

Módulo Herramientas de Desarrollo

Integrar Visual Studio Code con git

Objetivo: Entender cómo integrar git con Visual Studio Code

Introducción

Visual Studio Code es uno de los más populares editores de código para desarrollo web. Ha ganado mucha popularidad debido a características que posee como el control integrado de código fuente lo cual permite a los desarrolladores manejar de manera eficiente y cómoda desde la interfaz de usuario de Visual Studio Code todo lo relacionado al control de código.

Con la misma, los usuarios pueden **inicializar** un nuevo repositorio, **agregar archivos** al control de versiones, **confirmar cambios**, **crear ramas**, **fusionar ramas**, ver el **historial de cambios**, y mucho más. Además, también permite trabajar con **repositorios remotos**, como **GitHub** o **GitLab**, lo que permite a los desarrolladores **colaborar** en proyectos de manera efectiva y eficiente. En resumen, el control de versiones que provee Visual Studio Code es una herramienta esencial para cualquier desarrollador que trabaja con Git.

Inicializar un repositorio Git

Siempre que se comienza un nuevo proyecto es necesario configurar el repositorio que todo el equipo utilizará para el control de versiones de nuestros archivos. Para iniciar un nuevo repositorio Git, haga clic en el icono de Git en la barra de estado inferior de Visual Studio Code (símbolo de división de calles) de ahora en más **panel de control de versiones** y seleccione abrir directorio que solicitará la selección de una carpeta, una vez realizado esto es necesario "Inicializar Repositorio". Esto creará un archivo .git en dicho directorio.





SOURCE CONTROL



In order to use git features, you can open a folder containing a git repository or clone from a URL.



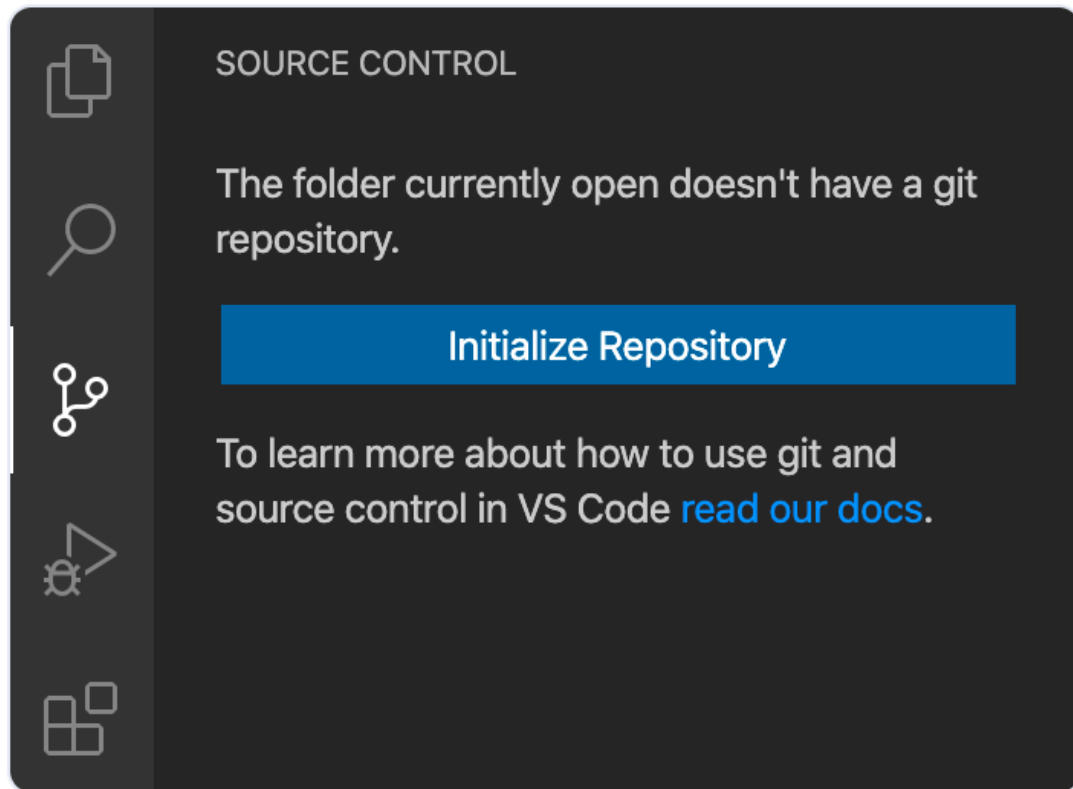
Open Folder

Clone Repository



To learn more about how to use git and source control in VS Code [read our docs](#).



