

CONFIGURACIÓN DEL ENTORNO LOCAL Y SUBIDA DEL CÓDIGO A GITHUB



¿Cuál es nuestro objetivo para este módulo?

Exploración de HTML, CSS, JavaScript y otros archivos que se utilizan para crear una página web que aloja un juego.

¿Qué logramos en clase el día de hoy?

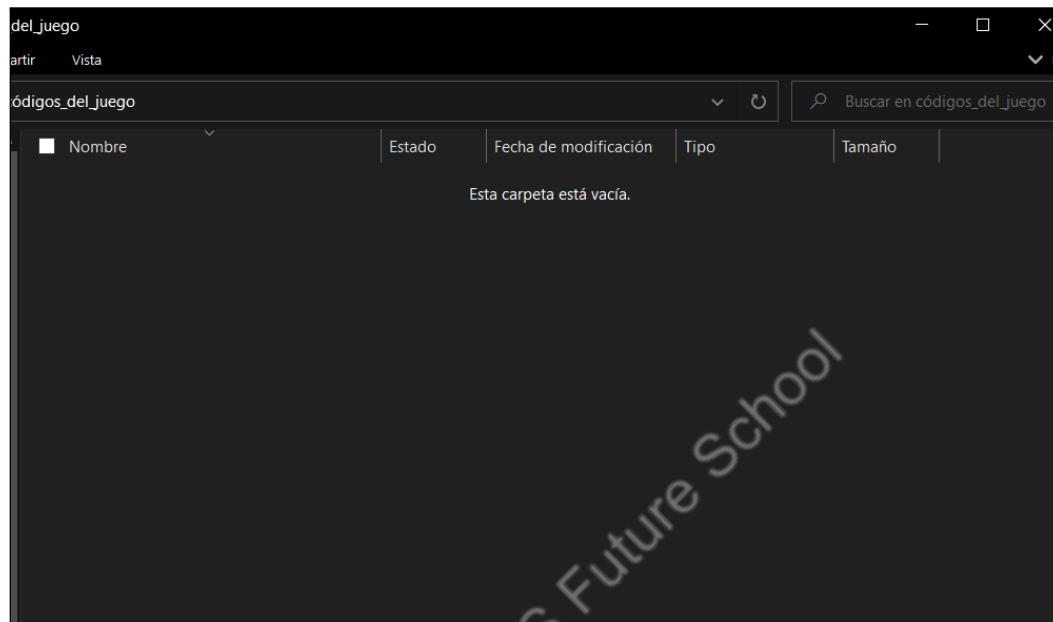
- Aprendimos sobre el papel de HTML, CSS y JavaScript en el diseño de una página web que puede albergar un juego.
- Aprendimos sobre las bibliotecas de JavaScript y cómo usarlas dentro de nuestro código.
- Aprendimos a configurar un entorno local para escribir y probar código usando JavaScript.
- Subimos el código en GitHub.

¿Qué conceptos y bloques de código vimos el día de hoy?

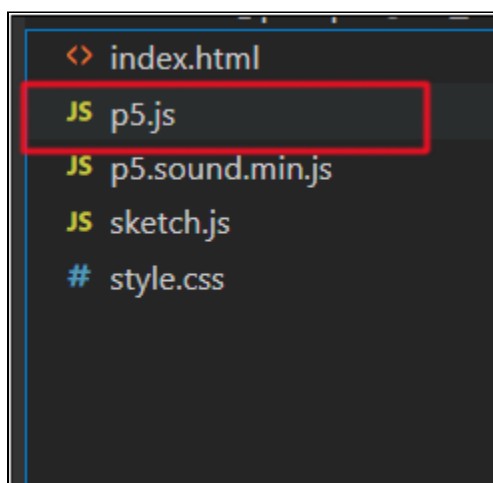
- El papel de los archivos de la biblioteca de JavaScript.
- El concepto de la plataforma de alojamiento online GitHub.
- El entorno local VSC (Visual Studio Code).

¿Cómo hicimos las actividades?

