
IIC2026

Ayudantía 7 - Tone.js y GitHub Pages

— Javiera Azócar y Camila Basulto —

Tone.js

¿Qué es?

Es una librería de JavaScript que permite crear música interactivamente desde el browser



¿Cómo lo instalamos?

```
4 <head>
5   <meta charset="UTF-8">
6   <title>Ayudantia 7: Tone</title>
7   <link rel="stylesheet" href="tone.css">
8   <script src="https://cdnjs.cloudflare.com/ajax/libs/tone/15.1.3/Tone.js"></script>
9 </head>
```

En el siguiente link se puede encontrar formas de acceder a la librería y ver otras versiones
<https://cdnjs.com/libraries/tone>

¿Qué se puede hacer en Tone.js?

- Reproducir notas musicales
- Hacer loops
- Hacer efectos
- Sintetizar
- Ocupar samples



Instrumentos

Classes - Instrument

☐ AMSynth

☐ MembraneSynth

☐ NoiseSynth

☐ Sampler

☐ DuoSynth

☐ MetalSynth

☐ PluckSynth

☐ Synth

☐ FMSynth

☐ MonoSynth

☐ PolySynth

¿Qué hace Synth?

Synth se compone simplemente de un OmniOscillator enrutado a través de un AmplitudeEnvelope. Al igual que Tone.FMSynth, Tone.AMSynth and Tone.NoiseSynth solo pueden emitir un sonido a la vez.

```
+-----+ +-----+  
| OmniOscillator +>--> AmplitudeEnvelope +>--> Output  
+-----+ +-----+
```

¿Qué hace Synth?

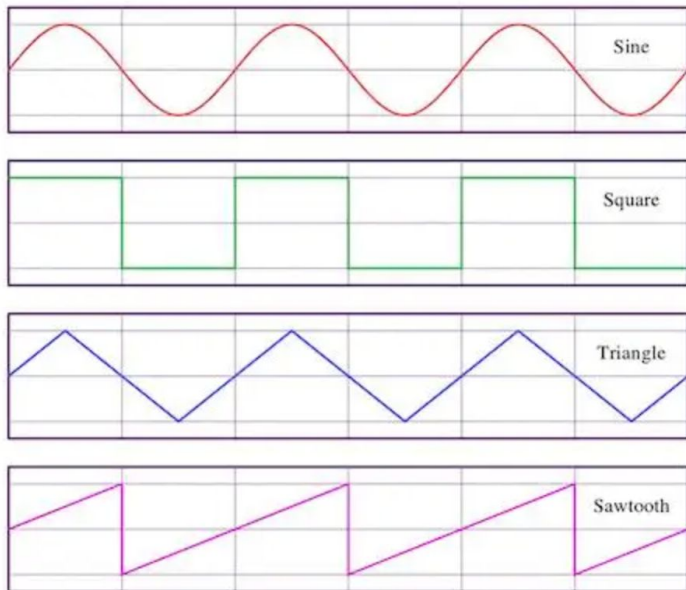
Synth se compone simplemente de un OmniOscillator enrutado a través de un AmplitudeEnvelope. Al igual que Tone.FMSynth, Tone.AMSynth and Tone.NoiseSynth solo pueden emitir un sonido a la vez.

```
+-----+ +-----+  
| OmniOscillator +>--> AmplitudeEnvelope +>--> Output  
+-----+ +-----+
```

Pero... qué significa eso?