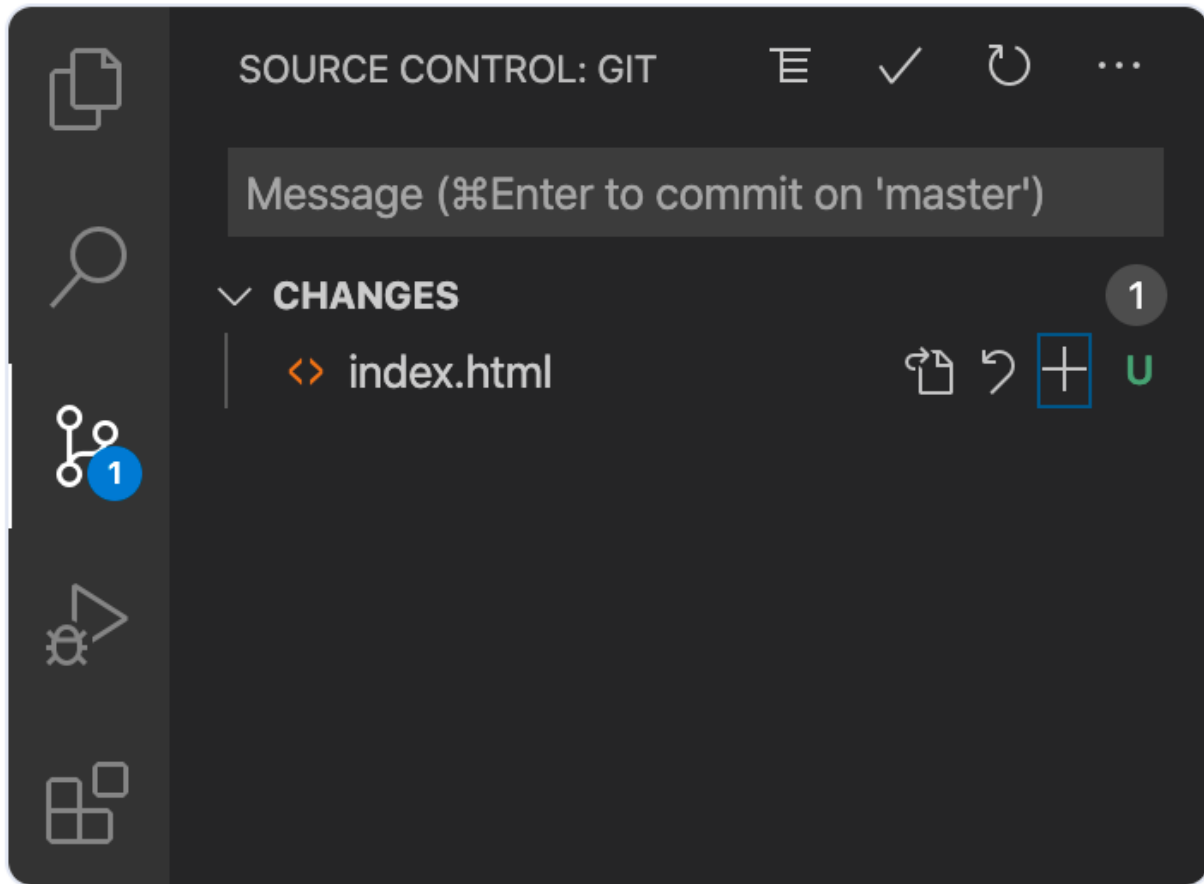


Nota: se puede verificar que luego de inicializar el repositorio se crea una carpeta **.git** lo cual se puede lograr por medio del terminal con el comando **ls -la**