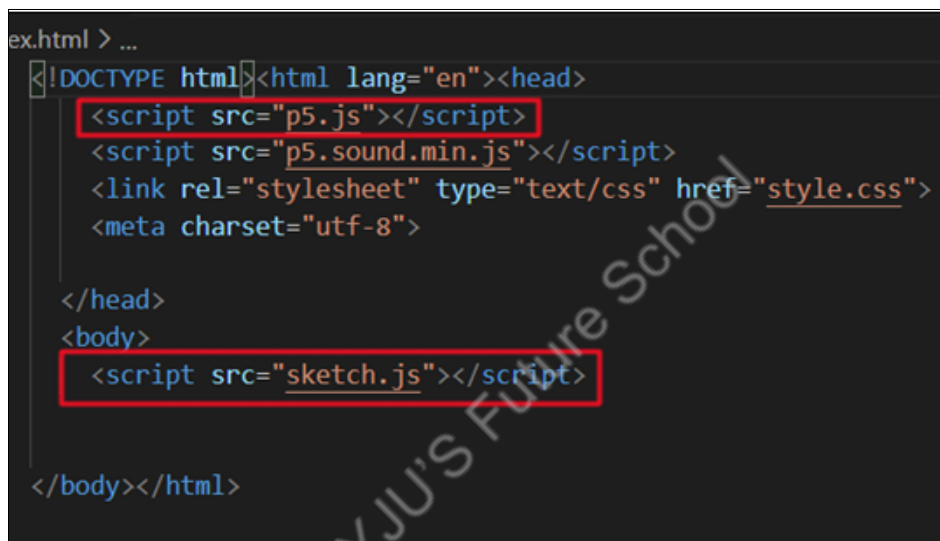
1. Crear una carpeta `códigos_del_juego` en tu computadora. Después, crear una carpeta separada para guardar el código de cada actividad de clase y el proyecto.



2. Usar las bibliotecas predefinidas para realizar funcionalidades relacionadas con los sprites en `p5.play.js`. Los programas JavaScript incluyen muchas bibliotecas; es decir, colecciones de código que pueden ser reutilizadas por otros programas en su código.

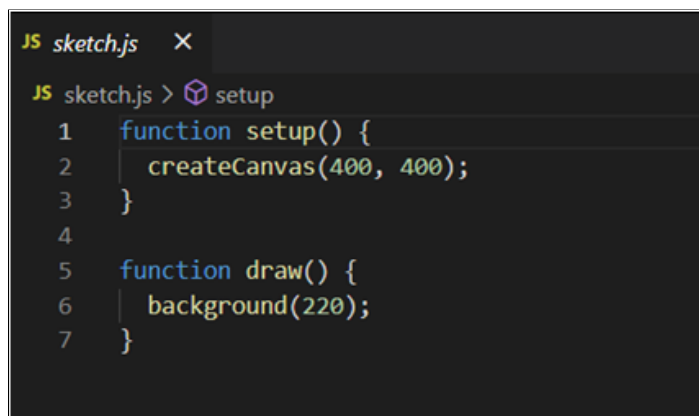


3. Conectar sketch.js y archivos de soporte (bibliotecas) incluyendo todos los archivos en index.html.
- El código JavaScript se puede ejecutar en el navegador con la ayuda de index.html.
 - Dentro de un par de etiquetas `<head></head>` le indicamos a la computadora las diferentes bibliotecas que queremos cargar.
 - Dentro de un par de etiquetas `<body></body>` le indicamos a la computadora qué mostrar en la página web. Se agrega sketch.js en la etiqueta del cuerpo.
 - Dentro de un par de etiquetas `<script></script>` proporcionamos la ubicación de cualquier código JavaScript.



```
ex.html > ...
<!DOCTYPE html><html lang="en"><head>
  <script src="p5.js"></script>
  <script src="p5.sound.min.js"></script>
  <link rel="stylesheet" type="text/css" href="style.css">
  <meta charset="utf-8">
</head>
<body>
  <script src="sketch.js"></script>
</body></html>
```

- Sketch.js incluye el código ejecutable.



