

# **Prepara tu entorno de desarrollo**

Autor: Génesis Natalya Ushca Pizarro

Fecha: 03/10/2025




## Índice

1. Crea una cuenta en Github.....	3
2. Instala Visual Code Studio.....	4
3. Instala Git.....	4
4. Crea tu primer repositorio hello-world.....	5
5. Crea una rama. Introduce cambios. Haz una solicitud de incorporación de cambios. Combina los cambios. Elimina la rama.....	6
6. Clona el proyecto creado en Github en tu máquina mediante Visual Studio Code.....	10
7. Configura tu editor Visual Studio Code.....	11
8. Busca sobre las extensiones de VSC. instala las que consideres más interesantes para optimizar tu trabajo como desarrollador. Muestra la captura de pantalla de las extensiones instaladas y comenta su funcionalidad.....	11
9. Practica con la abreviaciones Emmet.....	14

# 1. Crea una cuenta en Github

Entramos en la página oficial de GitHub desde el siguiente [enlace](#). Si ya tenemos una cuenta iniciamos sesión pero si no es así deberemos registrarnos.

## Sign up for GitHub

 Continue with Google

or

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\*

Spain

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

Create account >

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.

Rellenamos los campos obligatorios siguiendo las reglas para la contraseña y el username. Una vez hecho, creamos la cuenta.

En mi caso, ya tengo una creada así que este paso no fue necesario.

## 2. Instala Visual Code Studio

Para descargar la última versión entramos en su página oficial desde el siguiente [enlace](#) y elegimos la opción de descarga que se adapte a nuestro sistema operativo. En mi caso, Windows 11.

También, podemos elegir entre Linux o Mac y descargarnos Visual Code Studio en diferentes archivos.

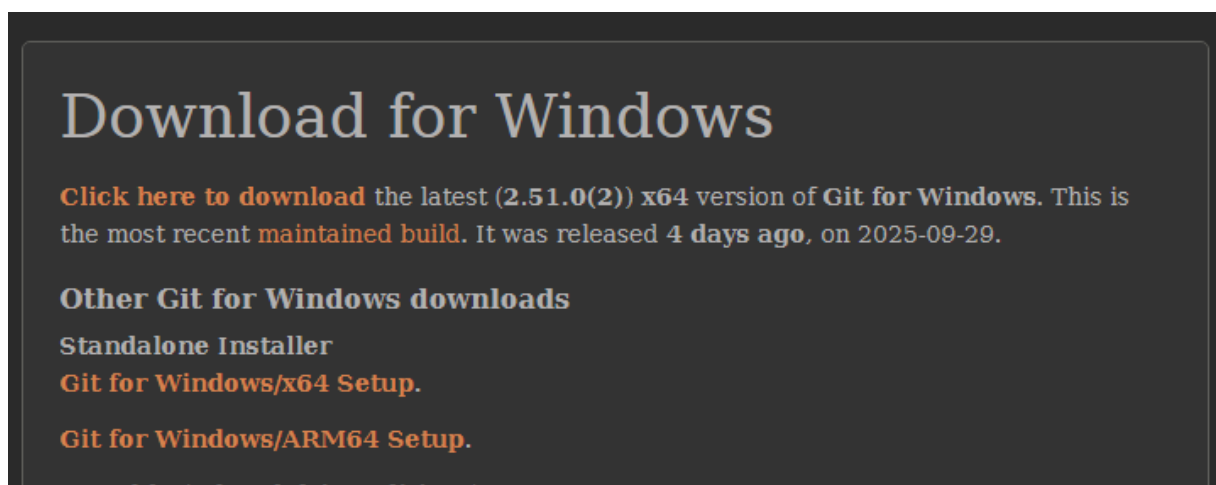


Por defecto, se descarga la última versión disponible de Visual Code Studio.

Una vez descargado el archivo lo ejecutamos, seguimos los pasos de instalación y ya podemos usarlo.

## 3. Instala Git

Para instalar la versión de Git para escritorio entramos en el siguiente [enlace](#).

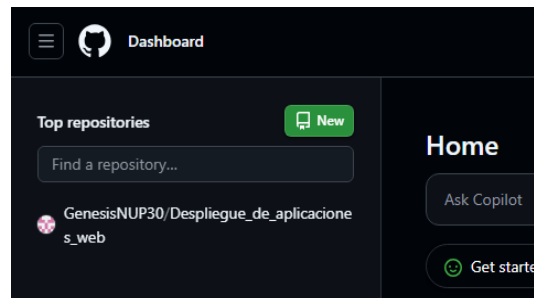


Una vez descargado el archivo lo ejecutamos y seguimos los pasos para instalarlo.