Agregar archivos al control de versiones

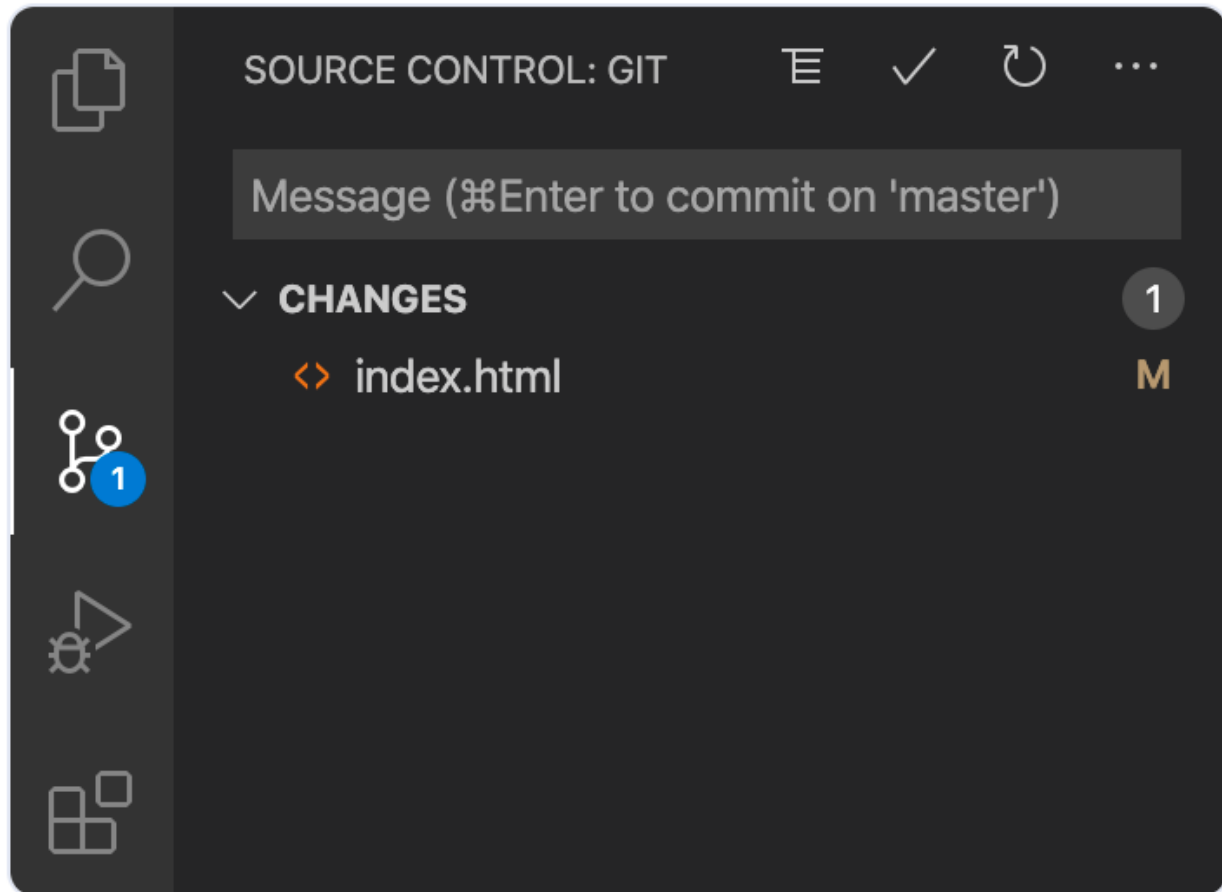
Es evidente que para desarrollar nuestras aplicaciones será necesario interactuar con distintos tipos de archivos los cuales eventualmente deberán ser integrados a nuestro repositorio es por ello que primero debemos saber como agregar archivos al control de versiones de Git.