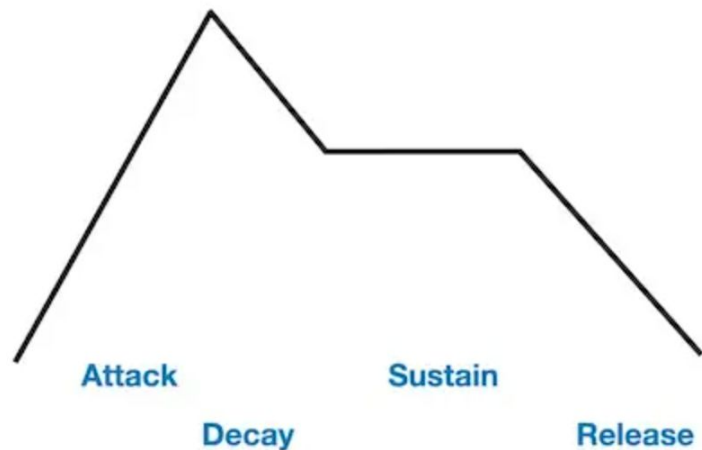
Oscilador

Genera el sonido



Amplitud Envelop

Forma el sonido



¿Cómo reflejo esto en el código?

Con `const synth = new Tone.Synth()` se genera la instancia pero para que se conecte al parlante del computador es necesario poner `.toDestination()`

De esta forma el código quedaría:

```
const synth = new Tone.Synth().toDestination();
```

¿Qué hago para que suene?

`triggerAttackRelease` permite que se emita el sonido pero si escribo solo `synth.triggerAttackRelease("C4", "8n");` ocurre lo siguiente:

⚠ ▶ The AudioContext is "suspended". [Debug.ts:99](#)
Invoke `Tone.start()` from a user action to start the audio.

esto se debe a que los browser no quieren que suenen las páginas cuando recién se abren por lo que suspenden el sonido. Para solucionarlo se puede hacer un botón por ejemplo para que gatille el sonido

Quedaría de la siguiente forma

```
1  const synth = new Tone.Synth().toDestination();
2
3  document.getElementById("boton-musica").addEventListener("click", () => {
4    if (Tone.context.state !== "running") {
5      Tone.start();
6    }
7    synth.triggerAttackRelease("C4", "8n");
8  });
```

Otro instrumento: Sampler

Al pasar un objeto que asigne el tono de la nota o el valor mediante un URL, luego se podrá activar el ataque y liberación de esa nota como otros instrumentos.

```
1  const sampler = new Tone.Sampler({
2    urls: {
3      C4: "beethoven.mp3", // Archivo de sonido
4    },
5    release: 1,
6    baseUrl: "./", // Asegura que busque el archivo en la misma carpeta
7  }).toDestination();
8
9  // Evento de clic para el botón
10 document.getElementById("boton-musica").addEventListener("click", () => {
11   // Verifica si el contexto de audio no está activo
12   if (Tone.context.state !== "running") {
13     Tone.start().then(() => {
14       sampler.triggerAttackRelease("C4", 10);
15     });
16   }
17   sampler.triggerAttackRelease("C4", 10);
18
19 });
```

Efectos

- ☐ AutoFilter
- ☐ AutoPanner
- ☐ AutoWah
- ☐ BitCrusher
- ☐ Chebyshev
- ☐ Chorus
- ☐ Distortion
- ☐ FeedbackDelay
- ☐ Freeverb
- ☐ FrequencyShifter

- ☐ Freeverb
- ☐ FrequencyShifter
- ☐ JCREverb
- ☐ Phaser
- ☐ PingPongDelay
- ☐ PitchShift
- ☐ Reverb
- ☐ StereoWidener
- ☐ Tremolo
- ☐ Vibrato

FeedbackDelay

```
1  const synth = new Tone.Synth();
2
3  const feedbackDelay = new Tone.FeedbackDelay();
4
5  synth.connect(feedbackDelay);
6  feedbackDelay.toDestination();
7
8  document.getElementById("boton-musica").addEventListener("click", () => {
9      if (Tone.context.state !== "running") {
10         Tone.start();
11     }
12     synth.triggerAttackRelease("C4", "8n");
13 });
```

FeedbackDelay

Con el código anterior el cambio es muy sutil, para prolongar las pausas se tienen que poner parámetros

1º Parámetro: tiempo de retardo que es el retraso aplicado a la señal entrante.

