

A dark blue vertical bar on the left side of the slide. A blue arrow points to the right from the bar, containing the date.

25-9-2019

# Tutorial de uso de editores de texto y navegadores

Several thin, curved lines in dark blue and light grey originate from the bottom left corner and curve upwards and to the right.

Francisco José Sánchez Romero

DESARROLLO WEB EN ENTORNO CLIENTE - 2ºDAW

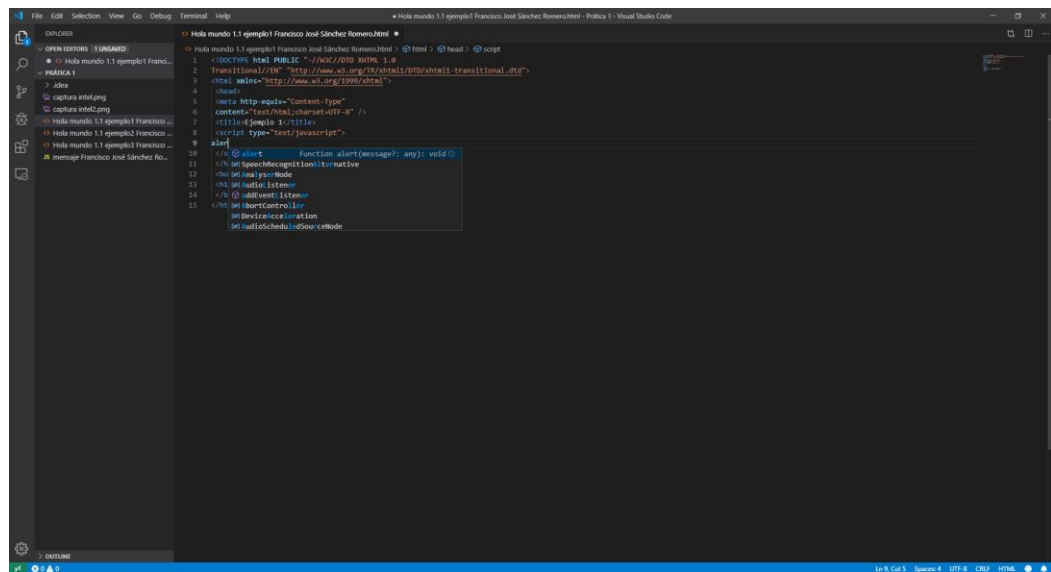
# Índice

	Pág.
1. Instalaciones necesarias para reconocimiento de código html y javascript .....	2
1.1 Visual Studio Code .....	2
1.2 Sublime Text.....	3
1.3 Atom .....	4
2. Atajos en ambos lenguajes .....	5
2.1 HTML.....	5
2.2 JavaScript .....	7
3. Teclas de acceso rápido que muestren las herramientas de desarrollo. ....	9
3.1 Google Chrome.....	9
3.2 Microsoft Edge .....	10
3.3 Opera .....	10
4. Acceso a la consola.....	11
4.1 Google Chrome.....	11
4.2 Microsoft Edge .....	11
4.3 Opera .....	12
5. Acceso a los ficheros .....	12
5.1 Google Chrome.....	12
5.2 Microsoft Edge .....	13
5.3 Opera .....	13
6. Reglas CSS.....	14
6.1 Google Chrome.....	14
6.2 Microsoft Edge .....	14
6.3 Opera .....	15
7. Depurador.....	15
7.1 Google Chrome.....	15
7.2 Microsoft Edge .....	16
7.3 Opera .....	16

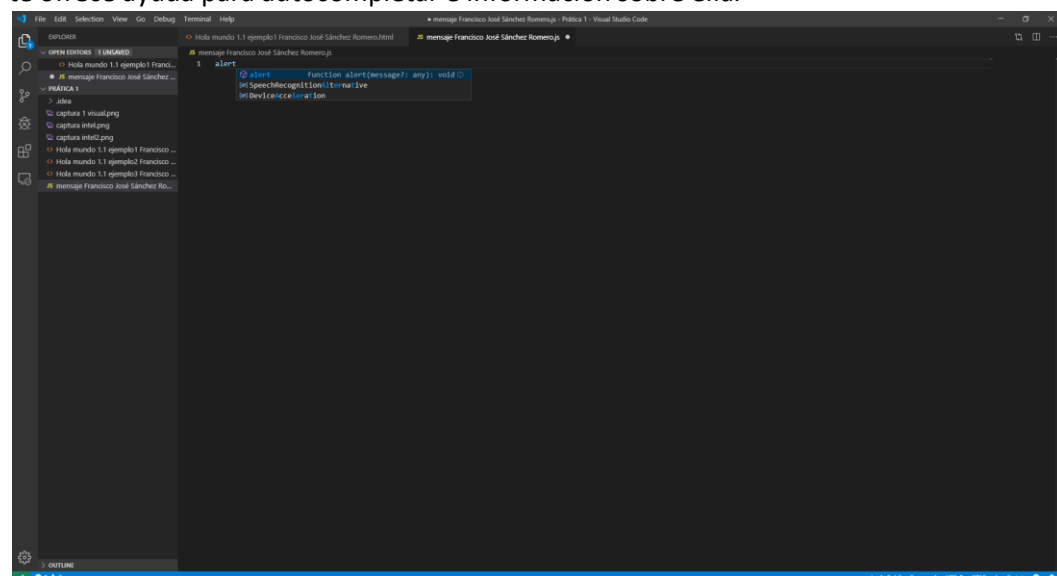
# 1. Instalaciones necesarias para reconocimiento de código html y javascript

## 1.1 Visual Studio Code

En el caso de este editor de texto, no ha sido necesario la instalación de ningún ‘plugin’ adicional para el reconocimiento de código html y javascript. Simplemente al guardar el documento con extensión ‘.html’, la herramienta reconoce automáticamente que se trata de un documento html y javascript.