## 4. Crea tu primer repositorio hello-world

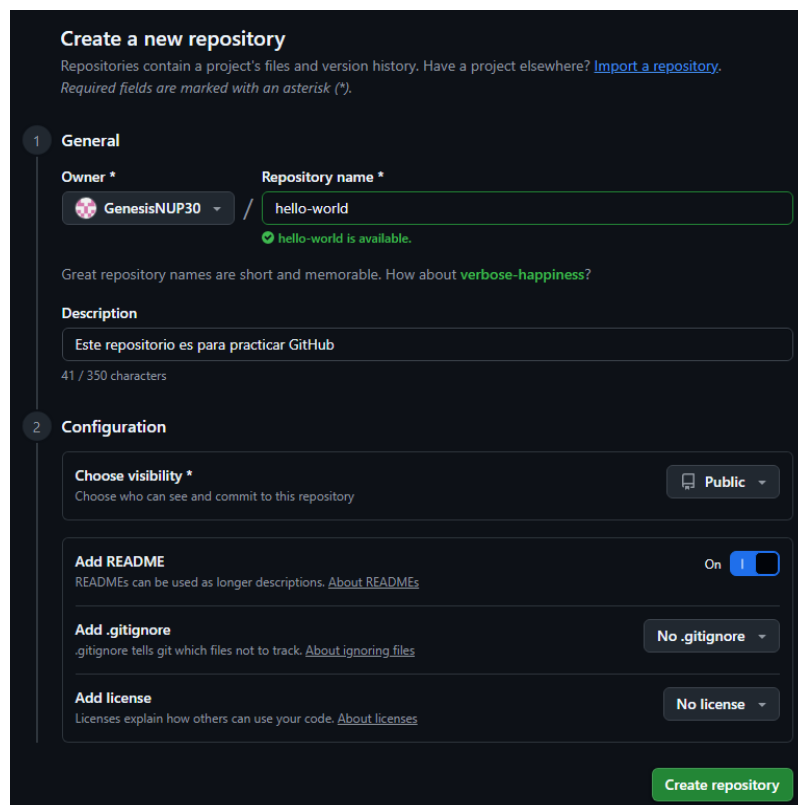
Podemos usar GitHub instando su aplicación para escritorio o desde la página web. Yo lo voy a hacer desde la página web.

Para crear un repositorio hacemos clic en New.



A continuación rellenamos los campos obligatorios:

- Nombre del repositorio: escribimos hello-world.
- Descripción: ponemos una pequeña descripción

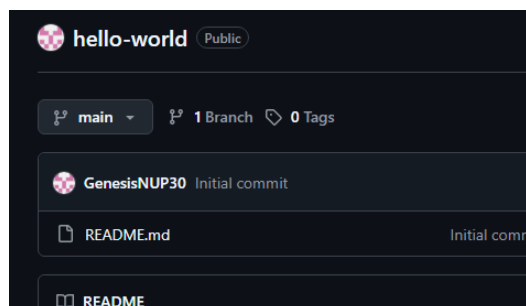


Ponemos la visibilidad como **Public**, añadimos el archivo **README** y creamos el repositorio en **Create repository**

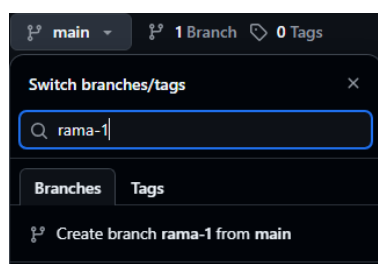
## 5. Crea una rama. Introduce cambios. Haz una solicitud de incorporación de cambios. Combina los cambios. Elimina la rama.