Primero es necesario tener un archivo por ende se puede crear un nuevo archivo en la carpeta raíz de nuestro proyecto en Visual Studio Code. Luego, haga clic en el icono de control de versiones en el cual se verá un archivo que muestra una letra **U** que hace referencia a **Untracked Changes** lo cual hace referencia a que nuestro archivo es nuevo o se ha cambiado y que aún no se ha agregado al repositorio. Desde la parte derecha del archivo se encontrará una opción **(+)** la cual agrega el archivo a nuestro repositorio cambiando la letra del mismo a **A** (**Added**).



Confirmar cambios

Para confirmar los cambios es necesario agregar un comentario relativo al cambio que se está realizando. Luego se debe presionar el botón de confirmación ✓ (tilde). Una vez realizado esto notará que no hay más cambios pendientes.

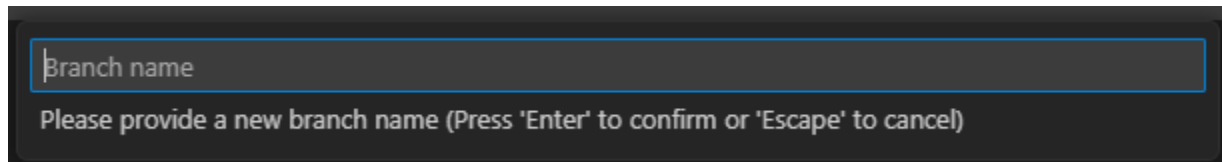
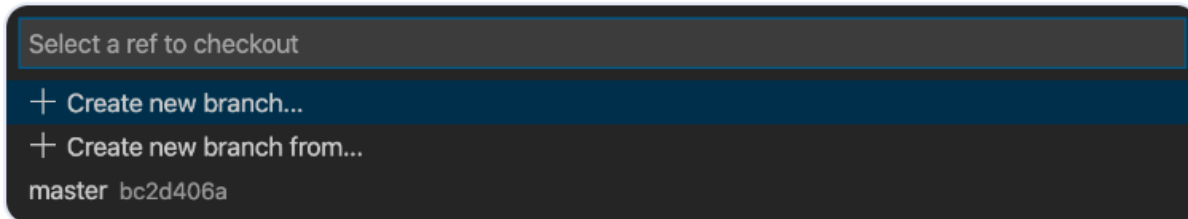


Crear ramas

En la parte inferior izquierda de VS Code existe la posibilidad de crear y cambiar de ramas, en donde se puede ver el icono utilizado en el control de versiones seguido del nombre de la rama actual.



Para crear una nueva rama, haga clic en el nombre de la rama, en el menú emergente seleccione “Crear rama” y luego ingrese el nombre deseado.



Trabajar con repositorios remotos

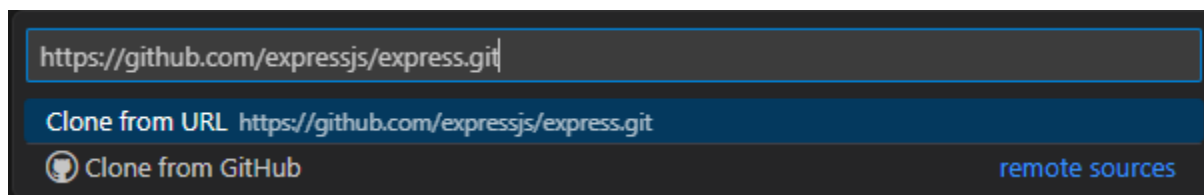
En general siempre se va a trabajar con repositorios remotos por lo cual es importante que VS Code nos de una forma fácil y rápida de realizar esta tarea y de esa manera evitar la línea de comandos.