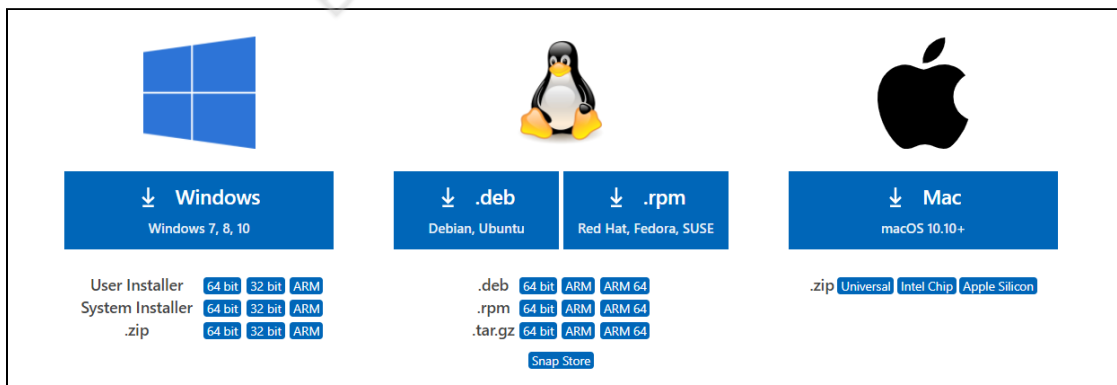
```
JS sketch.js X
JS sketch.js > setup
1 function setup() {
2   createCanvas(400, 400);
3 }
4
5 function draw() {
6   background(220);
7 }
```

- El archivo Style.css se usa para cambiar/agregar estilo a la página web, como

el tipo de color de fuente, el color de fondo, la posición de elementos como el botón y las imágenes.

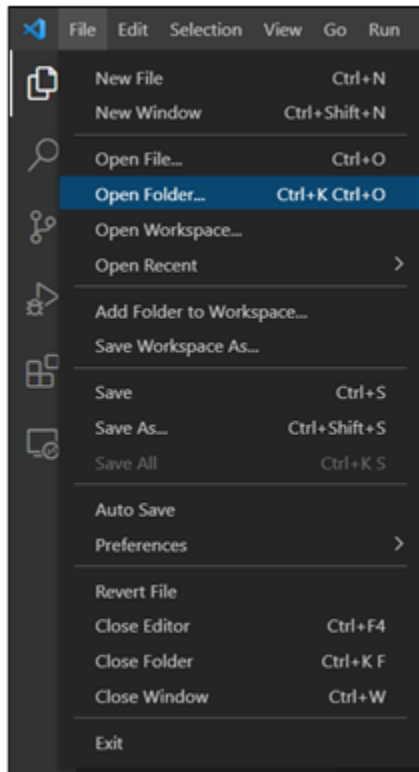
```
# style.css > html
1  html, body {
2    margin: 0;
3    padding: 0;
4  }
5  canvas {
6    display: block;
7  }
8
```

4. Visual Studio Code Editor nos permite escribir el código en nuestra computadora. Tiene varios beneficios diferentes a Code.org.
- No depende de Internet.
 - El tamaño del canvas se puede cambiar según el requisito, mientras que en Code.org el tamaño del canvas se fija en 400 x 400
 - Al verificar el output, el reproductor no tendrá la opción de ver el código.
 - El código completo se guarda en nuestra computadora.

