Github Made Easy : Basics of Github

Table of Contents

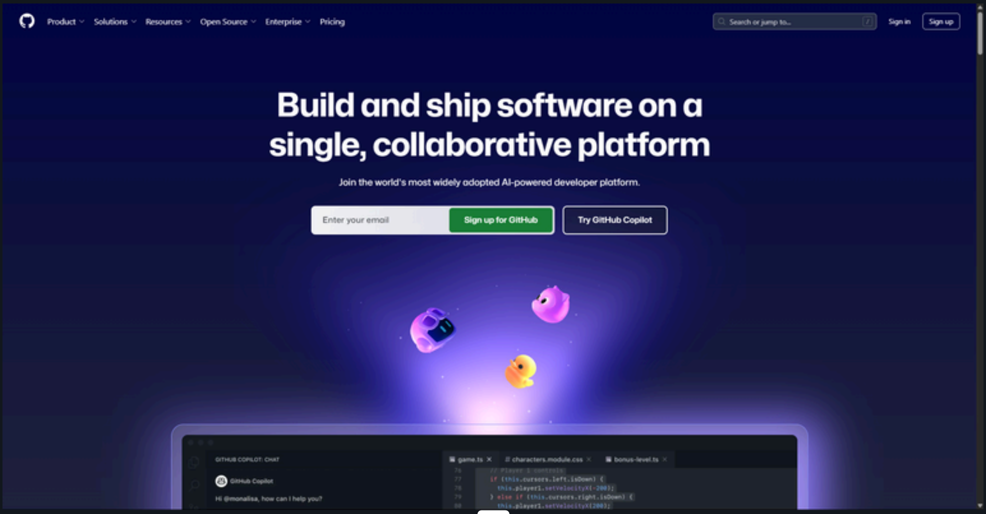
Topic	Page
Account Setup	01
Create a Repository (Project Folder)	02
Add Files to Your Repository	03
Understand: README.md, .gitignore, and LICENSE	05
Free Website Hosting (Enable GitHub Pages)	07
Preparing & Uploading HTML/CSS/JS to GitHub Properly	09
Meta-Prompting	11

Click the page number to navigate directly to that page.

This PDF/guide was created by Runarok.
To find the latest version of the PDF guide, visit : Runarok Github Manual

Account Setup

Navigate to GitHub : Go to Github



Build and ship software on a single, collaborative platform

Join the world's most widely adopted AI-powered developer platform.

Enter your email | Sign up for GitHub | Try GitHub Copilot

If You're New:

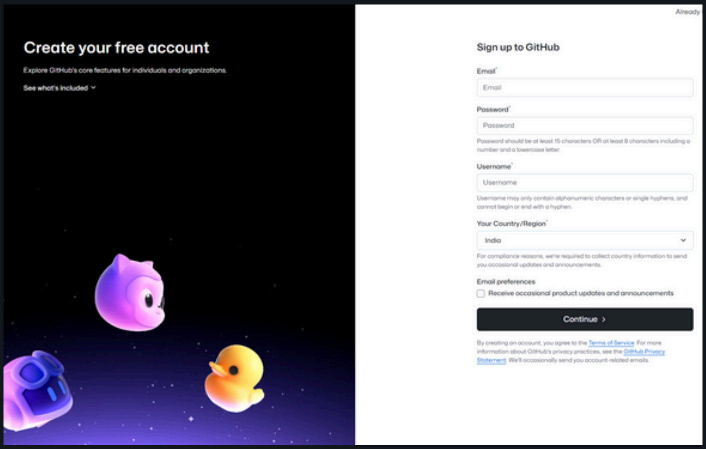
Click Sign Up

Fill in: Email , Password , Username

Choose Free plan

  -(Pro not needed for public site hosting)

Complete CAPTCHA & setup



Create your free account

Explore GitHub's core features for individuals and organizations.

See what's included

Already

Sign up to GitHub

Email*
Email

Password*
Password

Password should be at least 15 characters OR at least 8 characters including a number and a lowercase letter.

Username*
Username

Username may only contain alphanumeric characters or single hyphens, and cannot begin or end with a hyphen.

Your Country/Region*
India

For compliance reasons, we're required to collect country information to send you occasional updates and announcements.

Email preferences
☐ Receive occasional product updates and announcements

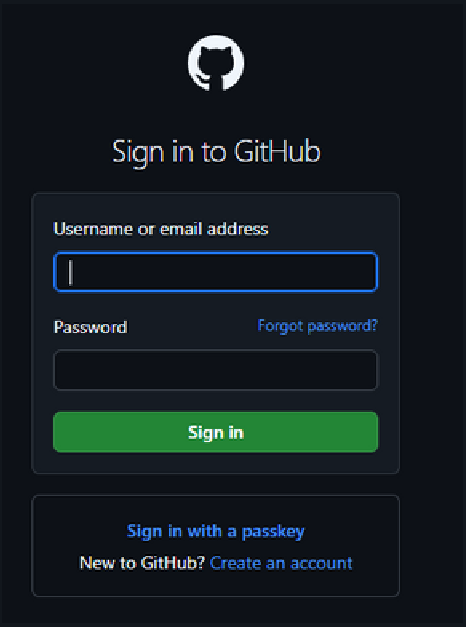
Continue ›

By creating an account, you agree to the Terms of Service. For more information about GitHub's privacy practices, see the GitHub Privacy Statement. We'll occasionally send you account-related emails.

If You Already Have an Account:

Click Sign In (top-right)

Enter your username/email and password



Sign in to GitHub

Username or email address

Password | Forgot password?