2º Parámetro: retroalimentación que es la cantidad de señal efectuada que se retroalimenta a través del retardo.

```
17  const feedbackDelay = new Tone.FeedbackDelay("8n", 0.5).toDestination();
18  const tom = new Tone.MembraneSynth({
19    |   octaves: 4,
20    |   pitchDecay: 0.1
21  | }).connect(feedbackDelay);
22
23  document.getElementById("boton-musica").addEventListener("click", () => {
24    |   if (Tone.context.state !== "running") {
25    |     |   Tone.start();
26    |   }
27    |   tom.triggerAttackRelease("A2", "32n");
28  | });|
```


También se pueden ocupar varios efectos en simultáneo!

```
27 const player = new Tone.Player({
28   url: "beethoven.mp3",
29   loop: true,
30   autostart: true,
31 });
32
33 const distortion = new Tone.Distortion(0.4).toDestination();
34 const filter = new Tone.Filter(400, "lowpass").toDestination();
35
36 player.connect(distortion);
37 player.connect(filter);
38
39 document.getElementById("boton-musica").addEventListener("click", () => {
40   if (Tone.context.state !== "running") {
41     Tone.start().then(() => {
42       player.start();
43     });
44   }
45   player.start();
46
47 });
```

Links importantes

- Documentación de Tone.js

<https://tonejs.github.io/docs/15.0.4/classes/Abs.html>

GitHub Pages

Para “hostear” páginas

Primero, GitHub

“GitHub es una forja para alojar proyectos utilizando el sistema de control de versiones Git. Se utiliza principalmente para la creación de código fuente de programas de ordenador.”



Se crean repositorios que son las **“carpetas” contenedoras** en las cuales se suben los archivos (de cualquier extensión: **.pdf, .xlsx, .png, etc.**; de cualquier lenguaje: **.py, .js, .html, .c, etc.**) necesarios para ejecutar, visualizar y utilizar cualquier proyecto que creamos.

Si aún no tienen cuenta de GitHub, deben crear una para utilizar GitHub Pages. Se recomienda utilizar el correo @uc para así optar a beneficios de estudiante.

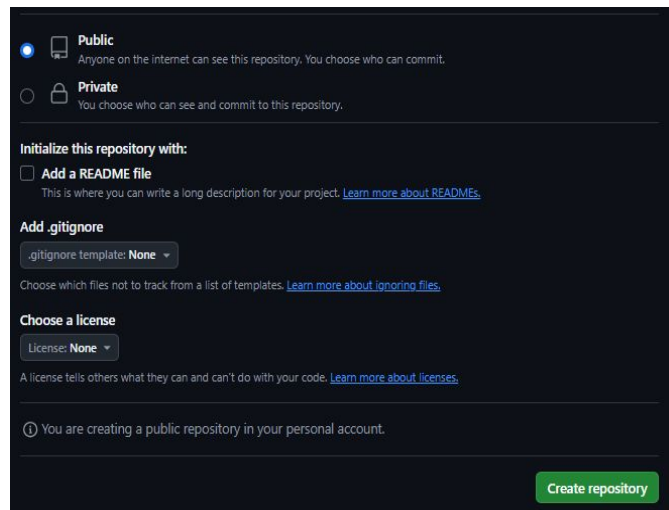
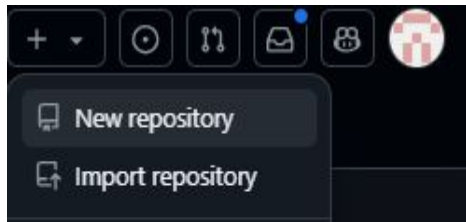
GitHub Pages

"Directamente alojado desde tu repositorio de GitHub. Solo edita, haz push, y tus cambios estarán en vivo."