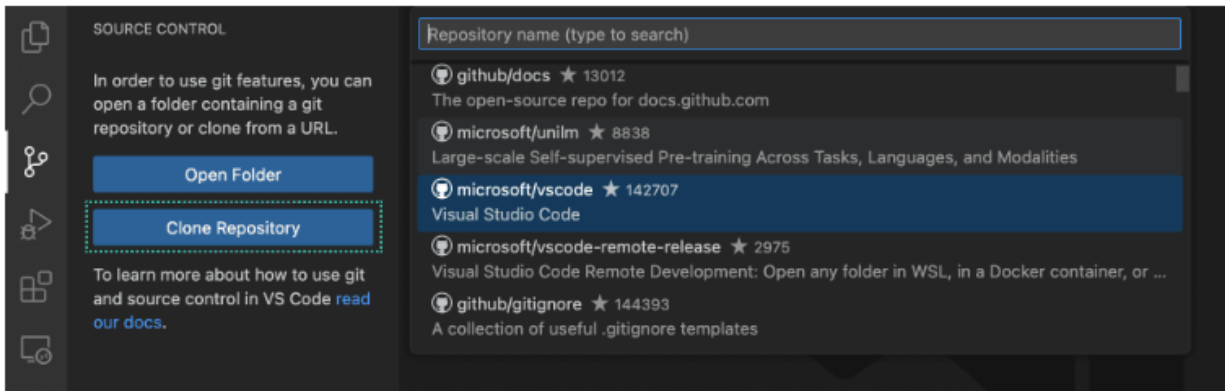
Se necesitará una carpeta donde se guardar los repositorios remotos es necesario disponer de la URL del repositorio remoto. Nota: dicha url debe tener el `.git` al final y puede ser obtenida desde el repositorio remoto con el que se trabaje.

En el ejemplo se muestra un repositorio de GitHub pero el procedimiento es el mismo para cualquier otro proveedor.

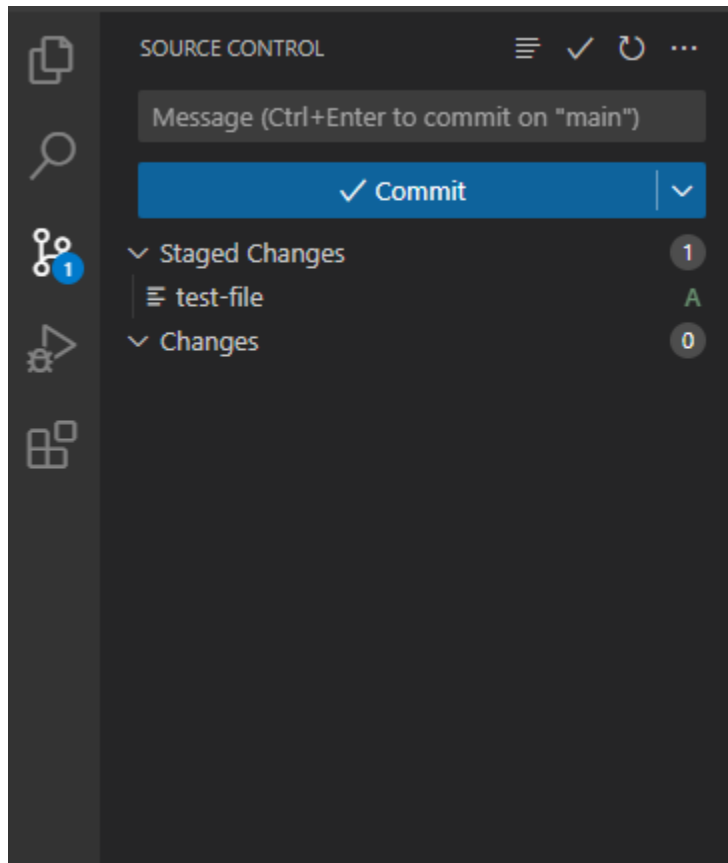
Una vez que se tiene la url desde Visual Studio Code se puede clonar el repositorio buscando el comando **git clone** desde la opción **View => Command Palette**. Otra opción es desde el botón **Clone Repository** dentro de la ventana de control de versiones.

Se ingresa la url obtenida anteriormente y se selecciona una carpeta donde se va a clonar el repositorio remoto, luego de un instante (dependiendo del tamaño del repositorio) VS Code sugerirá si se quiere abrir el repositorio clonado o agregarlo a un espacio de trabajo. En caso de abrirlo nos mostrará el árbol de archivos y ya se podrá trabajar creando ramas, como así también sincronizar las ramas con el repositorio remoto.