Para crear la rama seguimos estos pasos:

1. Encima de la lista de archivos, hacemos clic en el menú desplegable **main**.



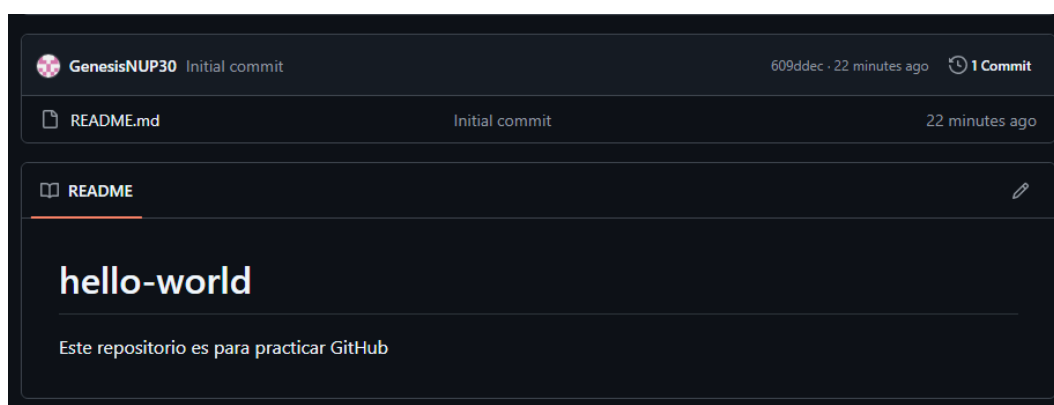
2. Escribimos el nombre de la rama.



3. Creamos la rama pulsamos en **Create branch rama-1 from main**.

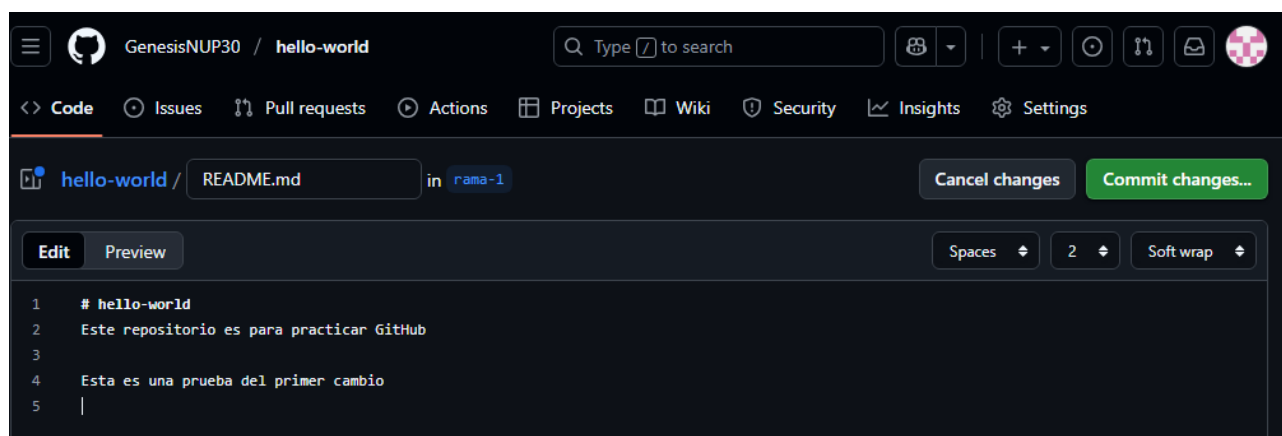
Ahora vamos a hacer un cambio.

Para eso desde la rama que acabamos de crear **rama-1** editamos el archivo **README** con el botón del lápiz.





Escribimos algo en nuestro archivo.