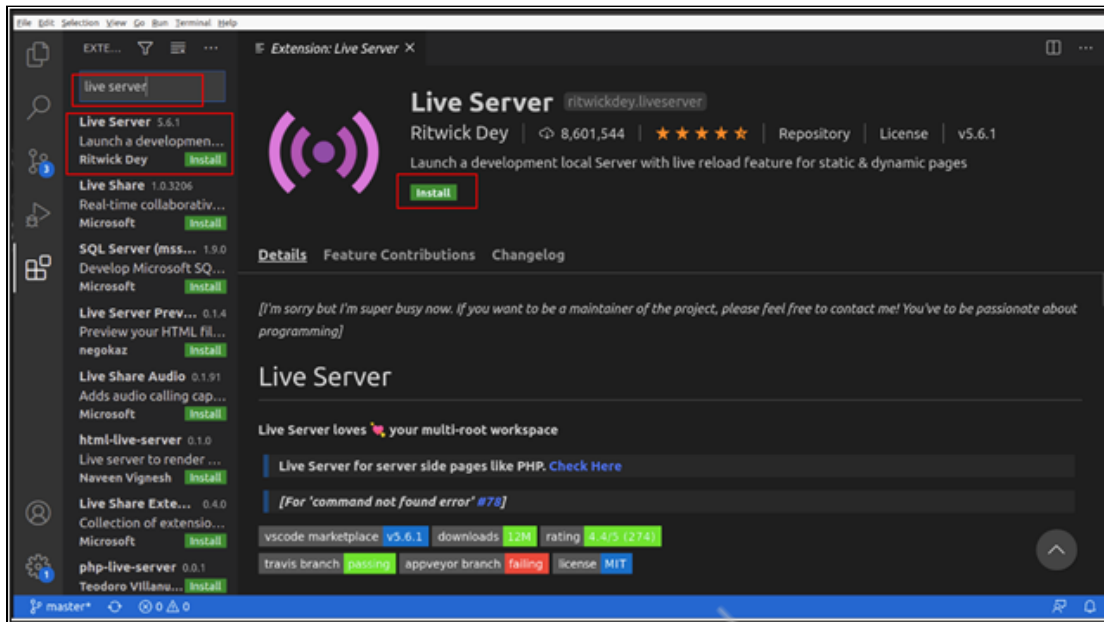


5. Descargar y guardar el código en la carpeta local. Luego, desde la opción Open

Folder - abrir carpeta, en la pestaña File - archivo, puedes elegir la carpeta para editar los archivos.

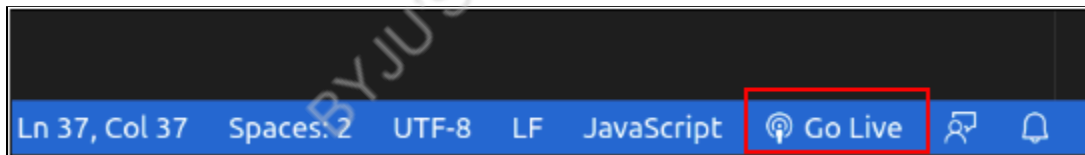


6. Ejecutar el código; para ver el output necesitamos un servidor local. VSC viene con una extensión para instalar Live Server. Instala Live Server con los siguientes pasos:
- Haz clic en View - ver, y luego selecciona 'Extensions' - extensiones, o presiona "Ctrl + Shift + x".
 - Busca "Live Server".
 - Haz clic en Install - instalar.



7. Iniciar Live Server.

- Abre la carpeta del proyecto en VSC.
- Haz clic en **Go Live** en la barra de estado para encender y apagar Live Server. O haz clic derecho en el archivo index.html y selecciona **Open with Live Server** - abrir con Live Server.



8. Utilizar la plantilla descargada en la carpeta códigos_del_juego. Escribe el código para crear una calculadora en VSC y ejecuta el código usando Live Server.

9. Declarar una variable `box` - caja, y crear el objeto utilizando `createSprite()` - crear sprite, en la función `setup()` - configurar.

```
var box;

function setup() {
  createCanvas(400,400);
  box = createSprite(200,200,30,30);
}

function draw()
{
  background(30);

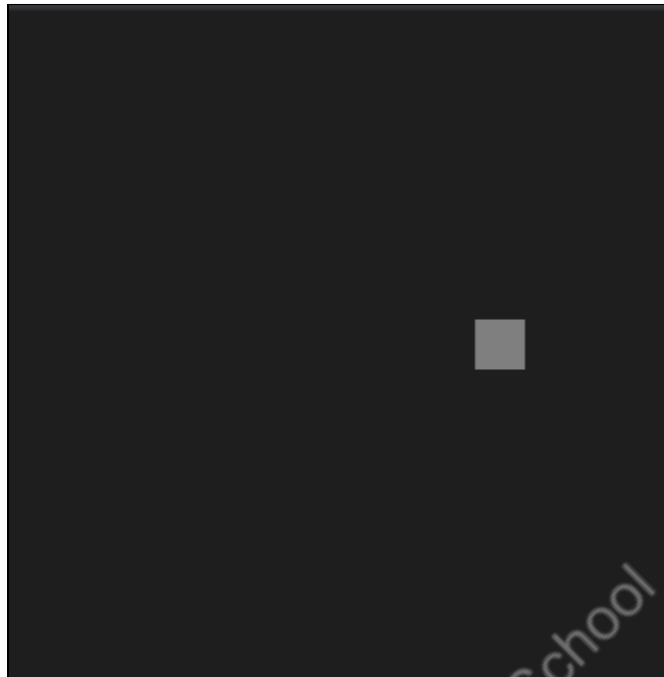
  drawSprites();
}
```