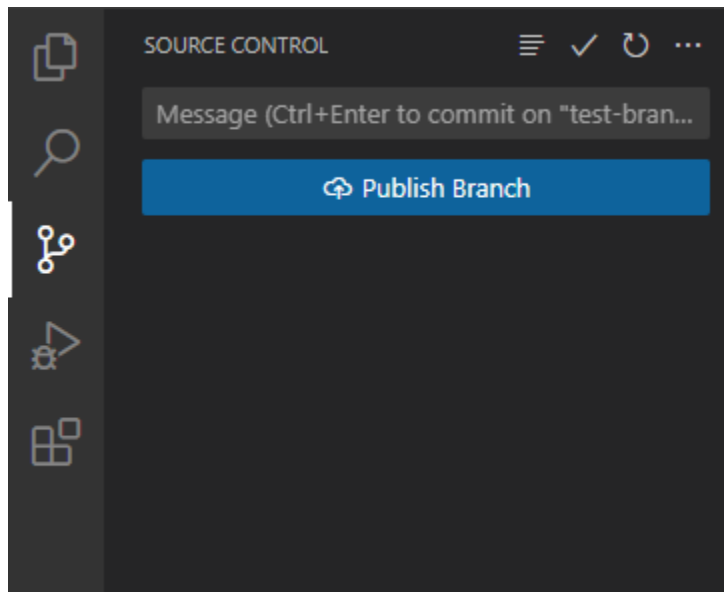




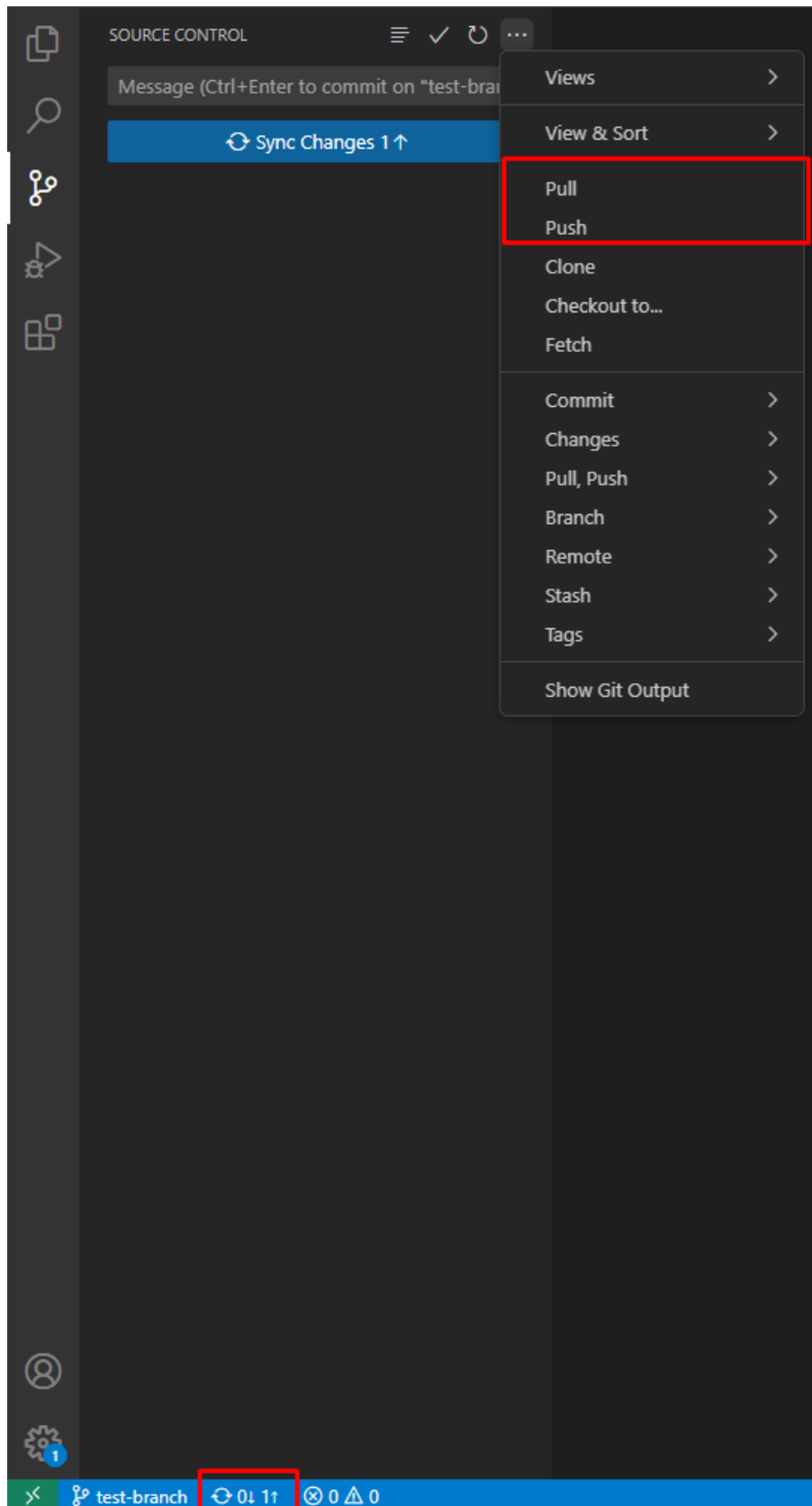
Se puede agregar un archivo y luego incluirlo lo que va a generar que el archivo pase a la sección de Staged Changes, agregando un comentario y confirmando los cambios