Sign in

Sign in with a passkey

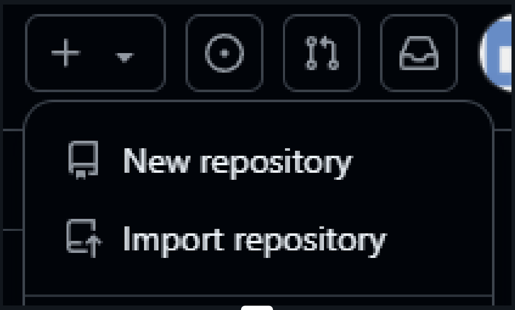
New to GitHub? Create an account

→ Back to Table of Contents

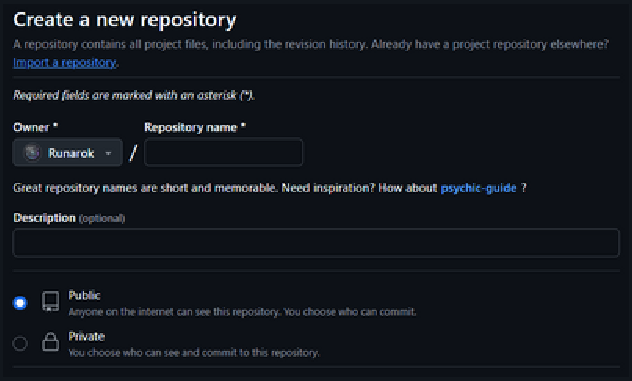
Create a Repository (Project Folder)

GitHub gives you two options when creating a new project:

Click the "+" icon (top-right of any GitHub page)



New Repository (Start from Scratch)



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 (*).

Owner * Runarok / Repository name *

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

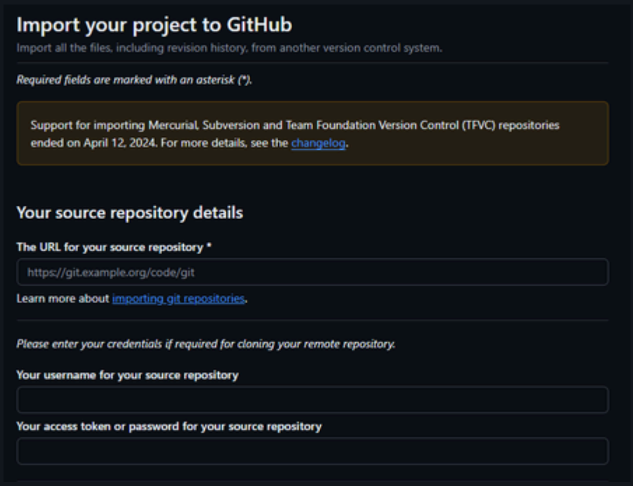
Use If:

- Starting a new project from scratch
- Uploading files manually (no importing)

Steps:

- Enter repository name
  - → e.g., my-website, project-1
- (Optional) Add a short description
- Choose visibility → Select Public (required for Free Pages hosting)
- Optional initializations
  - README.md , License , .gitignore
- Click Create repository

Import Repository (From another platform)



Import your project to GitHub

Import all the files, including revision history, from another version control system.

Required fields are marked with an asterisk (*).

Support for importing Mercurial, Subversion and Team Foundation Version Control (TFVC) repositories ended on April 12, 2024. For more details, see the changelog.

Your source repository details

The URL for your source repository *

https://git.example.org/code/git

Learn more about importing git repositories.

Please enter your credentials if required for cloning your remote repository.

Your username for your source repository

Your access token or password for your source repository

Use this if:

- You want to copy code from a GitLab, Bitbucket, or other repository to GitHub

Steps:

- Paste the URL of the existing repository
- Choose a new name for your GitHub repo
- Choose Public or Private
- Click Begin Import

GitHub will copy the files and commit history from that repo.

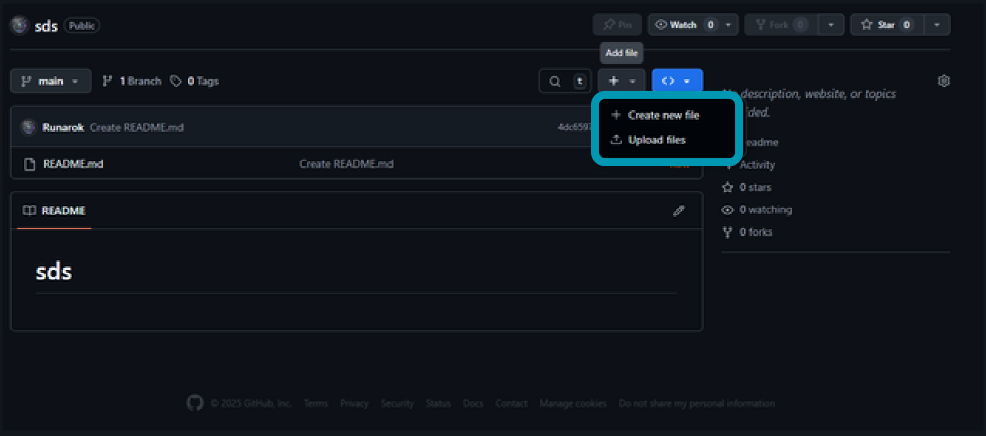
→ Back to Table of Contents

Add Files to Your Repository

Firstly , Was the Repo Initialized?