10. Mover el sprite en la dirección correcta al detectar la tecla presionada usando la función `keyIsdown()` y cambiando la posición del sprite de la caja..

```
function draw()
{
  background(30);

  if (keyIsDown(RIGHT_ARROW))
  {
    box.position.x = box.position.x +5;
  }

  drawSprites();
}
```



11. Mover el sprite en la dirección izquierda al detectar la tecla **LEFT_ARROW** - *Flecha Izquierda*, presionada y disminuye la posición x.

```
function draw()
{
  background(30);

  if (keyIsDown(RIGHT_ARROW))
  {
    box.position.x = box.position.x +5;
  }

  if (keyIsDown(LEFT_ARROW))
  {
    box.position.x= box.position.x -5;
  }

  drawSprites();
}
```


12. Mover el objeto en las cuatro direcciones utilizando las teclas de flecha.

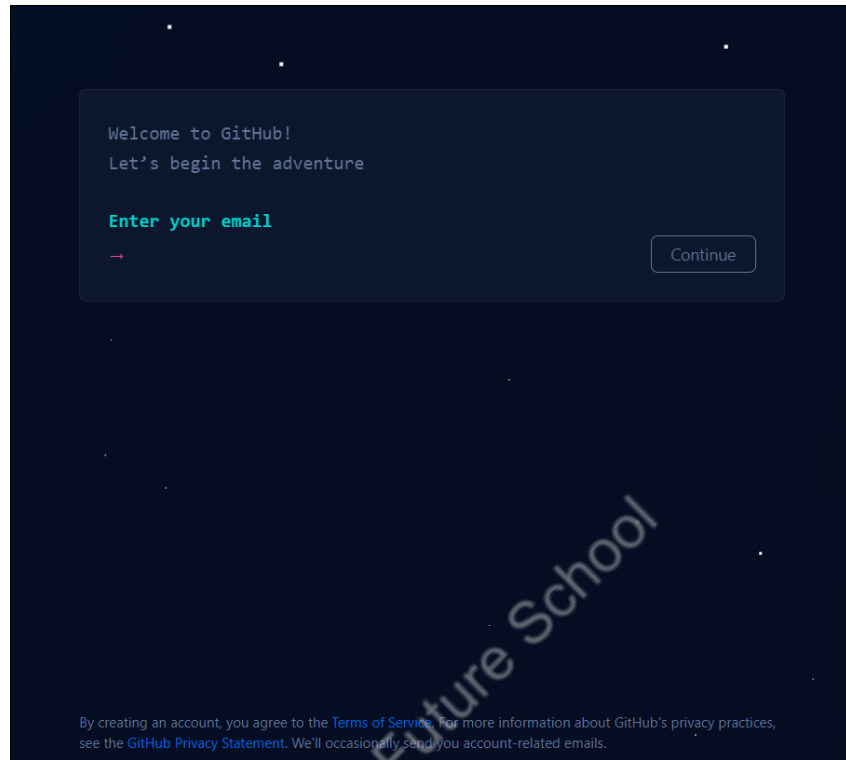
```
if (keyIsDown(UP_ARROW))  
{  
    box.position.y = box.position.y - 5;  
}  
  
if (keyIsDown(DOWN_ARROW))  
{  
    box.position.y = box.position.y + 5;  
}
```

