

Lesson 01: Publishing a Website with GitHub Pages

Start



Learning Objectives



Create a public
GitHub repository
for a website



Create and edit an
index.html file




Publish a website
using GitHub
Pages



Update website
content and
redeploy it



Explain, in simple
terms, how
GitHub Pages
works

 These objectives describe what you will be able to do, not what you need to memorize.

Deliverables



✓ A public
GitHub repository




✓ A published
website
accessible
through a GitHub
Pages URL



✓ An index.html
file with
customized
content



✓ At least one
committed
update to the
website

 Your website will be public and accessible online.



AI Learning Support for This Lesson

In this lesson, the AI tutor is here to support you.

You are not expected to know everything yet — we have prepared a guided prompt that you can use to help you:

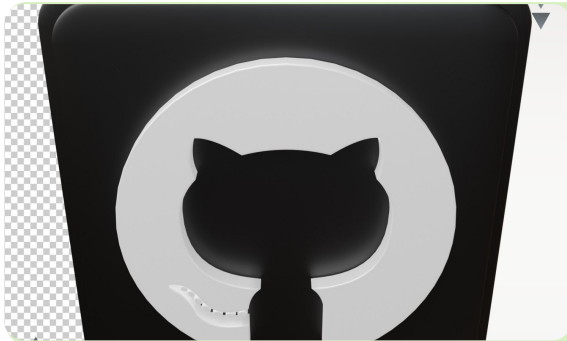
- Take your first steps with GitHub
- Create your first HTML “Hello World” page
- Navigate GitHub and find the right sections and settings
- Successfully publish your first website for free using GitHub Pages

💡 Use the AI as a guide, not as a shortcut.

You are always encouraged to read, understand, and ask follow-up questions: **Lesson 01 Prompt**



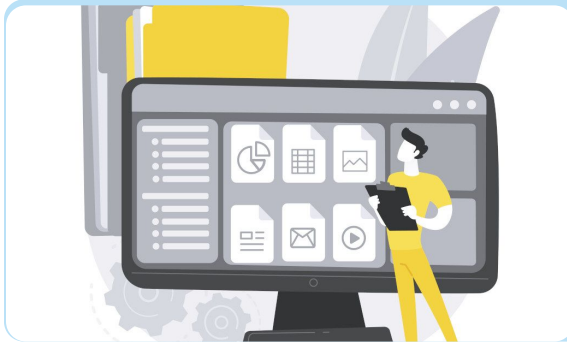
GitHub: Free code repository for your static website



GitHub is an online platform to:

- Store code
- Track changes
- Collaborate

Used by developers, designers, and companies



What Is a Repository?

- A repository (repo) is a project folder.

It contains:

- Files (HTML, CSS, JS)
- Images
- History of changes



Why We Use GitHub Pages:

- Free hosting
- No servers to configure

Perfect for:

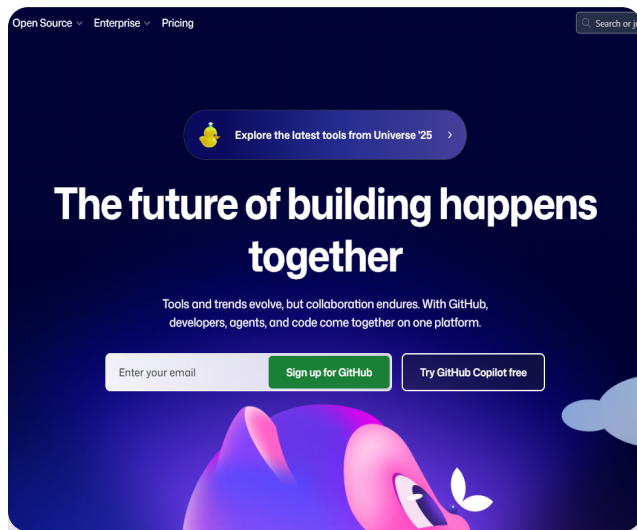
- Portfolios
- Prototypes
- AI-generated websites