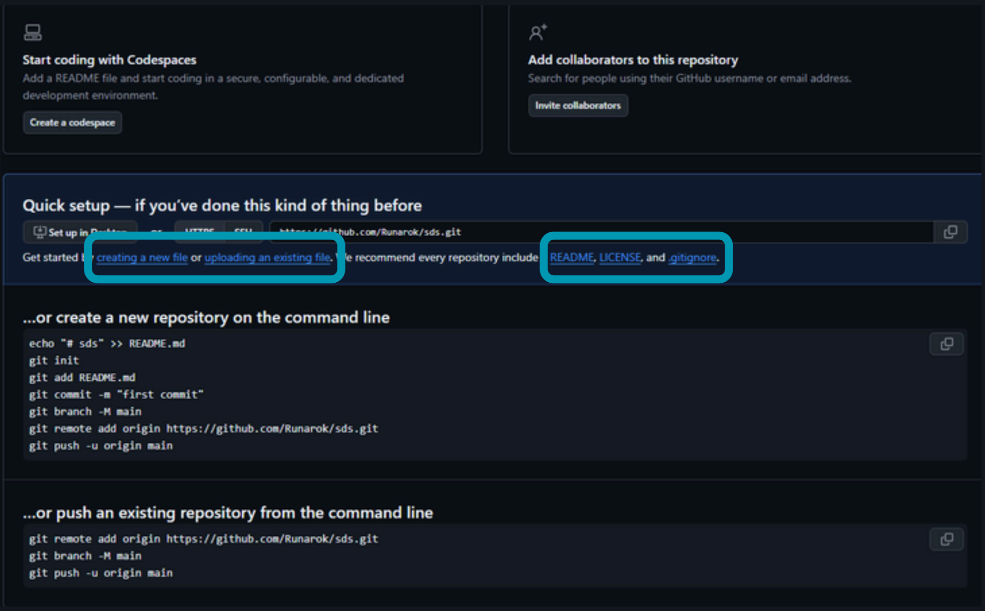
If you selected README / License / .gitignore during creation:

→ You'll see a file list right away (repo is initialized)



If you skipped all initialization:

→ GitHub shows "Quick setup" page (repo is empty)



Start coding with Codespaces

Add a README file and start coding in a secure, configurable, and dedicated development environment.

Add collaborators to this repository

Search for people using their GitHub username or email address.

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

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

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

```
git remote add origin https://github.com/Runarok/sds.git
git branch -M main
git push -u origin main
```

→ Back to Table of Contents

Upload Files (For bulk/manual upload)

Use If:

- You already have HTML/CSS/image files on your computer
- You want to drag and drop/upload multiple files

Where:

- In the repo → Click "Add file" → Upload files

Steps:

1. Drag & drop files (e.g., index.html, about.html, style.css)
2. Add Commit message and Description(Optional)
3. Scroll down → Add commit message

Can be repeated anytime to update or replace files.

Create New File (For writing online)

Use If:

- You want to create a small file manually (e.g., index.html)
- You're just editing or adding a page/content directly in browser

Where:

- Click "Add file" → Create new file

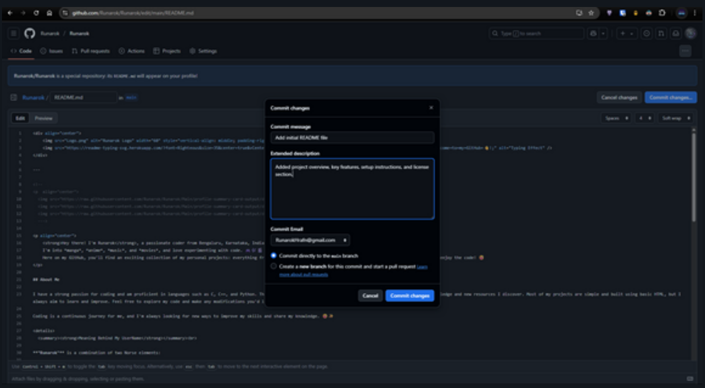
Steps:

1. Name the file (e.g., index.html)
2. Add your content (HTML, Markdown, etc.)
3. Scroll up → click Commit changes
4. Add a commit message (optional description)
5. Click Commit changes to save it

You can create as many files as you want one after another this way.

Examples:

Create New File



github.com/Runarok/Runarok/edit/main/README.md

Runarok / Runarok

Runarok/Runarok is a special repository: its README.md will appear on your profile!

Runarok / README.md in main

Commit changes

Commit message: Add initial README file

Extended description: Added project overview, key features, setup instructions, and license section.

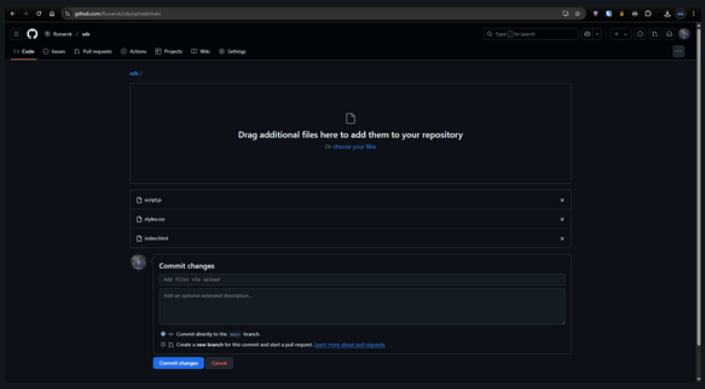
Commit Email: Runarokhrafn@gmail.com

● Commit directly to the main branch

○ Create a new branch for this commit and start a pull request Learn more about pull requests

Cancel | Commit changes

Upload Files



github.com/Runarok/ads/upload/main

Runarok / ads

ads /

Drag additional files here to add them to your repository

Or choose your files

script.js

styles.css

index.html

Commit changes

Add files via upload

Add an optional extended description...