13. Ejecutar el código.

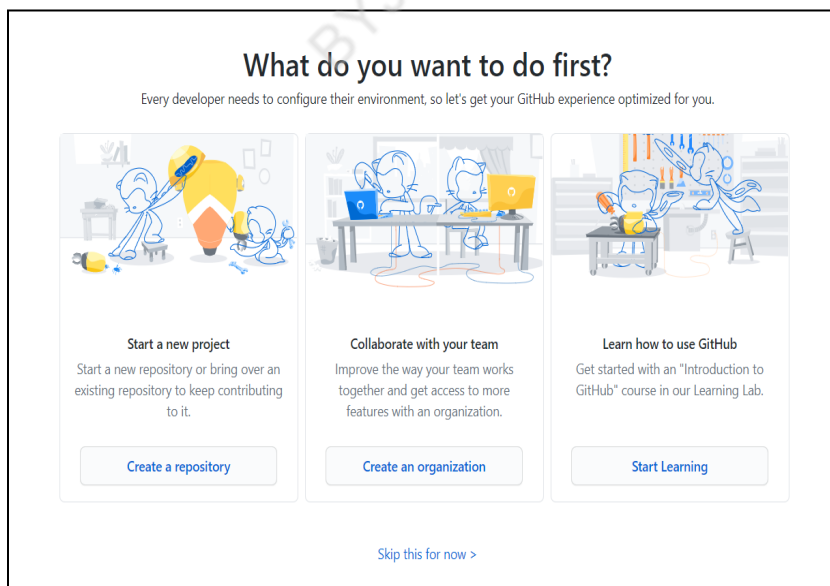


14. Subir el código a Github.

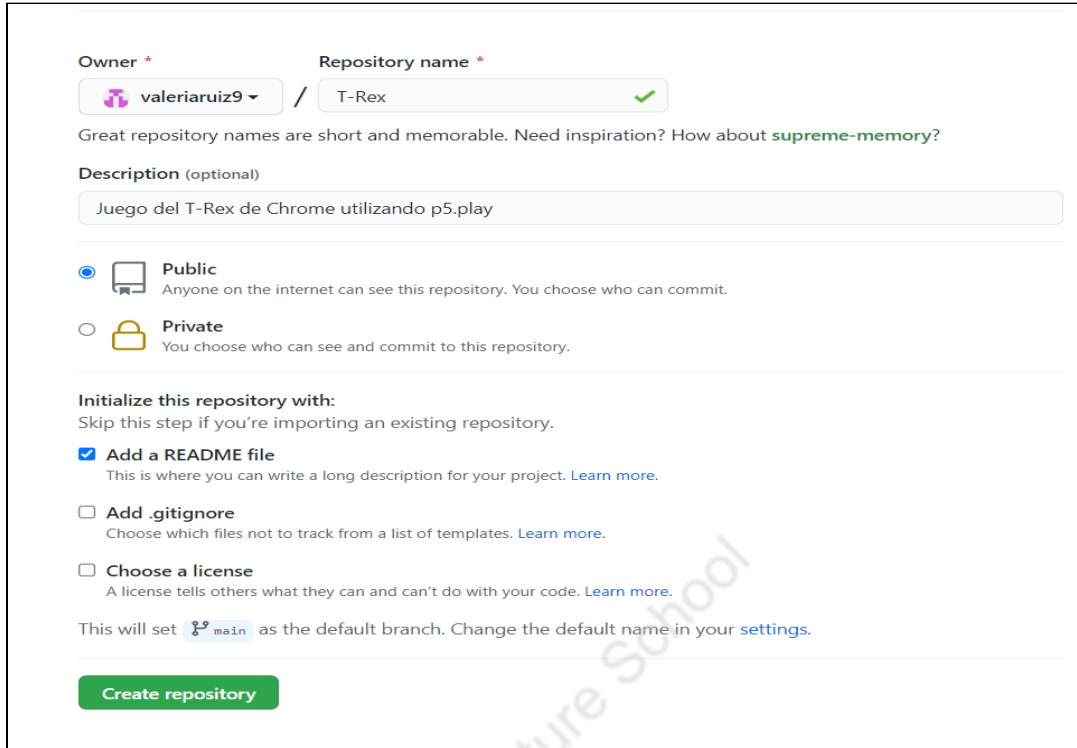
- Crear tu cuenta con tu ID de correo electrónico en el sitio web de GitHub.