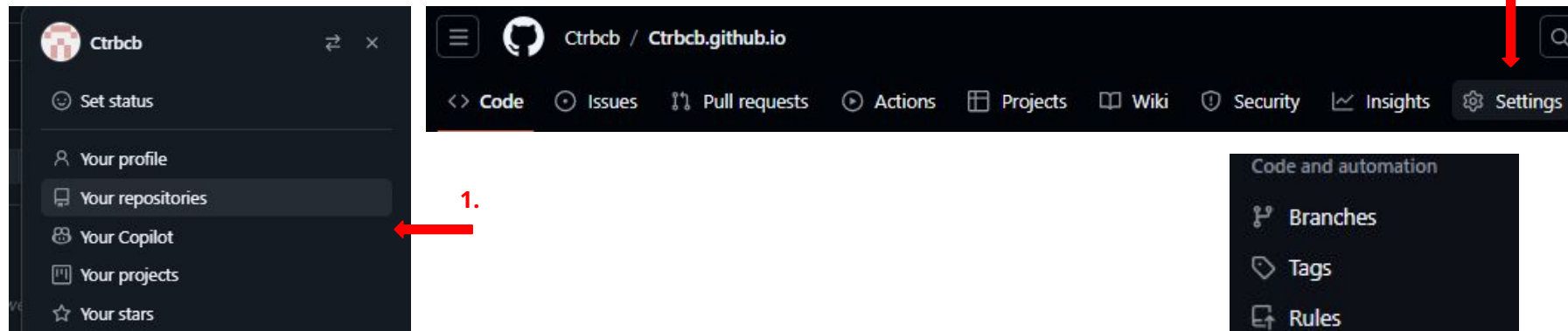
Cómo crear una página

0. Tener cuenta de GitHub. Para crear una cuenta <https://github.com/signup>

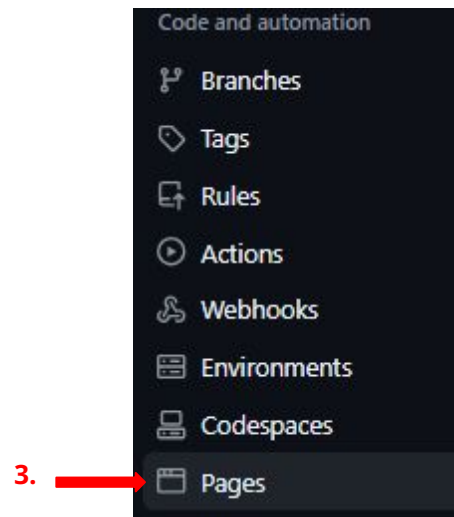
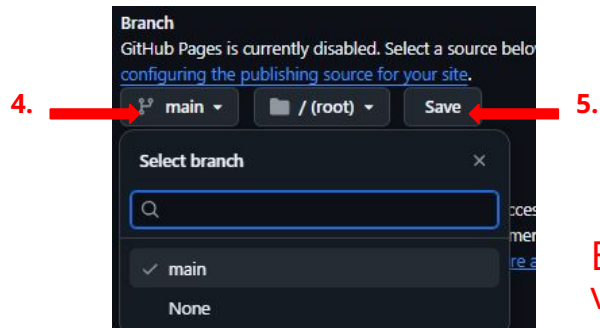
1. Crear un repositorio con nombre
<tu_usuario>.github.io