● Commit directly to the main branch.

○ Create a new branch for this commit and start a pull request. Learn more about pull requests.

Commit changes | Cancel

→ Back to Table of Contents

Understand: README.md, .gitignore, and LICENSE

README.md — Project Overview (Markdown Format)

Purpose:

- Shown on your GitHub repo page, not on the hosted site
- Written in GFM (GitHub Flavored Markdown)

Can Include:

- #, ##, ### → Headings
- [text](url) → Links
- ``` code ``` → Code blocks
- Lists, tables, bold/italic, etc.

Visibility:

- Always shown at github.com/username/repo-name
- If repo = username, it's shown on:
    - github.com/username

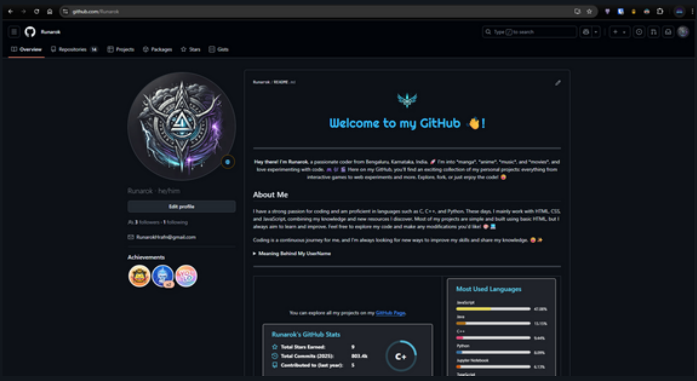
Use For:

- Explaining your project or website
- How to use/run/view
- Tool credits (e.g., "Made with ChatGPT")
- Any text you want in rich format via Markdown

Tip: New to Markdown? Check out this Markdown guide for a quick overview of formatting options.

Examples:

README shown at My Profile URL



Runarok / README.md

Welcome to my GitHub 👋!

Hey there! I'm Runarok, a passionate coder from Bengaluru, Karnataka, India. 🚀 I'm into "manga", "anime", "music", and "movies", and love experimenting with code. 🎮 🎶 📚 Here on my GitHub, you'll find an exciting collection of my personal projects: everything from interactive games to web experiments and more. Explore, fork, or just enjoy the code! 😊

About Me

I have a strong passion for coding and am proficient in languages such as C, C++, and Python. These days, I mainly work with HTML, CSS, and JavaScript, combining my knowledge and new resources I discover. Most of my projects are simple and built using basic HTML, but I always aim to learn and improve. Feel free to explore my code and make any modifications you'd like! 😊 💻

Coding is a continuous journey for me, and I'm always looking for new ways to improve my skills and share my knowledge. 😊 ✨

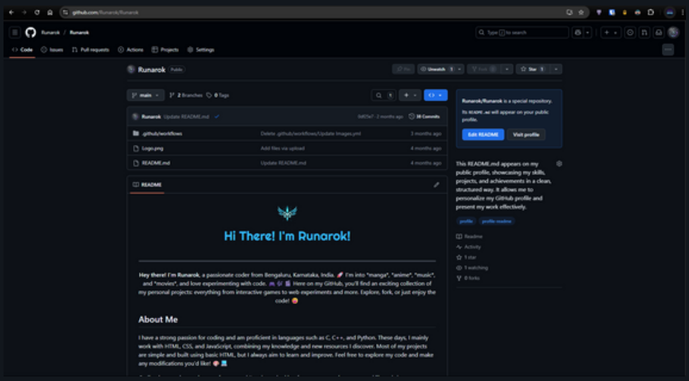
▸ Meaning Behind My UserName

You can explore all my projects on my GitHub Page.

Runarok's GitHub Stats

Total Stars Earned:	9
Total Commits (2025):	803.4k
Contributed to (last year):	5

README shown in My Profile repo



Runarok/Runarok is a special repository.

Its README.md will appear on your public profile.

This README.md appears on my public profile, showcasing my skills, projects, and achievements in a clean, structured way. It allows me to personalize my GitHub profile and present my work effectively.