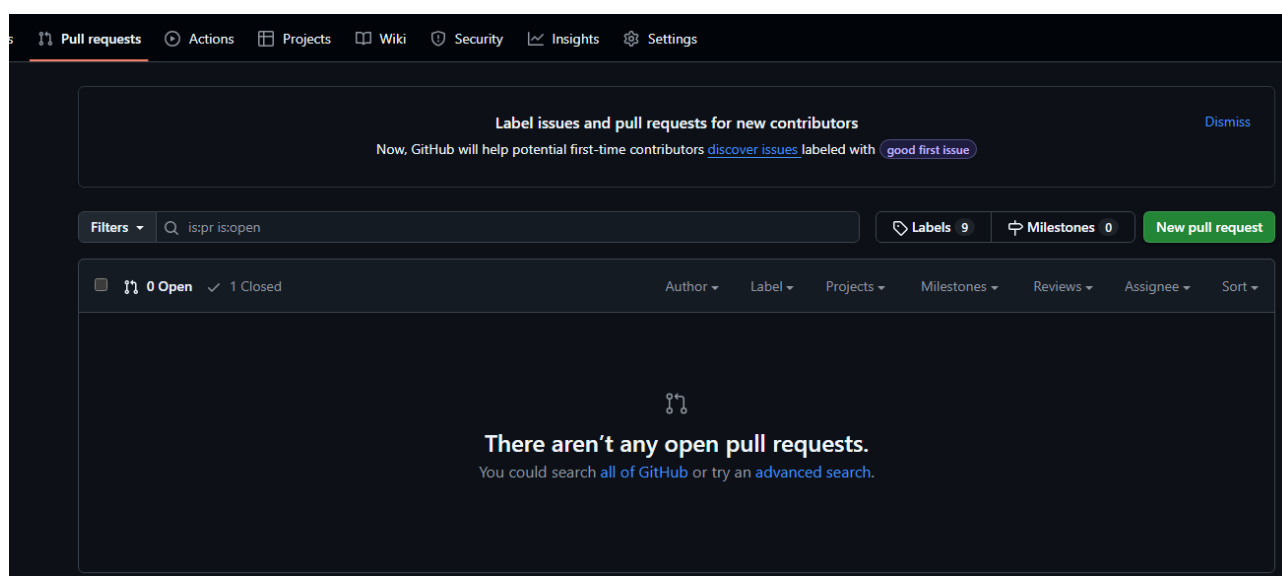
Para guardar los cambios pulsamos en el botón **Commit changes**.

Estos cambios solo se realizarán en el archivo README de la rama rama-1, por lo que ahora contiene contenido diferente al de main.

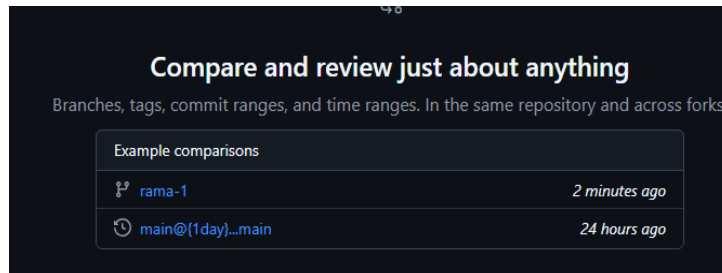
Ahora vamos a hacer una solicitud de incorporación de cambios.

Para eso hacemos lo siguiente:

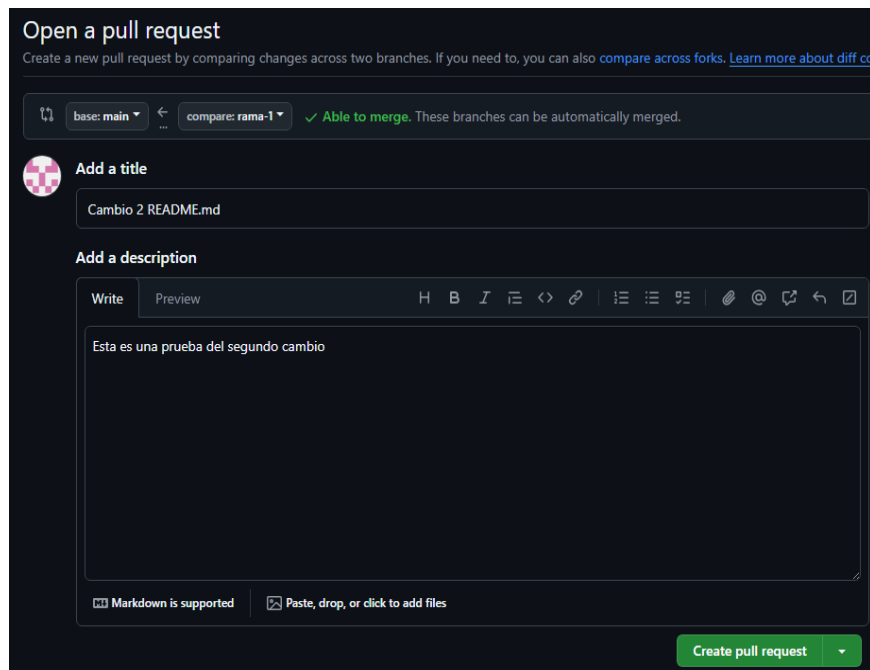
1. Volvemos a la página principal de **rama-1**.
2. Pulsamos en la opción **Pull request**
3. Para crear una solicitud, hacemos clic en el botón **New pull request**.