2. Entrar al repositorio, ir al apartado [Settings](#), ahí presionar [Pages](#).



3. En Branch seleccionar “[main](#)” y luego [Save](#).



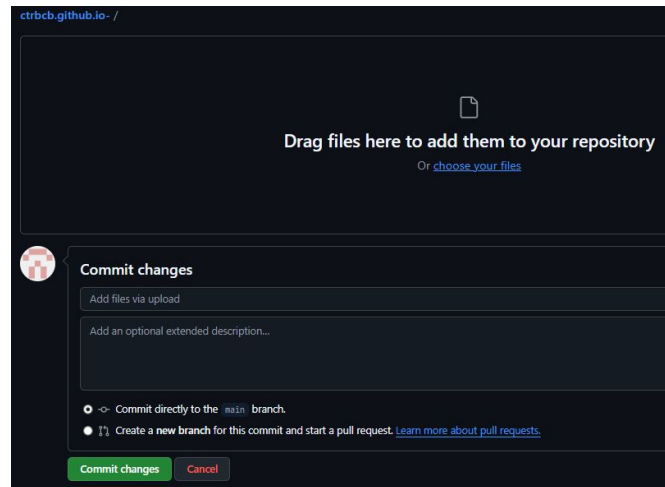
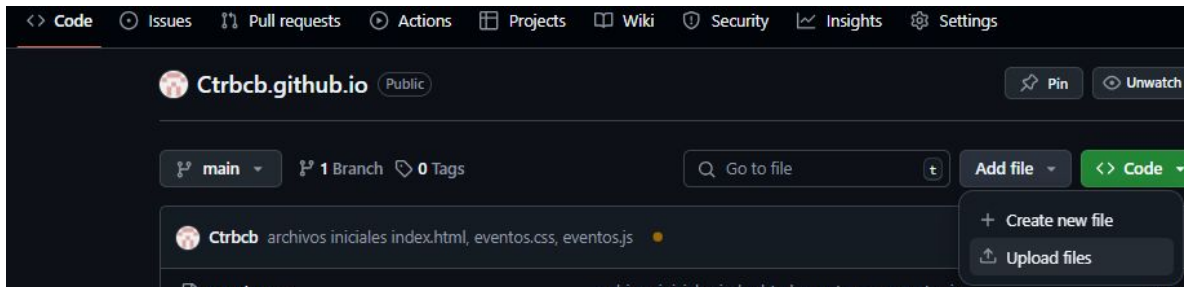
En este punto ya está “creada” su página, pero no se puede visitar si no tiene ningún archivo o nada creado para mostrar.

Cómo subir archivos

Una vez tengan su archivo .html con código y sus respectivos .js, .css (si es que son necesarios), hay dos opciones para cargar los archivos al repositorio:

1. Forma simple, desde la misma página del repositorio

Hacer click en [Add file](#) y [Upload files](#), aquí se pueden arrastrar o seleccionar los archivos que quieran subir.

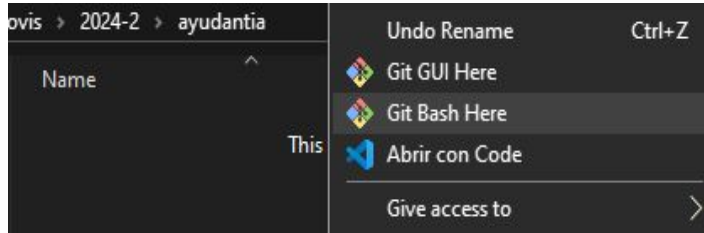


Cada vez que se actualiza el repositorio se hace un commit, esto sirve luego para llevar un registro de las ediciones. Elijan un nombre sugerente de qué se está realizando en el cambio para que luego lo puedan identificar si quieren volver en las versiones. Para guardar hacer click en [Commit changes](#).

1. Forma un poco compleja, con Git (opcional)

Git es una herramienta de GitHub que permite clonar repositorios en su ordenador para manejar los archivos localmente y desde ahí gestionar los cambios. Git se descarga e instalar, pueden utilizar el link <https://git-scm.com/downloads>.

En una carpeta, con click derecho se selecciona [Git Bash Here](#), se abrirá la terminal y deben escribir `git clone <link_repo>`



```
MINGW64:/d/Repos/infovis/2024-2/ayudantia 7/Ctrbcb.github.io
dell@dell-PC MINGW64 /d/Repos/infovis/2024-2/ayudantia 7
$ git clone https://github.com/Ctrbcb/Ctrbcb.github.io.git
Cloning into 'Ctrbcb.github.io'...
warning: You appear to have cloned an empty repository.
dell@dell-PC MINGW64 /d/Repos/infovis/2024-2/ayudantia 7
$ cd Ctrbcb.github.io/
```

Luego para entrar a la carpeta: `cd <nombre del repo>` (para entrar a la carpeta creada del repositorio). Deben copiar sus archivos dentro de la carpeta o crearlos ahí mismo.