.github/workflows	Delete .github/workflows/Update Images.yml	3 months ago
Logo.png	Add files via upload	4 months ago
README.md	Update README.md	4 months ago

Hi There! I'm Runarok!

Hey there! I'm Runarok, a passionate coder from Bengaluru, Karnataka, India. 🚀 I'm into "manga", "anime", "music", and "movies", and love experimenting with code. 🎮 🎶 📚 Here on my GitHub, you'll find an exciting collection of my personal projects: everything from interactive games to web experiments and more. Explore, fork, or just enjoy the code! 😊

About Me

I have a strong passion for coding and am proficient in languages such as C, C++, and Python. These days, I mainly work with HTML, CSS, and JavaScript, combining my knowledge and new resources I discover. Most of my projects are simple and built using basic HTML, but I always aim to learn and improve. Feel free to explore my code and make any modifications you'd like! 😊 💻

→ Back to Table of Contents

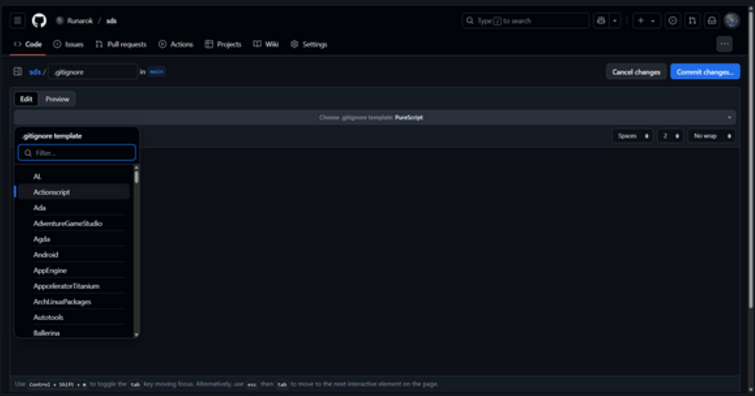
.gitignore – Ignore Unwanted Files

Purpose:

- Tells Git what not to track (e.g., system/dev files)

Only useful if:

- You're using:
  - Git CLI (terminal)
  - GitHub Desktop
- Not useful if:
  - You only upload files via GitHub website (manual uploads)



During file creation on GitHub, you can:

- Enter the file name (gitignore) and Use templates if available (for certain file types)
- Or Write your content directly in the editor
- Commit changes with a message to save the file

Tip: New to .gitignore? Check out [W3Schools]() and [freeCodeCamp]() for a clear understanding.

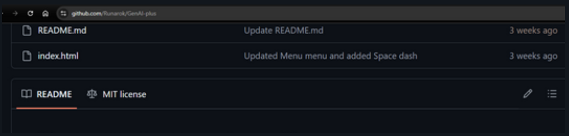
LICENSE – Open Source Permissions

Purpose:

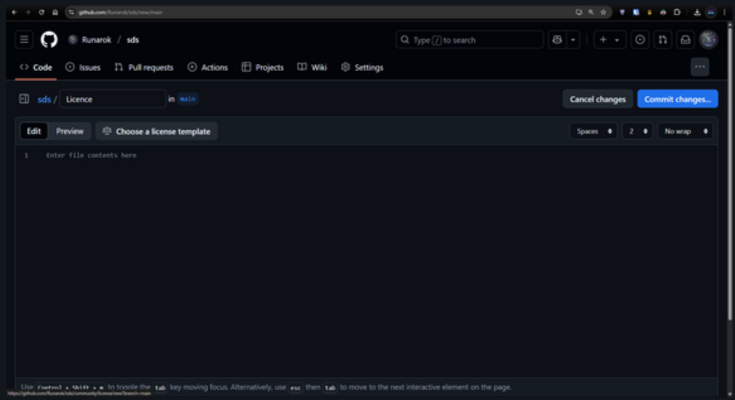
- Legally declares how others may use your code/files
- Visible publicly in the repo - Required if you want to make your project truly "open source"

Where It Appears:

- GitHub shows it as a label/tag on the repo
- Also visible as a file (LICENSE) in the repo's file list



README.md | Update README.md | 3 weeks ago
index.html | Updated Menu menu and added Space dash | 3 weeks ago



github.com/Runarok/sds/new/main
Runarok / sds
Type / to search
Code
Issues
Pull requests
Actions
Projects
Wiki
Settings
sds / Licence in main
Cancel changes
Commit changes...
Edit
Preview
Choose a license template
Spaces
2
No wrap
Enter file contents here
Use Control + Shift + m to toggle the tab key moving focus. Alternatively, use esc then tab to move to the next interactive element on the page.