4. En el cuadro Comparaciones de ejemplo, seleccionamos la rama que hemos creado, rama-1, para compararla con main (la original).

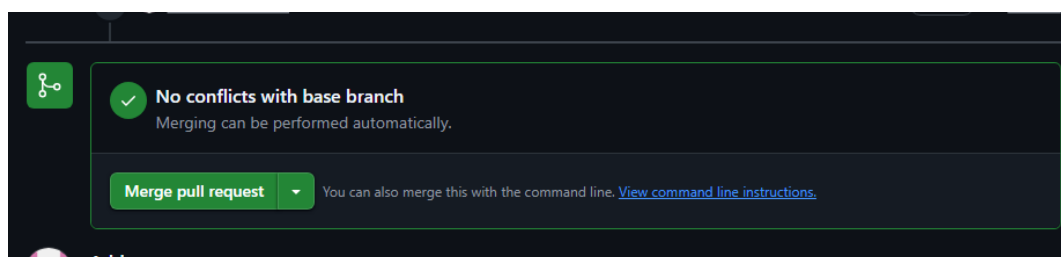


5. Volvemos a pulsar el botón **Create pull request**.  
Rellenamos el nombre y la descripción.



Pulsamos el Create pull request.

Para combinar los cambios en la misma pestaña pulsamos en **Merge pull request**.



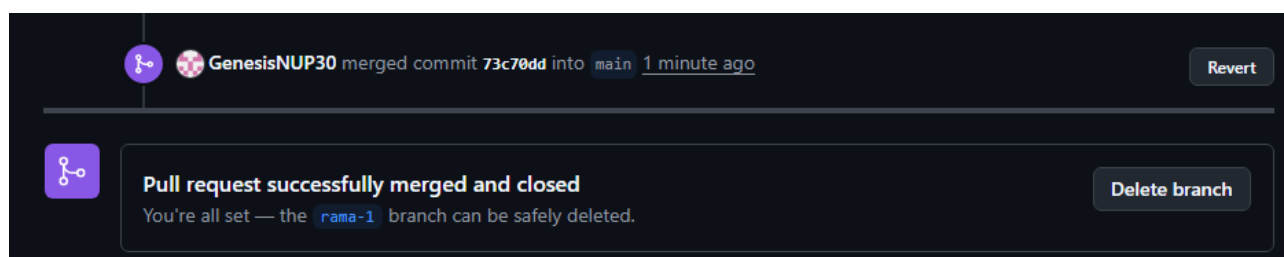




En la siguiente pestaña confirmamos los cambios con **Confirm merge**.

The screenshot shows a GitHub merge confirmation dialog. On the left is a green icon with a white branching diagram. The main area has a dark background. At the top, 'Commit message' is followed by a text input field containing 'Merge pull request #2 from GenesisNUP30/rama-1'. Below this, 'Extended description' is followed by a text area containing 'Cambio 2 README.md'. At the bottom, it says 'This commit will be authored by gushpiz998@g.educaand.es.' and has two buttons: 'Confirm merge' (green) and 'Cancel' (grey).

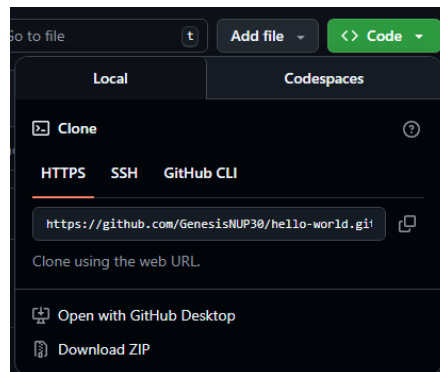
Por último, como ya hemos combinados los cambios, podemos eliminar la rama haciendo clic en **Eliminar rama**.



## 6. Clona el proyecto creado en Github en tu máquina mediante Visual Studio Code

Lo haremos desde el terminal de Visual Code Studio.

Antes vamos a copiar la URL de nuestro repositorio. Para eso hacemos clic en **Code** > **HTTPS**.



Abrimos la terminal con Ctrl + Ñ o bien la barra superior > Terminal.