3R

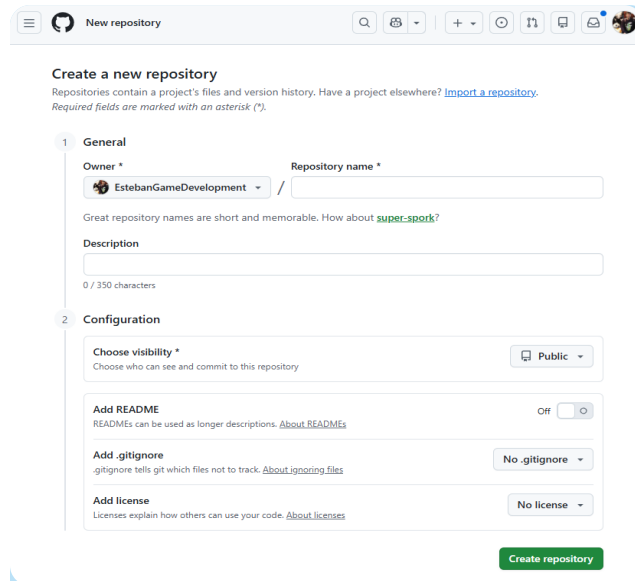


Creating a GitHub Account



Your GitHub Identity

1. Go to github.com
2. Sign up with email
3. Choose username (important!)

The image shows the "New repository" form on GitHub. It is divided into two sections: "General" and "Configuration". In the "General" section, there is a dropdown for "Owner" (set to "EstebanGameDevelopment"), a text field for "Repository name", and a text field for "Description" (with a character count of 0 / 350). In the "Configuration" section, there is a "Choose visibility" dropdown (set to "Public"), an "Add README" section with a toggle (set to "Off"), an "Add .gitignore" section with a dropdown (set to "No .gitignore"), and an "Add license" section with a dropdown (set to "No license"). A green "Create repository" button is at the bottom right.

Creating a new repository

1. Click New repository
2. Repository name: my-first-website
3. Set to Public
4. ✓ Check Add README file



README = Project Description

README explains:

- What the project is
- What it does

Written in plain text / Markdown



3C



Activating GitHub Pages

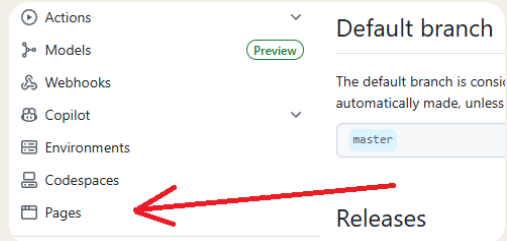
Turning Code into a Website

GitHub Pages takes:


- Files in your repository

And publishes them as:

- A real website



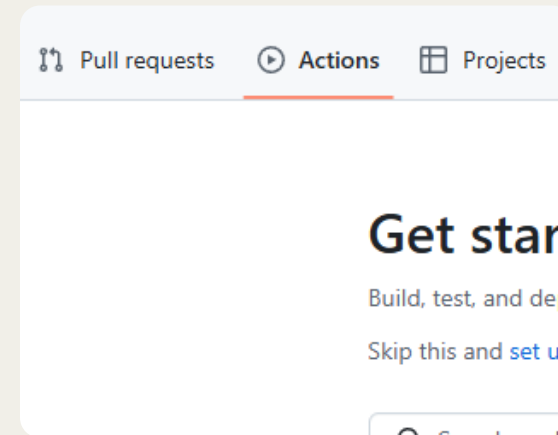
GitHub Actions works automatically

- GitHub runs an automated process
- This process:
 - Builds your website
 - Publishes it online
- Name to remember:  **GitHub Actions**



Enable Github Pages

- 1) Go to Settings
- 2) Open Pages
- 3) Source:
 - Branch: main
 - Folder: /root
- 4) Save



6



The Website Entry Point

What Is index.html?