During file creation on GitHub, you can:

- Enter the file name (Licence) and Use templates Or Write your content directly in the editor
- Commit changes with a message to save the file

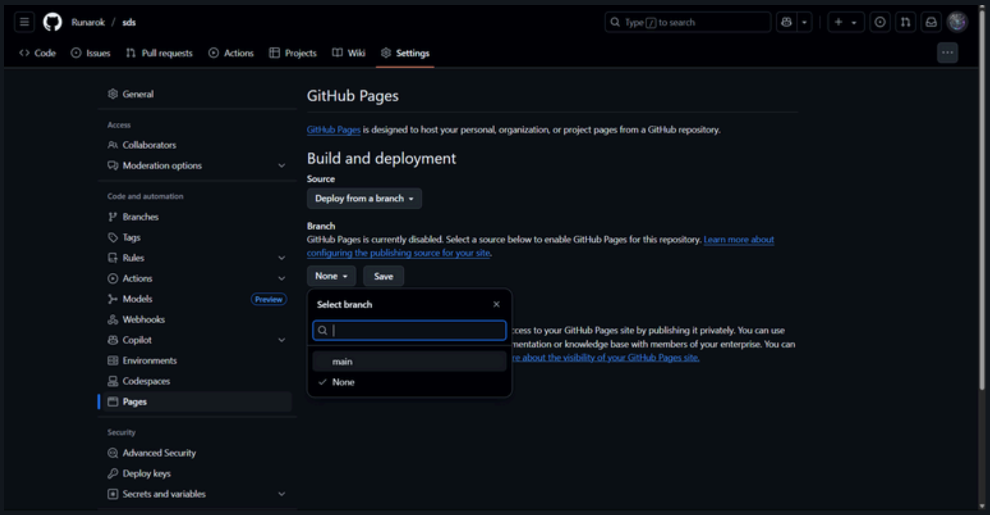
Tip: New to GitHub licensing? Visit [OSI]() for license details and [choosealicense.com]() to pick the right one.

→ Back to Table of Contents

Free Website Hosting (Enable GitHub Pages)

Steps to Host Any Repository Site:

- Go to your repository on GitHub
- Click Settings → Left sidebar → Pages
- Under Source, choose your Branch: main / master
- Folder: / (root) (or /docs if your files are inside a folder named docs) than Click Save



GitHub Pages

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

Build and deployment

Source

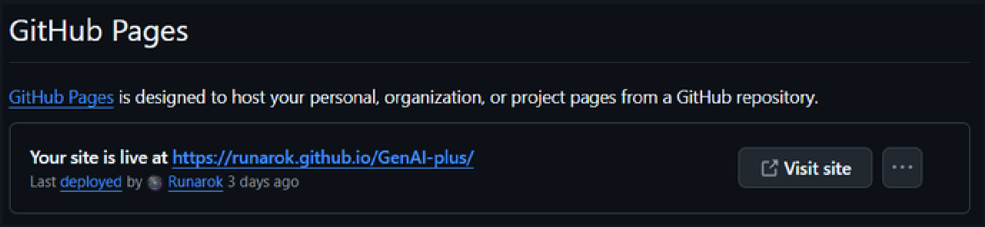
Branch

GitHub Pages is currently disabled. Select a source below to enable GitHub Pages for this repository. Learn more about configuring the publishing source for your site.

ccess to your GitHub Pages site by publishing it privately. You can use mentation or knowledge base with members of your enterprise. You can re about the visibility of your GitHub Pages site.

After You Click Save:

- GitHub automatically builds and hosts your site
- Wait for a while - A box will appear showing the live link:
  - Your site is live at: https://username.github.io/repo-name
  - Copy and share the link! It becomes active within 10–60 seconds.



GitHub Pages

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

Your site is live at https://runarok.github.io/GenAI-plus/
Last deployed by Runarok 3 days ago

Special Case: Personal Main Site (Main Portfolio Or Site)

- Repository name: username.github.io (Replace username with your actual GitHub username)
- Required page: index.html (This will serve as the homepage of your personal site)
- Site URL: https://username.github.io/
- Important notes:
  - You can only have one personal main site like this per GitHub account.
  - This site acts as your primary portfolio or personal homepage on GitHub Pages.

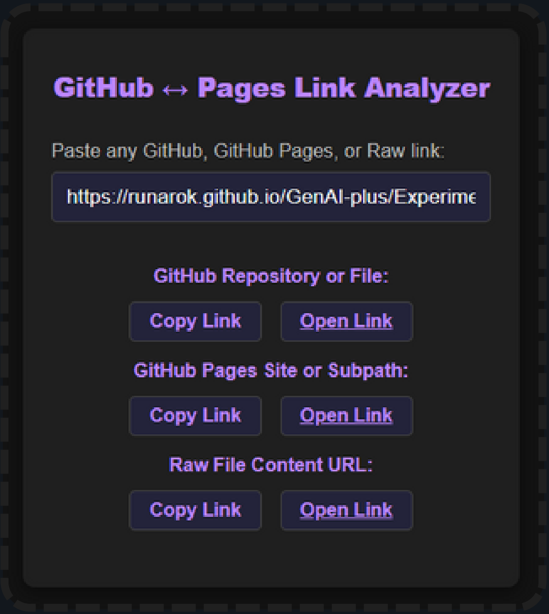
→ Back to Table of Contents

Project Sites (Multiple Repositories)

- Repository name: Any name (e.g., portfolio, blog, project-x)
- Required page: index.html (placed in the root or docs/ folder)
- Site URL: https://username.github.io/project-name
- Important notes:
    - Create as many project sites as you want this way.