Escribimos el comando **git clone** <https://github.com/GenesisNUP30/hello-world.git>

```
PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL  PUERTOS

PS C:\Users\Génesis> git clone https://github.com/GenesisNUP30/hello-world.git
Cloning into 'hello-world'...
remote: Enumerating objects: 7, done.
remote: Counting objects: 100% (7/7), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 7 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (7/7), done.
Resolving deltas: 100% (1/1), done.
PS C:\Users\Génesis> 
```

Después hacemos **cd hello-world** para movernos a nuestro repositorio clonado y para abrirlo en Visual Code Studio hacemos **code ..**

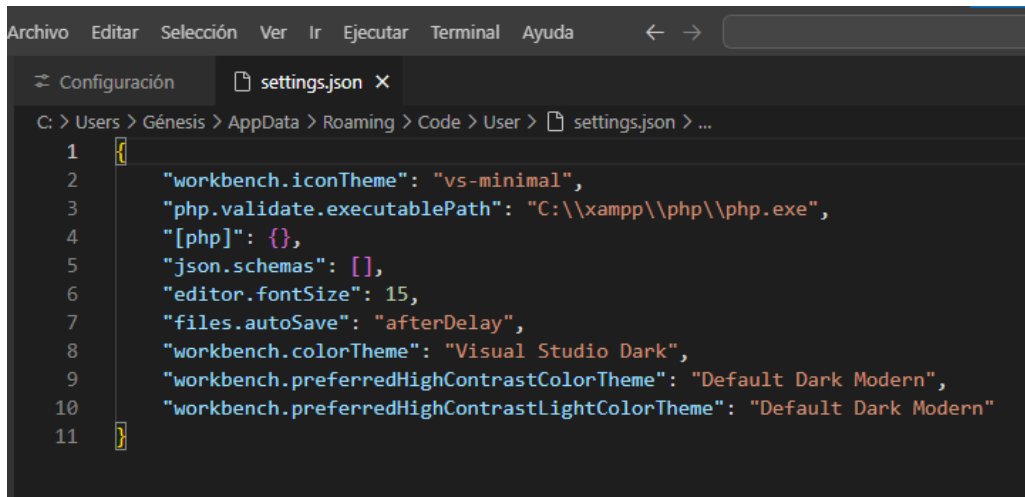
```
PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL  PUERTOS

PS C:\Users\Génesis\hello-world> code .
PS C:\Users\Génesis\hello-world> 
```

Y se nos abre una nueva ventana de Visual Code con nuestro repositorio.

## 7. Configura tu editor Visual Studio Code

Desde Archivo > Preferencias he ido modificando diferentes opciones según mi preferencia y este es el contenido de mi archivo settings.json.



```
Archivo  Editar  Selección  Ver  Ir  Ejecutar  Terminal  Ayuda  <  >
Configuración  settings.json x
C: > Users > Génesis > AppData > Roaming > Code > User > settings.json > ...
1  {
2      "workbench.iconTheme": "vs-minimal",
3      "php.validate.executablePath": "C:\\xampp\\php\\php.exe",
4      "[php]": {},
5      "json.schemas": [],
6      "editor.fontSize": 15,
7      "files.autoSave": "afterDelay",
8      "workbench.colorTheme": "Visual Studio Dark",
9      "workbench.preferredHighContrastColorTheme": "Default Dark Modern",
10     "workbench.preferredHighContrastLightColorTheme": "Default Dark Modern"
11 }
```

## 8. Busca sobre las extensiones de VSC. instala las que consideres más interesantes para optimizar tu trabajo como desarrollador. Muestra la captura de pantalla de las extensiones instaladas y comenta su funcionalidad.

### 1. Error Lens

Esta extensión resalta los errores y advertencias directamente en el código, así no tengo que estar revisando la barra de problemas todo el tiempo. Es muy útil porque puedo ver los errores mientras escribo.

### 2. HTML CSS Support

Me ayuda con el autocompletado de CSS dentro de archivos HTML. Por ejemplo, si tengo una clase en mi CSS, me la sugiere cuando escribo en el HTML.



### 3. HTML Preview

Con esta extensión puedo previsualizar mis archivos HTML sin abrir un navegador externo. Así puedo ver los cambios que hago en tiempo real.

### 4. PHP Debug

Permite depurar código PHP usando Xdebug. Puedo poner breakpoints y ver paso a paso qué hace mi código, lo que facilita encontrar errores.

### 5. PHP Intelephense

Esta extensión me da autocompletado y sugerencias en PHP. También detecta errores de sintaxis, así que es más fácil programar sin cometer fallos.

