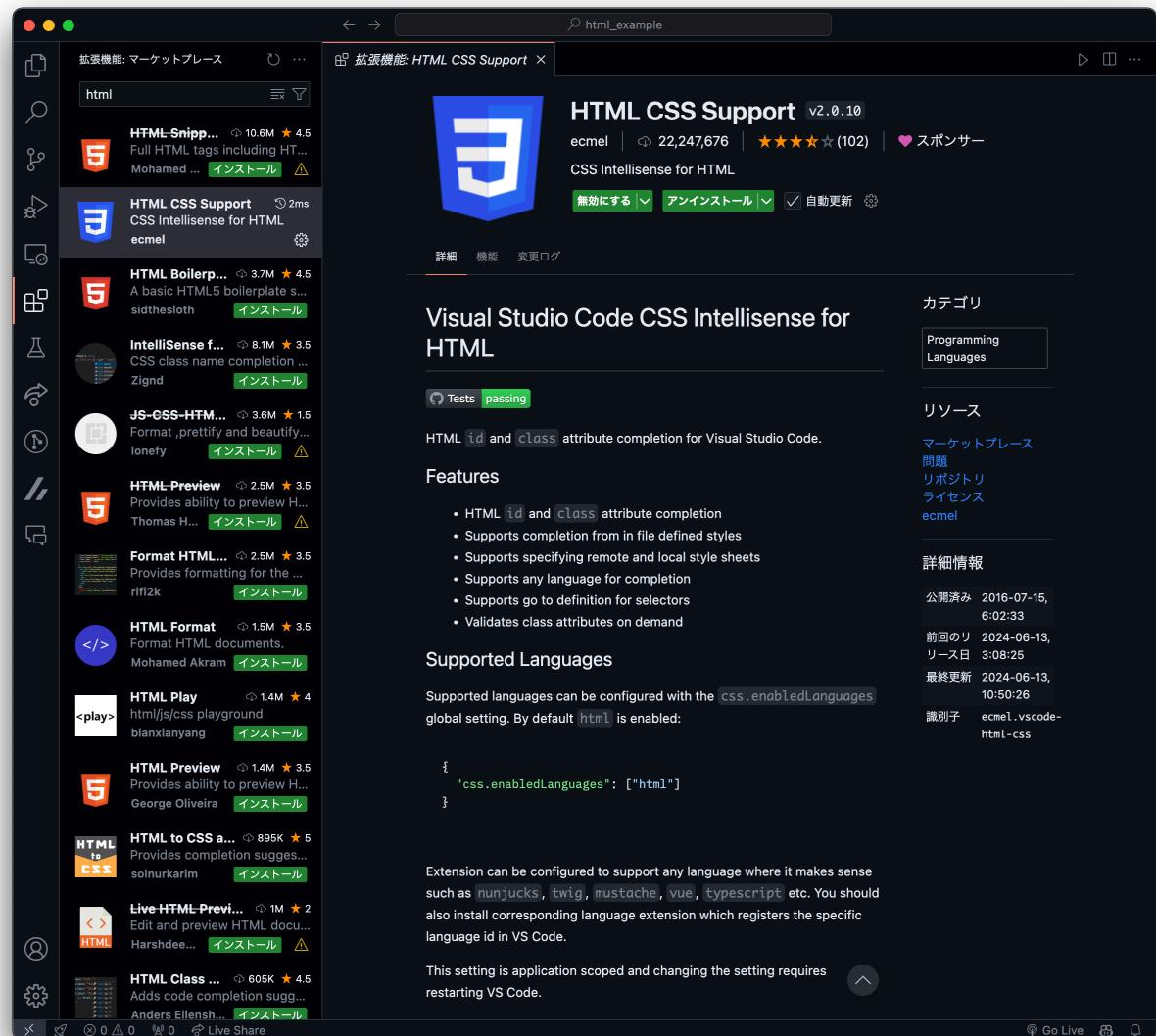
補足: Visual Studio Code とは

- Microsoft が主導して開発しているオープンソースのテキストエディタ
- 一般的に **VSCODE** と呼ばれるので、本資料でも以降は VSCODE と表記します
- 「Visual Studio」 と 「Visual Studio Code」 は別物です

VSCode の設定

拡張機能のインストール

- VSCode を起動し、左側のアイコンが縦に並んだエリアから「拡張機能」タブを探す
- 「HTML CSS Support」と「Live Server」を検索してインストール