File / Folder Structure	Example File	Link Example	Resulting URL
File in root folder	about.html	<a href="about.html">About</a>	https://username.github.io/repo-name/about.html
File in root folder	contact.html	<a href="contact.html">Contact</a>	https://username.github.io/repo-name/contact.html
File inside folder	pages/info.html	<a href="pages/info.html">Info</a>	https://username.github.io/repo-name/pages/info.html
File inside nested folder	blog/post1.html	<a href="blog/post1.html">Blog Post 1</a>	https://username.github.io/repo-name/blog/post1.html

Tip: GitlinkSyncer lets you input any GitHub URL and instantly get its raw, repo, and GitHub Pages versions—great for checking subpage links.


GitHub ↔ Pages Link Analyzer

Paste any GitHub, GitHub Pages, or Raw link:

https://runarok.github.io/GenAI-plus/Experime

GitHub Repository or File:

Copy Link | Open Link

GitHub Pages Site or Subpath:

Copy Link | Open Link

Raw File Content URL:

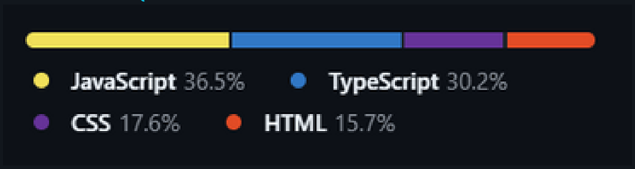
Copy Link | Open Link

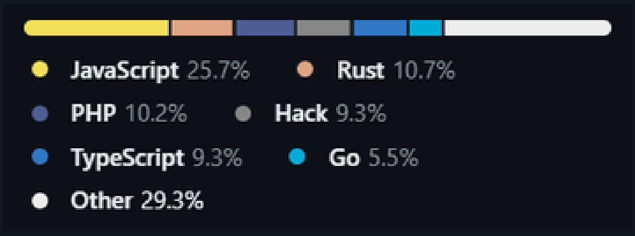
→ Back to Table of Contents

Preparing & Uploading HTML/CSS/JS to GitHub Properly

Step 1: Understand How GitHub Detects Languages

- On your repository page (top right section), GitHub automatically shows what languages are used and in what percentage.
  - Example breakdown:





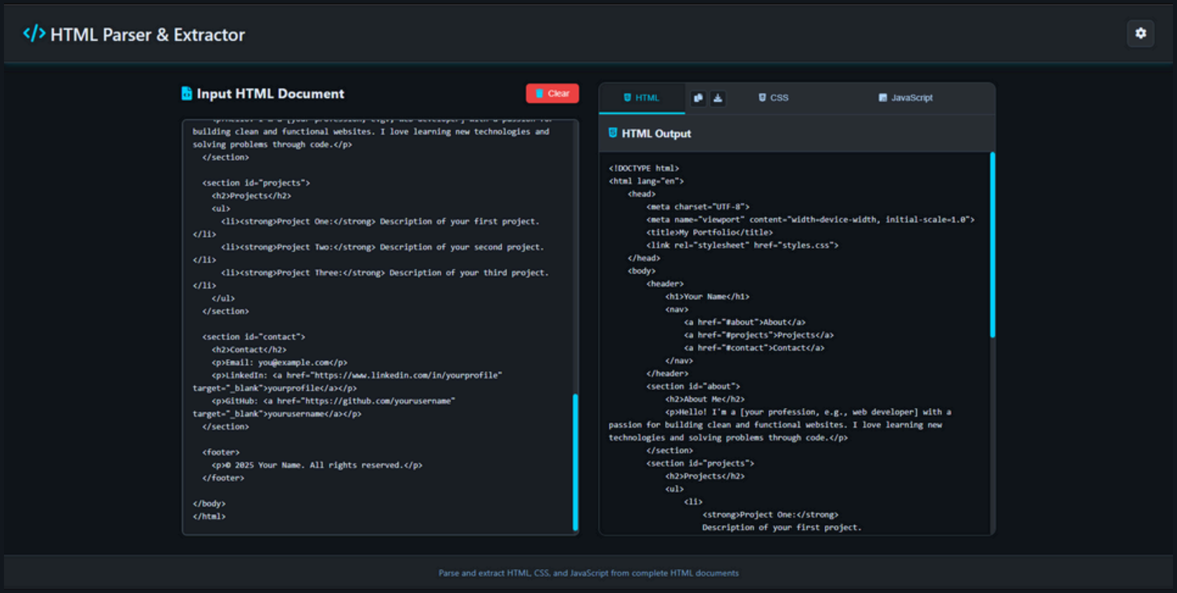
GitHub only detects languages based on the actual files present in your repo (e.g., .html, .css, .js).

Step 2: Know the Problem With Combined Files

- If your website code is in a single .html file (copied from somewhere with inline CSS and JS):
- GitHub will only count it as HTML
- CSS and JavaScript won't show up as separate languages
- Your repo will look incomplete or unstructured
- Not good for readability, reusability, or hosting dynamic sites

Step 3: Split Combined Code into HTML + CSS + JS

- Go to this tool: [HTML Splitter]
- Paste your entire HTML file (even if it includes CSS/JS) into the left side of the screen.
- The tool will automatically split the content into:
  - index.html , style.css , script.js



</> HTML Parser & Extractor
Input HTML Document
Clear
HTML
CSS
JavaScript
HTML Output
Parse and extract HTML, CSS, and JavaScript from complete HTML documents

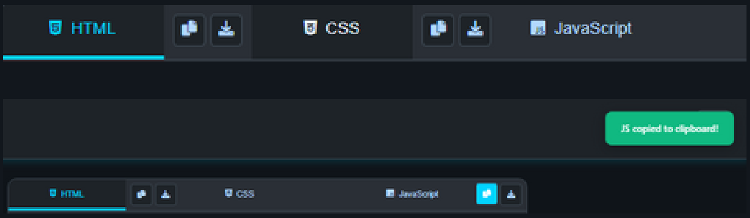
```
building clean and functional websites. I love learning new technologies and
solving problems through code.</p>
  </section>