The Website Entry Point



First Website File

Create index.html



Simplest Possible Website

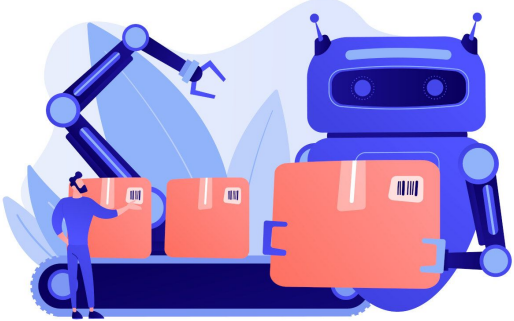
Hello World HTML



6



Publishing via Github Actions



Github Actions do the automatic publishing process

GitHub Actions:

- Detects changes
- Rebuilds website
- Publishes automatically

pages-build-deployment
on: dynamic



Select the tab and watch the process

1. Go to Actions
2. Yellow ⌚ = working
3. Green ✓ = published

Hello World

Your Website is Live:

1. Go to Settings → Pages
2. Click the URL

You website is a real website:

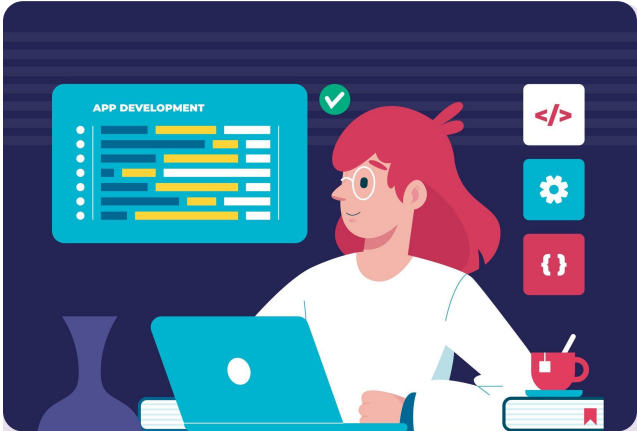
1. Public
2. Accessible from anywhere
3. Hosted for free.



3R



Editing Files in the Browser



GitHub Is Also an Editor

1. Open index.html
2. Click  Edit
3. Modify content

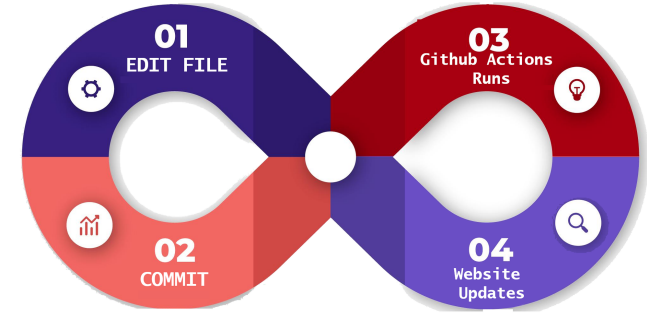
Content



Saving Changes = Commit

A commit:

- Saves changes
- Describes what changed
- Click Commit changes



The Loop

Process:

1. Edit file
2. Commit
3. GitHub Actions runs
4. Website updates



3C



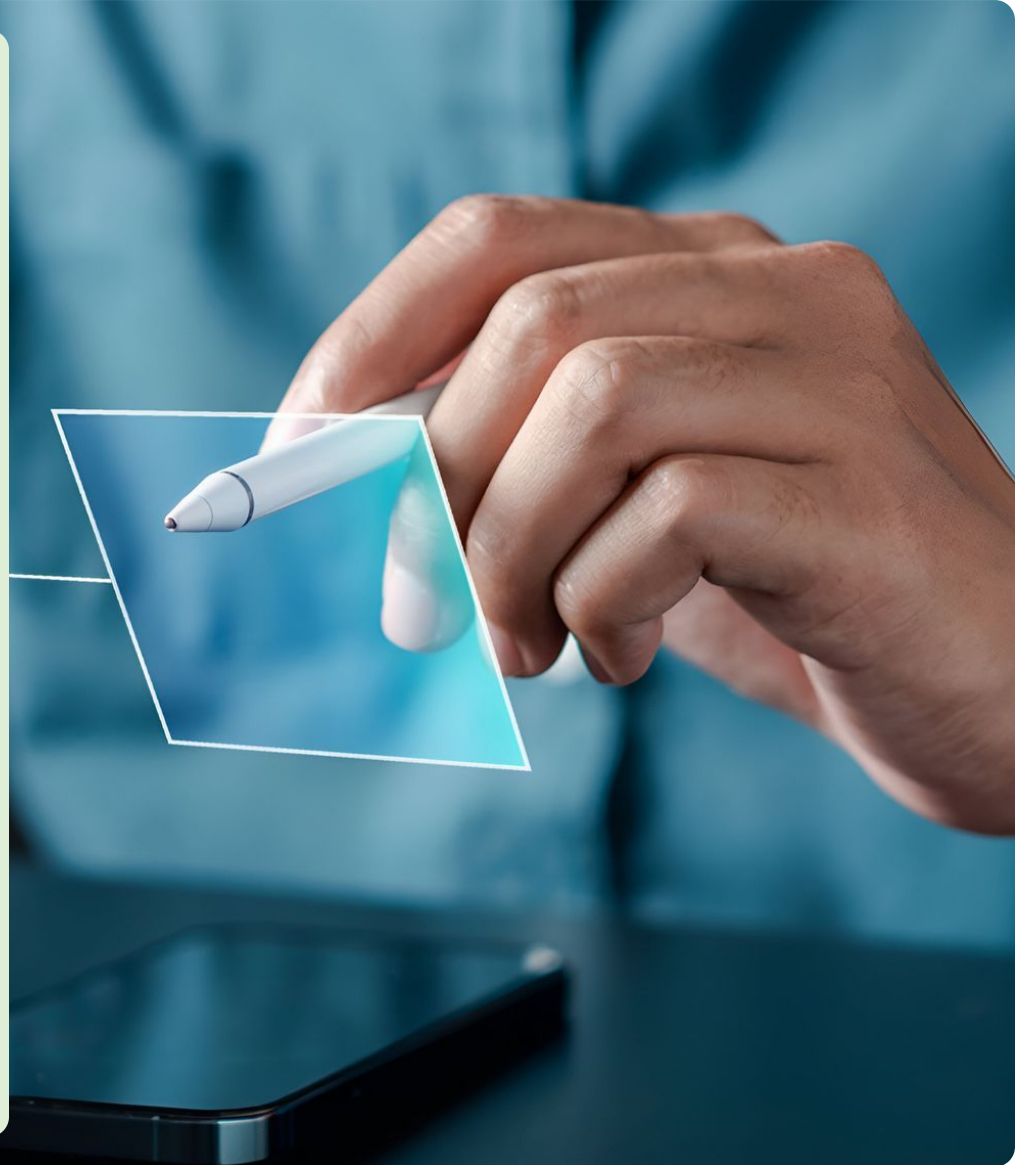


```
<html>  
  <header>My header</header>  
  <body>My content for hello world</body>  
</html>
```

How GitHub Pages Works

Mental model:

- 📁 GitHub stores it
- ⚙️ GitHub Actions publishes it
- 🌐 GitHub Pages shows it



Knowledge Test Questions

Publishing a Website with GitHub Pages



Purpose of GitHub Pages

Question



Website Entry Point

Question



Publishing Workflow

Question



Lesson Deliverable Understanding

Question





What is the main purpose of GitHub Pages in this course?

To edit HTML files directly
in the browser

To store backups of
website files

To publish repository files
as a public website



Which of the following best describes a successful completion of Lesson 1?

Creating a private repository with multiple files

Having a public website accessible through a GitHub Pages URL

Writing complex HTML and CSS code



Why is the file named `index.html` important when using GitHub Pages?

It is the default file GitHub Pages loads as the website homepage

It is required to enable GitHub Actions

It contains special GitHub configuration settings



What happens after you commit changes to a file in a GitHub repository with GitHub Pages enabled?

GitHub Actions automatically builds and publishes the website

You must manually upload the files to a server

The website updates immediately without any process

Help us to improve

With your feedback we will keep improving for you to get the best results



Which part of this lesson was the most confusing or unclear for you, and why?

Write your answer here.

Send

Did you feel the pace of the lesson was appropriate for your level of experience?

Write your answer here.

Send

What change would most improve this lesson for future students?

Write your answer here.

Send