01-3. GitHub Pages での公開

GitHub Pages とは

- GitHub が提供する静的 Web ホスティングサービス
- 以前はレンタルサーバーを契約し、FTP クライアントなどを使って公開したいファイルをアップロードする必要があったが、近年は Git でソースコードを管理し、GitHub に push するだけで Web サイトを開けるサービスが増えている
- 同様の他のサービスとしては Netlify や Vercel, Firebase Hosting, Cloudflare Pages などがある（いずれも無料で公開可能）

Git と GitHub の違い

- Gitはプログラムのバージョンを管理するためのソフトウェア
- GitHubは Git で管理されたプログラムをクラウド上で管理するためのWeb サービス
 - 同様の Web サービスとして GitLab, Bitbucket などもある
- GitHub は Git の機能に加え、Issue(プロジェクトの問題やタスク) を管理する機能や共同開発を支援する機能などを提供している
- GitHub Pages は GitHub が提供するサービスの一つ

GitHub でリポジトリを作成する

- GitHub にログインし、右上の「+」をクリックして「New repository」を選択
- Repository name にはお好きなものを
 - [https://username.github.io/\[ここがリポジトリ名になる\]/](https://username.github.io/[ここがリポジトリ名になる]/)
- Visibility は Public のまま
 - 認証情報（シークレットキーや個人情報）を載せないよう注意！
- Create repository をクリック
- 作成後に遷移したページの URL をコピー
 - 4 枚後のスライドで使います

Create a new repository

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

Required fields are marked with an asterisk (*).

Repository template

No template ▾

Start your repository with a template repository's contents.

Owner *



newt239 ▾

Repository name *

portfolio

portfolio is available.

Great repository names are short and memorable. Need inspiration? How about [scaling-octo-guide](#) ?

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.

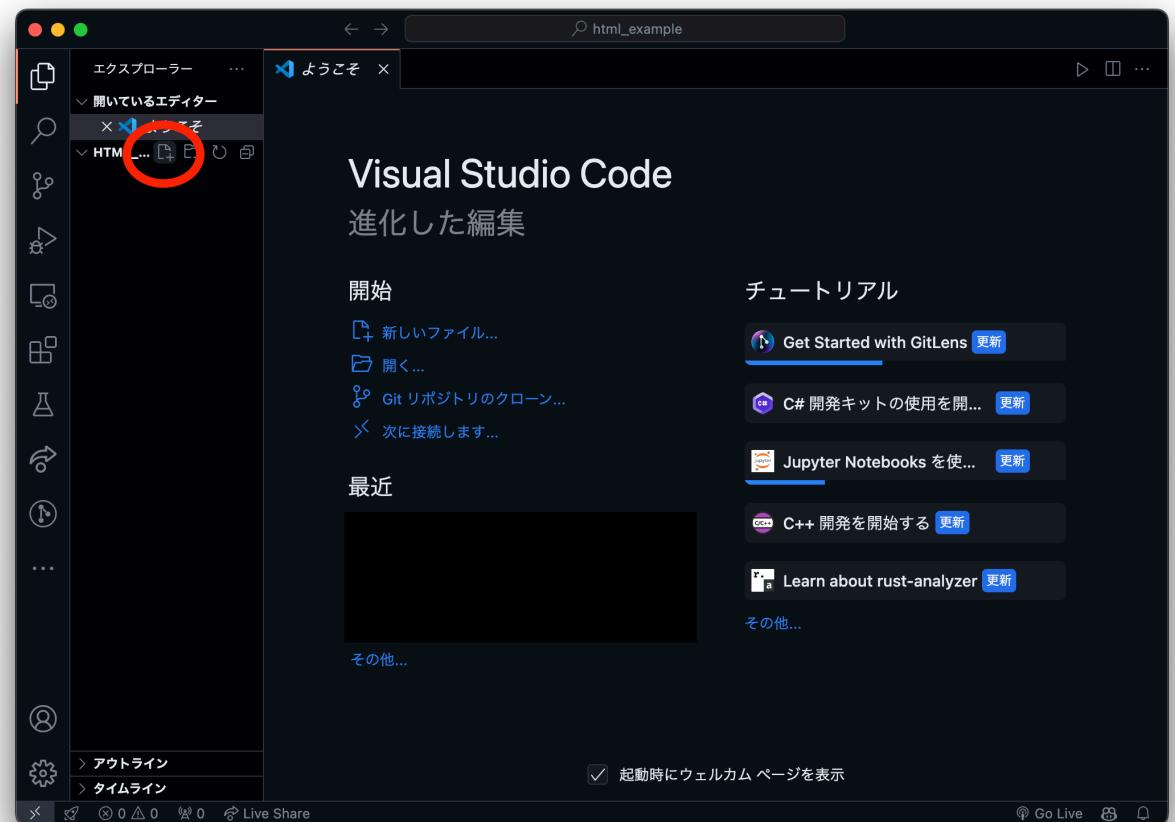
Initialize this repository with:

Add a README file

This is where you can write a long description for your project. [Learn more about READMEs.](#)

HTML ファイルを作成してみよう

1. パソコン上で作業用のフォルダ（portfolioなど）を作成
2. VSCode を起動し、左上の「ファイル」→「フォルダを開く」から作成したフォルダを開く
3. 新しいファイルを作成（右図の赤丸）
4. ファイル名を `index.html` として Enter
 - フォルダ作成ボタンと間違えないように注意
 - もしくは左上の「ファイル」→「新しいファイル」



HTML をブラウザで表示してみよう

- 作成した HTML ファイルに先ほどのコードを書き込む（コピー推奨）

```
<!DOCTYPE html>
<html lang="ja">
  <head>
    <meta charset="UTF-8" />
    <title>ページのタイトル</title>
  </head>
  <body>
    <h1>見出し</h1>
    <p>段落</p>
  </body>
</html>
```

- ファイルを保存し、VSCode 右下の「Go Live」をクリック
 - エクスプローラーから HTML ファイルをダブルクリックすることでも表示できますが、この方法を用いることでリアルタイムで（再読みすことなく）変更を確認できます

HTML をブラウザで表示してみよう



Git でコミット＆プッシュする

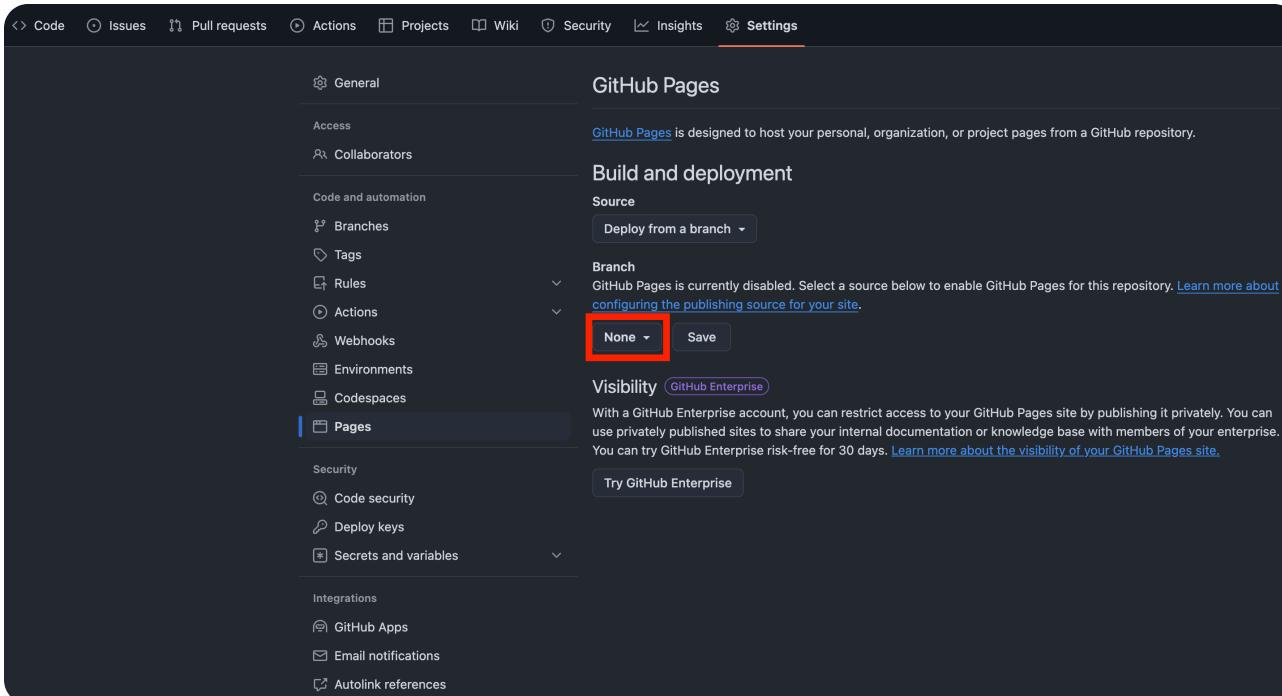
- VSCode 左上のメニューから「ターミナル」→「新しいターミナル」を選択
- 以下のコマンドを入力

```
git init
git remote add origin さっきコピーしたURL
git add .
git commit -m "first commit"
git push origin main
```

- コマンドは 1 行ずつコピーして貼り付け、実行 (Enter)
- 最後のコマンドに失敗する場合、`main` を `master` に変更
- いまのところコマンドの意味を理解する必要はありません！