Luego se puede publicar la rama en el repositorio remoto.



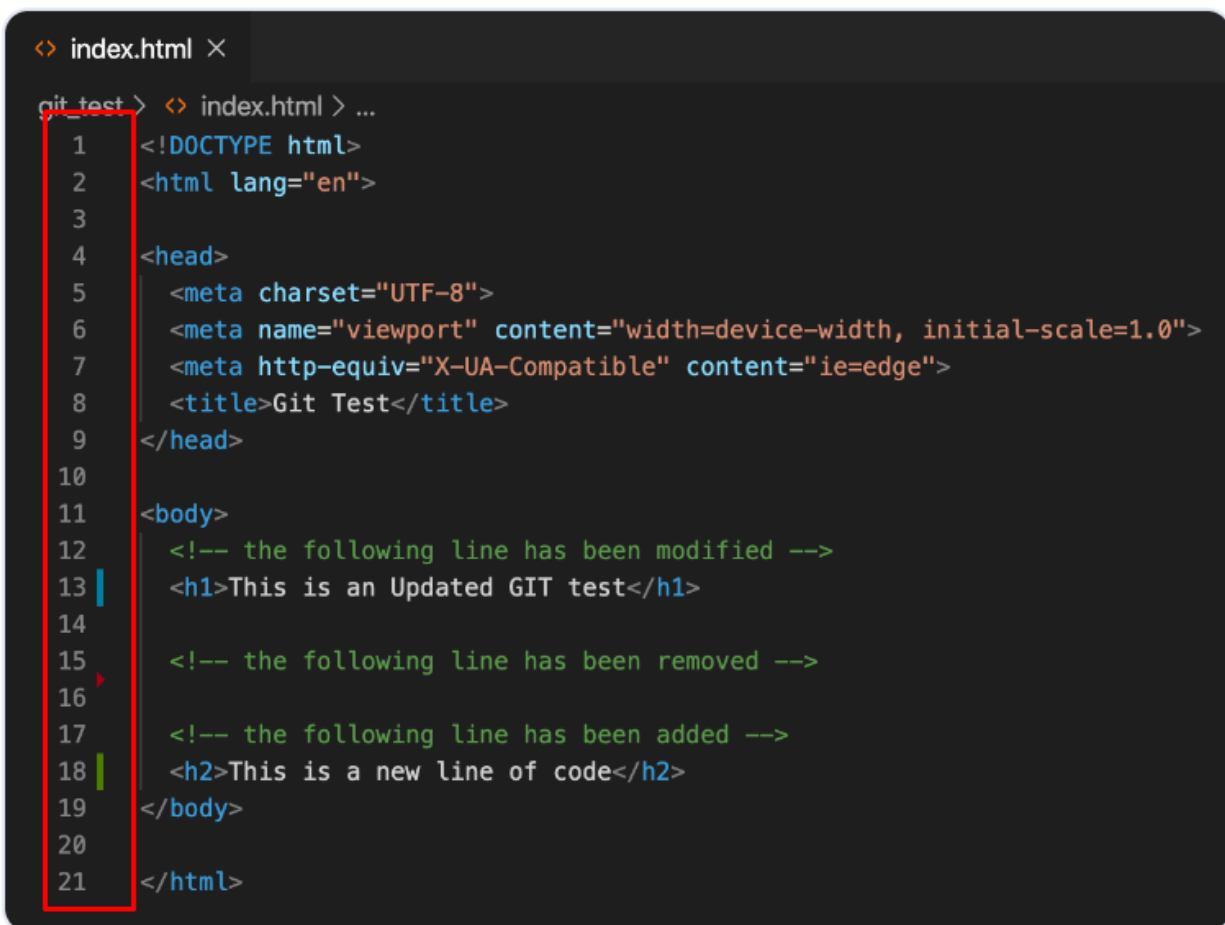
En caso de tener una rama creada y agregar más archivos se puede **enviar (Push)** y **obtener (Pull)** cambios desde el repositorio remoto. O también es posible hacerlo desde la barra inferior derecha lo cual consiste en hacer una sincronización con el repositorio remoto haciendo un Pull y Push.