- Haz clic en Start a new project - *iniciar un nuevo proyecto*.




- Crear un nuevo repositorio.




Owner *  valeriaruiz9 / Repository name * T-Rex ✓

Great repository names are short and memorable. Need inspiration? How about [supreme-memory](#)?

Description (optional)
Juego del T-Rex de Chrome utilizando p5.play

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

☒ **Add a README file**
This is where you can write a long description for your project. [Learn more.](#)

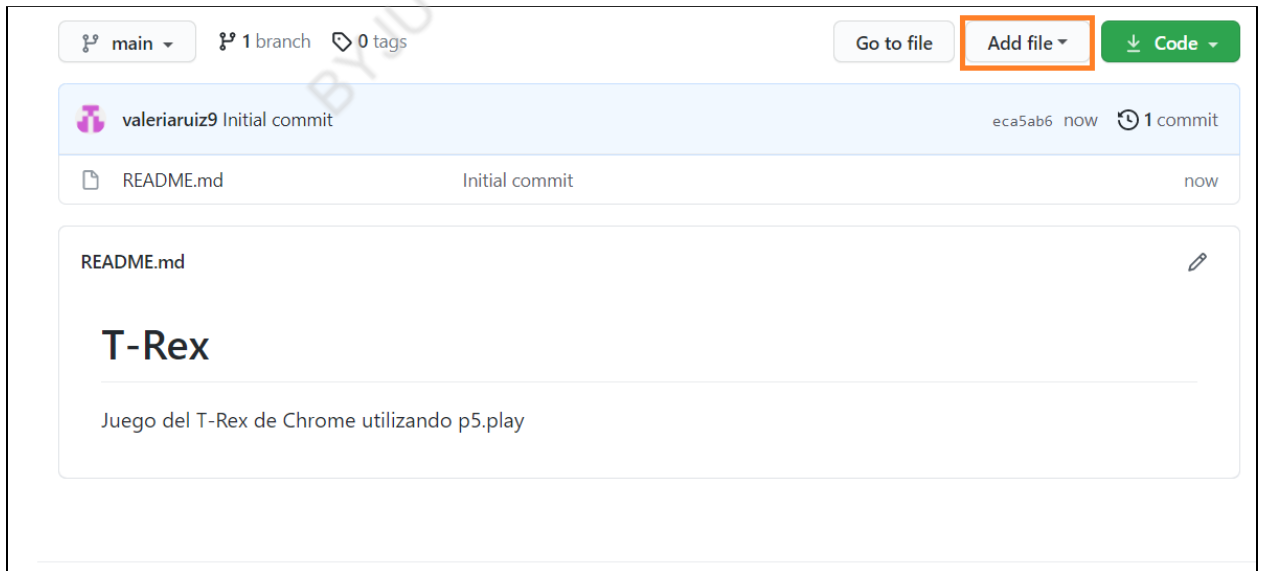
☐ **Add .gitignore**
Choose which files not to track from a list of templates. [Learn more.](#)




☐ **Choose a license**
A license tells others what they can and can't do with your code. [Learn more.](#)


This will set  **main** as the default branch. Change the default name in your [settings](#).


[Create repository](#)


- Subir los archivos del proyecto en Github.



 main  1 branch  0 tags [Go to file](#) [Add file](#) [Code](#)