### 6. Pylance

Es un motor de inteligencia para Python. Me da autocompletado, ayuda con los tipos de datos y detecta errores antes de ejecutar el programa.

### 7. Python

Es la extensión oficial para Python. Sirve para usar IntelliSense, depurar, trabajar con entornos virtuales y ejecutar código Python desde VS Code.

### 8. Python Debugger

Me permite depurar programas en Python usando breakpoints y revisar variables mientras corre el código.

### 9. Python Environments

Ayuda a cambiar entre distintos entornos de Python. Esto es útil porque cada proyecto puede necesitar diferentes librerías.

### 10. Spanish Language Pack

Traduce Visual Studio Code al español, lo que me facilita entender todas las opciones y configuraciones del editor.

### 11. Auto Close Tag















Cierra automáticamente las etiquetas HTML o XML que abro. Así no me olvido de cerrar una etiqueta y evito errores.

### 12. Auto Complete Tag

Sugiere y completa automáticamente etiquetas en HTML o XML. Trabaja junto con Auto Close Tag y me hace más rápido escribir código.

### 13. Auto Rename Tag

Si cambio el nombre de una etiqueta de apertura, cambia automáticamente la de cierre. Esto evita errores cuando modifico el código.

	<b>Auto Close Tag</b>	6ms
	Automatically add HTML/XML close tag, same as Visual Studio IDE or Sublime Text	
	Jun Han	
	<b>Auto Complete Tag</b>	
	Extension Packs to add close tag and rename paired tag automatically for HTML/XML	
	Jun Han	
	<b>Auto Rename Tag</b>	83ms
	Auto rename paired HTML/XML tag	
	Jun Han	
	<b>Error Lens</b>	30ms
	Improve highlighting of errors, warnings and other language diagnostics.	
	Alexander	
	<b>HTML CSS Support</b>	10ms
	CSS Intellisense for HTML	
	ecmel	
	<b>HTML Preview</b>	
	Provides ability to preview HTML documents.	
	George Oliveira	
	<b>PHP Debug</b>	
	Debug support for PHP with Xdebug	
	Xdebug	
	<b>PHP Intelephense</b>	
	PHP code intelligence for Visual Studio Code	
	Ben Mewburn	
	<b>Pylance</b>	
	A performant, feature-rich language server for Python in VS Code	
	Microsoft	
	<b>Python</b>	
	Python language support with extension access points for IntelliSense (Pylance), Debugg...	
	Microsoft	
	<b>Python Debugger</b>	
	Python Debugger extension using debugpy.	
	Microsoft	
	<b>Python Environments</b>	
	Provides a unified python environment experience	
	Microsoft	
	<b>Spanish Language Pack for Visual Studio Code</b>	
	Español	
	Microsoft	
	<b>Supermaven</b>	228ms
	The fastest copilot.	
	Supermaven	

## 9. Practica con la abreviaciones Emmet

Esta es mi pequeña pagina con html usando abreviaciones.

```
Archivo  Editar  Selección  Ver  Ir  Ejecutar  Terminal  Ayuda  <  >  Buscar

emmet.html X
G: > Mi unidad > 2º DAW > DIW > emmet.html > html > body > table > tr > td

1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Document</title>
7  </head>
8  <body>
9      <div class="container">Esto es un div</div>
10     <div id="header" class="container">Esto es un div con id header y con clas
11     <ul>
12         <li>lista 1</li>
13         <li>lista 2</li>
14         <li>lista 3</li>
15         <li>lista 4</li>
16         <li>lista 5</li>
17     </ul>
18     <table border="1">
19     <tr>
20         <td>Celda 1</td>
21         <td>Celda 2</td>
22         <td>Celda 3</td>
23     </tr>
24     <tr>
25         <td>Celda 4</td>
26         <td>Celda 5</td>
27         <td>Celda 6</td>
28     </tr>
29     <tr>
30         <td>Celda 7</td>
31         <td>Celda 8</td>
32         <td>Celda 9</td>
33     </tr>
34 </table>
35 </body>
36 </html>
```

Preview emmet.html X

Esto es un div  
Esto es un div con id header y con clase container

- lista 1
- lista 2
- lista 3
- lista 4
- lista 5

Celda 1	Celda 2	Celda 3
Celda 4	Celda 5	Celda 6
Celda 7	Celda 8	Celda 9