Interpretando indicadores de cambios

Visual Studio Code provee una herramienta visual para interpretar los distintos cambios sobre los archivos de manera sencilla. Esto lo logra por medio de indicadores están ubicados a la derecha de donde se muestran las líneas de código.

A continuación se muestra una imagen en donde se han realizado varios cambios y se puede apreciar los mismos marcados con distintos indicadores y colores las acciones que se han realizado en el archivo.



```

<> index.html X
git test > <> index.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3
4  <head>
5    <meta charset="UTF-8">
6    <meta name="viewport" content="width=device-width, initial-scale=1.0">
7    <meta http-equiv="X-UA-Compatible" content="ie=edge">
8    <title>Git Test</title>
9  </head>
10
11 <body>
12   <!-- the following line has been modified -->
13   <h1>This is an Updated GIT test</h1>
14
15   <!-- the following line has been removed -->
16
17   <!-- the following line has been added -->
18   <h2>This is a new line of code</h2>
19 </body>
20
21 </html>
  
```

Por ejemplo en el primer indicador marcado como una barra azul, lo cual indica que la acción que se ha realizado fue **cambiar** el contenido dentro del tag **h1** por ende se está indicando que se ha modificado dicha línea de código.

Por otro lado se ve la flecha roja apuntando a la derecha la cual nos dice que se ha borrado una línea de código, esto puede representar que se ha borrado una o más líneas.

Por último, el indicador de barra verde señala que se ha agregado una nueva línea de código.

De esta manera se puede identificar sencillamente los cambios que se han realizado en un archivo solo viendo estos indicadores, en caso de necesitar más detalles sobre los cambios se puede analizar en la vista de diferencias que se verá a continuación.

Comparando archivos modificados

Visual Studio Code provee una herramienta básica para ver las diferencias entre archivos lo cual es útil para ver las diferencias entre el archivo el archivo original y el modificado.

Para ello se puede ir a la ventana de control de versiones y en la lista de archivos modificados hacer click sobre uno de los archivos el cual mostrará una vista a continuación en la cual se puede observar del lado izquierdo la versión actual y en el lado derecho la versión modificada en el cual se marca con un color dependiendo de la acción realizada el cambio perse.

En la siguiente imagen se muestra dicha ventana con un ejemplo de agregar un tag de tipo headline a un documento HTML.