  <section id="projects">
    <h2>Projects</h2>
    <ul>
      <li><strong>Project One:</strong> Description of your first project.
</li>
      <li><strong>Project Two:</strong> Description of your second project.
</li>
      <li><strong>Project Three:</strong> Description of your third project.
</li>
    </ul>
  </section>

  <section id="contact">
    <h2>Contact</h2>
    <p>Email: you@example.com</p>
    <p>LinkedIn: <a href="https://www.linkedin.com/in/yourprofile"
target="_blank">yourprofile</a></p>
    <p>GitHub: <a href="https://github.com/yourusername"
target="_blank">yourusername</a></p>
  </section>

  <footer>
    <p>© 2025 Your Name. All rights reserved.</p>
  </footer>

</body>
</html>
```

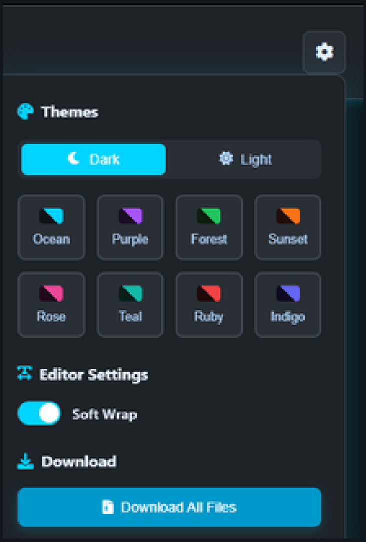
```
<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <title>My Portfolio</title>
        <link rel="stylesheet" href="styles.css">
    </head>
    <body>
        <header>
            <h1>Your Name</h1>
            <nav>
                <a href="#about">About</a>
                <a href="#projects">Projects</a>
                <a href="#contact">Contact</a>
            </nav>
        </header>
        <section id="about">
            <h2>About Me</h2>
            <p>Hello! I'm a [your profession, e.g., web developer] with a
passion for building clean and functional websites. I love learning new
technologies and solving problems through code.</p>
        </section>
        <section id="projects">
            <h2>Projects</h2>
            <ul>
                <li>
                    <strong>Project One:</strong>
                    Description of your first project.
```

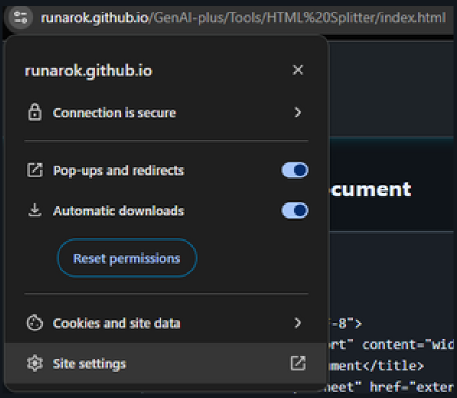
→ Back to Table of Contents

- You can hover over the right tabs (HTML / CSS / JS) to:
- Download each file individually
- OR click the Settings (top right) → Click "Download All"

You can also manually copy and paste each part if preferred.






runarok.github.io/GenAI-plus/Tools/HTML%20Splitter/index.html

runarok.github.io

Connection is secure

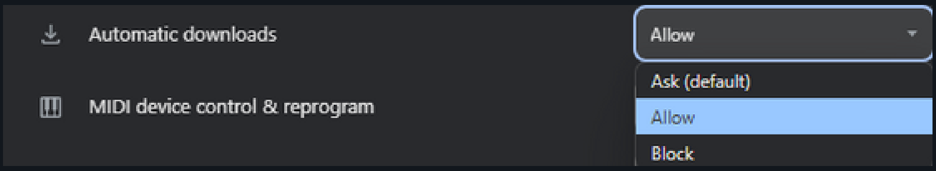
Pop-ups and redirects: on

Automatic downloads: on

Note : Allow Multi-file Downloads (if blocked)

- If you get blocked when trying to download all files:
- Look at the top-left of your browser's address bar
- Click the 🔒 (lock icon) → Site settings
- Find "Automatic downloads" and set it to Allow

Now the multi-file download should work.


Automatic downloads: Allow

Step 4: Organize and Upload to GitHub

- Create or open your GitHub repository
- Click "Add file" → Upload files
- Upload: • index.html → Root folder
  - script.js → Either root or js/ folder
  - style.css → Either root or css/ folder

If using folders, make sure your HTML file links correctly:

<link rel="stylesheet" href="css/style.css"> , <script src="js/script.js"></script>

As by default they are set to be in same root folder as the index.html

Result:
- GitHub now detects HTML, CSS, and JavaScript correctly
- Your repo appears professional and clean
- Hosting via GitHub Pages will work without rendering issues
- You can link between files and pages easily

→ Back to Table of Contents

Meta-Prompting

"Act as the best prompt engineer. Based solely on the information I provide, craft the most effective and optimized prompt possible, tailored specifically to my goal, audience, and the intended platform: [Platform]. Do not include code, explanations, or external content—only return the final, refined prompt I can use directly."

You can now swap [Platform] with anything—like "ChatGPT", "Midjourney", "Claude", "Notion AI", etc.

I used meta prompting to develop a playable Sudoku game.

The Example process is documented here: 🔗 Meta-prompting

You can play the final game here: 🎮 Sudoku Game

A handy collection of AI prompting cheat sheets I put together to help improve prompt crafting: GenAI+ Cheat Sheets

A categorized list of AI tools with brief features.

AI Coding & Assistant Platforms (Advanced reasoning, code generation, dev tools, web-based)

- GitHub Copilot – Code completion, IDE integration, real-time suggestions
- ChatGPT – General AI assistant, coding help, online chat interface
- Claude – Long-context reasoning, chat-based AI, strong document handling
- Gemini – Google's AI suite, coding & writing, deeply web-integrated
- Mistral – Open-source models, fast local deployment, modular LLMs

Lightweight AI Builders (Prompt chaining, mini-app creation, lightweight AI systems)

- Lovable – Emotional UX, playful chat experiences, minimal design
- Bolt – Modular prompt workflows, Notion-compatible, logic templates
- Rocket – Quick prompt creation, deploy to web, templates for interactive use
- V0 (v0.dev) – AI-powered UI builder from text, deploy via Vercel

Command-Line & Interface Tools (Terminal tools, AI-enhanced UI, dev productivity boosters)

- Windsurf – AI-enhanced IDE, autocomplete, inline code insight
- Trae – CLI-first tool for managing prompts and AI logic
- Warp – Next-gen terminal with AI command search & suggestions
- Cursor – AI-native code editor, autocomplete, debugging
- Raycast AI – Spotlight-like productivity commands + AI assistant
- CommandBar AI – AI command palettes for SaaS apps, in-app UX layer