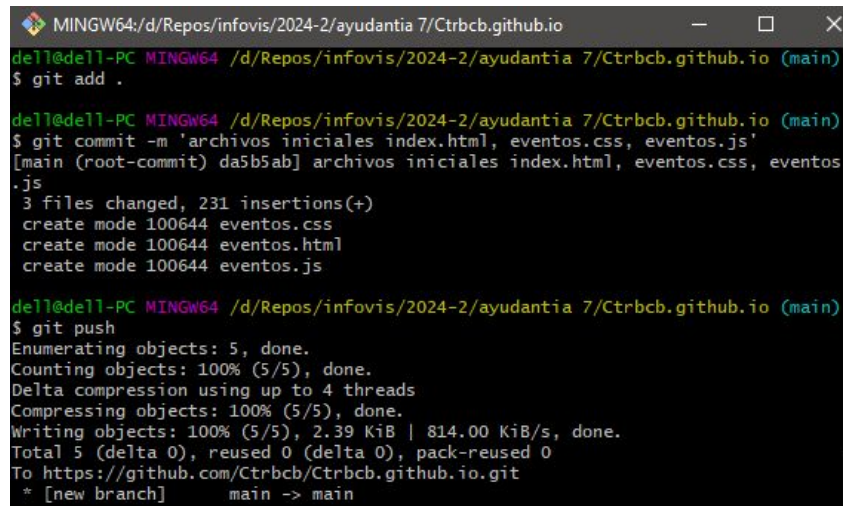
Cuando tengan su código listo, hay que subirlo al repositorio, para esto en la terminal de Git se ingresan los siguientes comandos:

`git add .`

`git commit -m '<descripción_del_commit>'`

`git push`

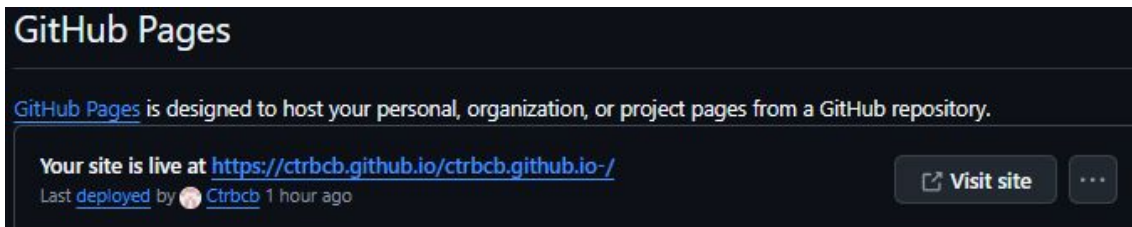
En este punto ya están subidos los cambios al repositorio, pueden verlo en la web entrando al repositorio desde GitHub.

A screenshot of a Windows terminal window with a dark background. The title bar shows the path 'MINGW64:/d/Repos/infovis/2024-2/ayudantia 7/Ctrbcb.github.io'. The terminal shows the following commands and output:
\$ git add .
\$ git commit -m 'archivos iniciales index.html, eventos.css, eventos.js'
[main (root-commit) da5b5ab] archivos iniciales index.html, eventos.css, eventos.js
3 files changed, 231 insertions(+)
create mode 100644 eventos.css
create mode 100644 eventos.html
create mode 100644 eventos.js
\$ git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 2.39 KiB | 814.00 KiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Ctrbcb/Ctrbcb.github.io.git
* [new branch] main -> main

Disclaimer: recomiendo ver un tutorial para la instalación de Git ya que puede complicarse y resultar en errores, pero esta herramienta es totalmente **OPCIONAL**, por lo que pueden usar el primer método. Si van a seguir programando generalmente, recomiendo utilizarlo ya que es común en el mundo de la programación. Además, este es el tutorial más básico que encontrarán para subir archivos :))

Cómo ver la página

Luego de haber subido los archivos a su repositorio de cualquiera de las formas presentadas, pueden entrar a su página. Para esto deben volver a entrar al repositorio, ir al apartado [Settings](#), ahí presionar [Pages](#) y luego [Visit site](#).