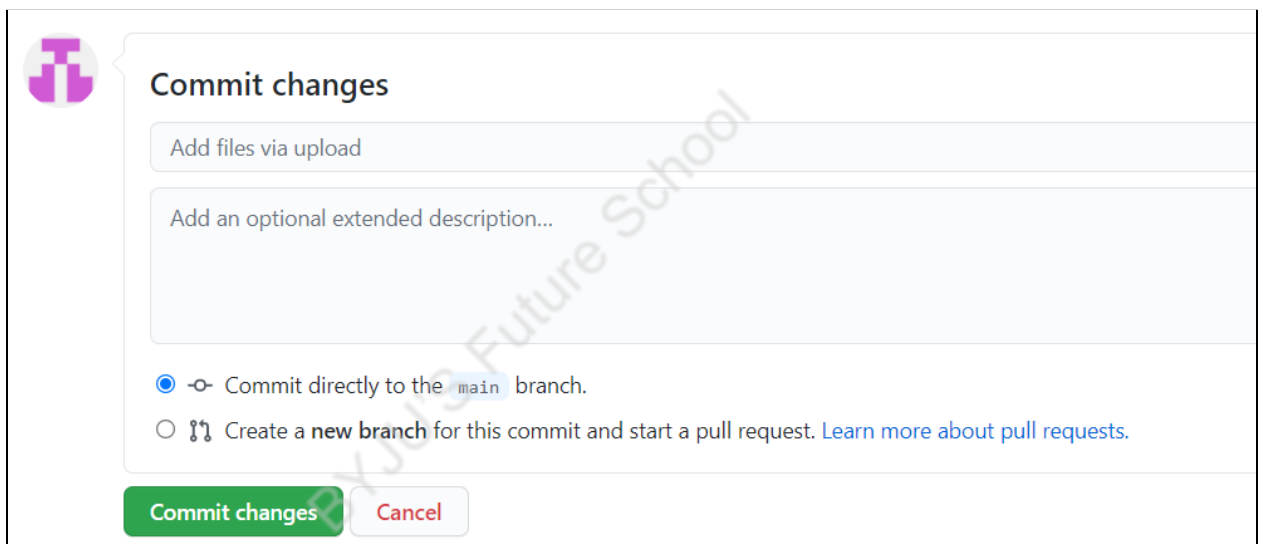
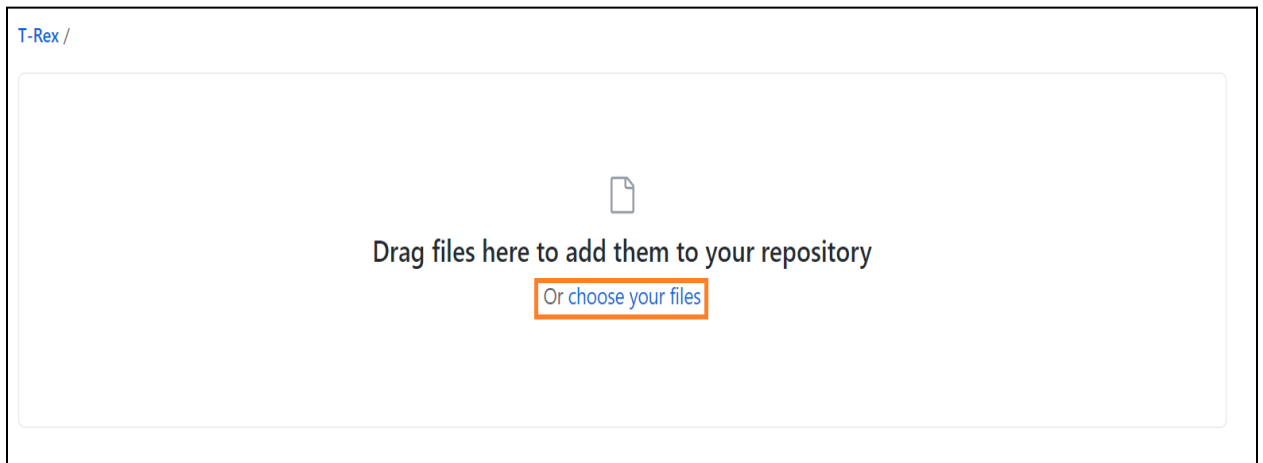
 valeriaruiz9 Initial commit eca5ab6 now 1 commit

File	Commit	Time
 README.md	Initial commit	now

README.md 

T-Rex

Juego del T-Rex de Chrome utilizando p5.play



- Subimos el código en GitHub.

¿Qué sigue?

En la siguiente clase aprenderás sobre los conceptos de cómo crear Infinite Runner Games - Juegos de corredores infinitos. Desarrollarás un programa en VSC.

Amplía tu conocimiento:

1. GitHub es una herramienta muy poderosa para desarrolladores; aprende más al respecto:

<https://kinsta.com/knowledgebase/what-is-github/>