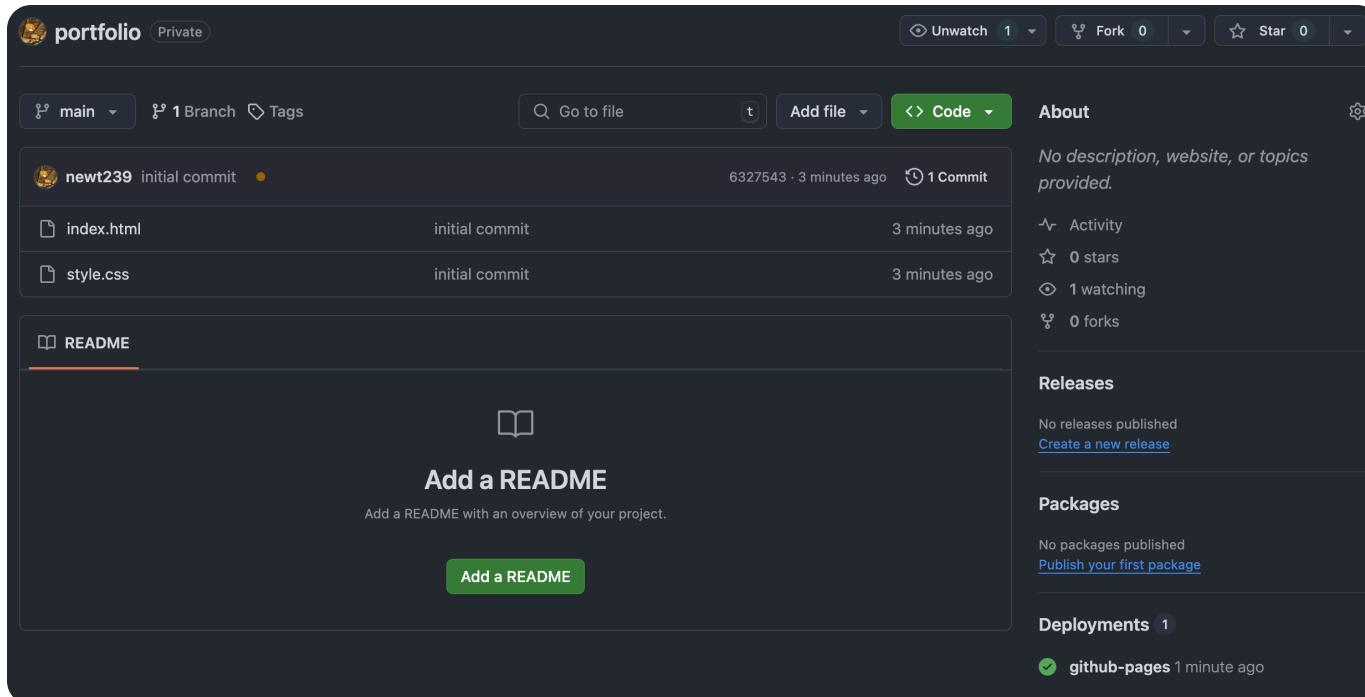
GitHub Pages で公開する

- ・コマンド実行後、GitHub で開いているページをリロード
- ・「Settings」 → 「Pages」 を選択
- ・Branch を main (または master) にして Save



公開できたことを確認

- Code タブに戻り、黄色い丸が緑のチェックマークになるまで待つ
 - 30 秒程度でリロードしてみてください
- 右下の Deployments の github-pages をクリックし、URL をクリック！



ウェブサイトのデザインを考える

世の中の Web サイトを見てみよう

- SANKOU! | Web デザインギャラリー・参考サイト集
 - <https://sankoudesign.com/>
- デジタル庁
 - <https://www.digital.go.jp/>
- pixiv
 - <https://www.pixiv.net/>
- Gigazine
 - <https://gigazine.net/>
- にじさんじフェス 2025
 - <https://fes.nijisanji.jp/2025/>

デザインを考える

- 大まかなページ構成を考える
 - トップページ、プロフィール、作品集、お問い合わせなど
- レイアウトを考える
 - サイドバー、ヘッダー、フッターなど
- 配色を考える

次ページでは Web デザインの制作でよく利用されている Figma というサービスを紹介しますが、今回は好きなツール（紙 or ペイントツール）でデザインを考えてもらって構いません。

Figma を使ってデザインを考える

- Figma にアクセスし、アカウントを作成
- 学割のプログラムがあるので申請すると良い
 - <https://www.figma.com/ja-jp/education/>

第1回の内容は以上です。お疲れ様でした！