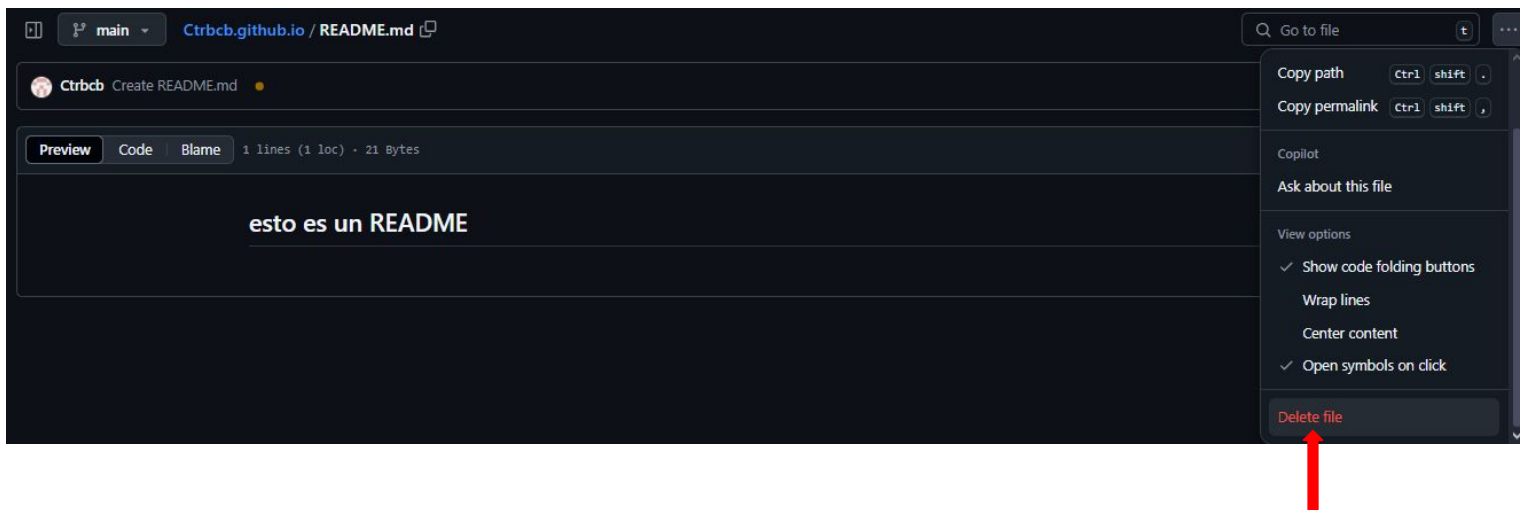
Como la página se hostea en GitHub, cualquiera que tenga este link podrá ver la página del repositorio. También lo pueden guardar para acceder más rápidamente.

Posibles problemas

Si al entrar en su página no sale nada o no carga, estas son las posibles razones y sus soluciones:

- **El usuario en el nombre del repositorio está mal escrito:** deben verificar que la parte del nombre de usuario en el nombre del repositorio sea EXACTAMENTE su nombre de usuario, incluyendo las mayúsculas/minúsculas (es case sensitive).
- **GitHub no reconoce el nombre de su .html:** el archivo de su página DEBE llamarse index.html, ya que esto es lo que busca el sistema para cargar su archivo.
- **Hay un error de código en sus archivos:** recomiendo probar su página localmente con las herramientas vistas en una ayudantía anterior (Live Server o localhost) para asegurarse que funciona la página y no hay errores de código. Si esta no funciona localmente, no funcionará con GitHub Pages.

- **Tienen un archivo README.md en su repositorio:** el sistema busca archivos index.html y README.md para cargar su página (es decir, también pueden hacer su código en el README), por lo que deben borrar este último. Pueden hacerlo desde la web (haciendo click en su archivo y lo que muestra la imagen) o borrarlo de la carpeta de su repositorio localmente y subir los cambios (con add ., commit y push) si es que usan Git.



IIC2026

Ayudantía 7 - Tone.js y GitHub Pages

— Javiera Azócar y Camila Basulto —