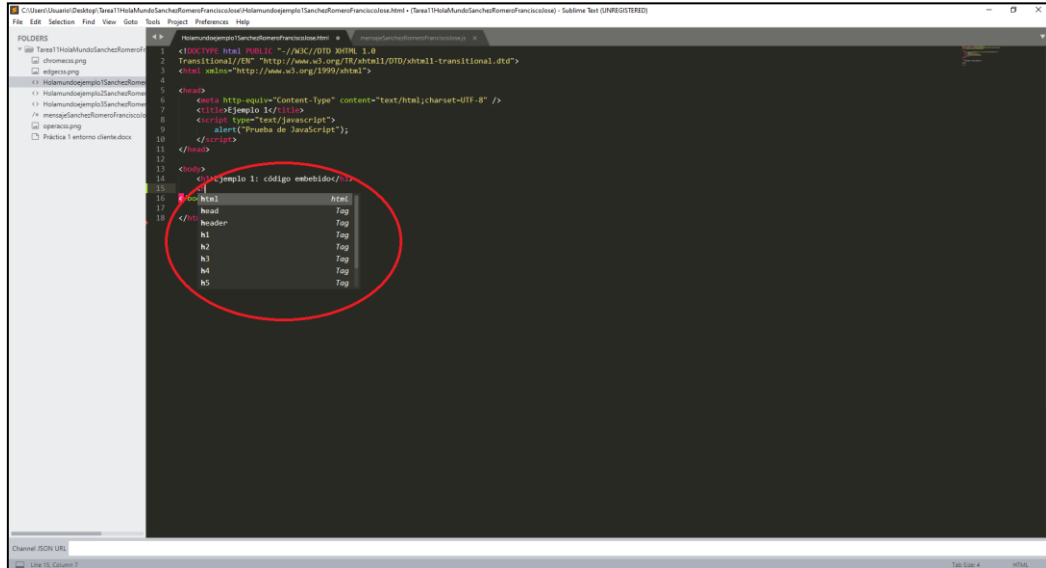
Como se puede observar en la imagen superior, al trabajar con un documento con extensión html, reconoce automáticamente la etiqueta que estás intentando escribir y te ofrece ayuda para autocompletar e información sobre ella.



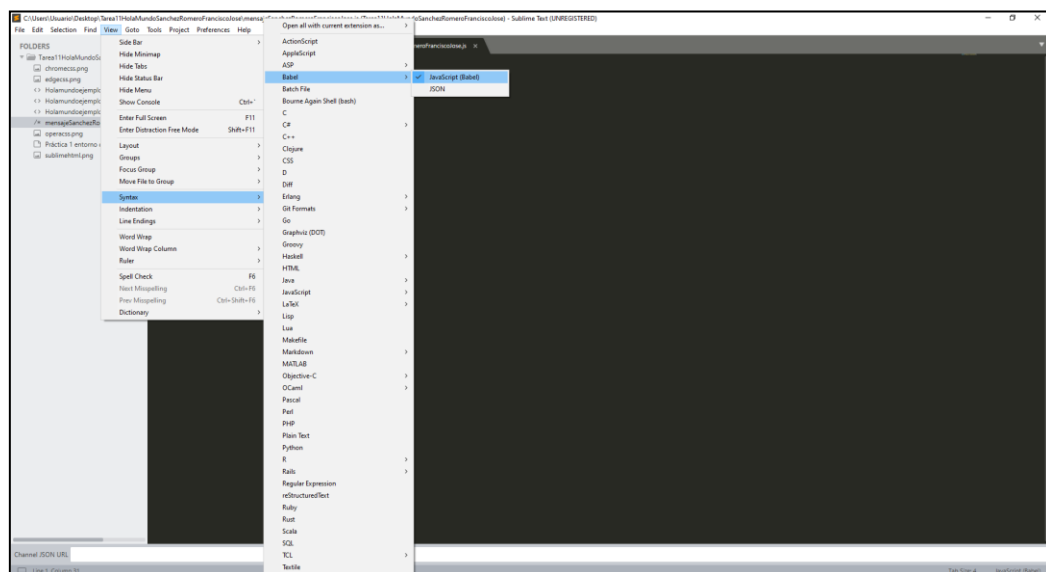
Al igual ocurre cuando hablamos de un documento con extensión ‘.js’, como podemos observar en la imagen superior, el editor de texto nos facilita el autocompletado de la etiqueta.

## 1.2 Sublime Text

En el caso de este editor, para el reconocimiento de HTML no ha sido necesaria la instalación de ningún plugin adicional, simplemente con la extensión ‘.html’ en nuestro fichero el editor es capaz de reconocer de que se trata de un fichero html y ofrecernos ayuda durante la edición. Como se puede ver en la siguiente imagen.

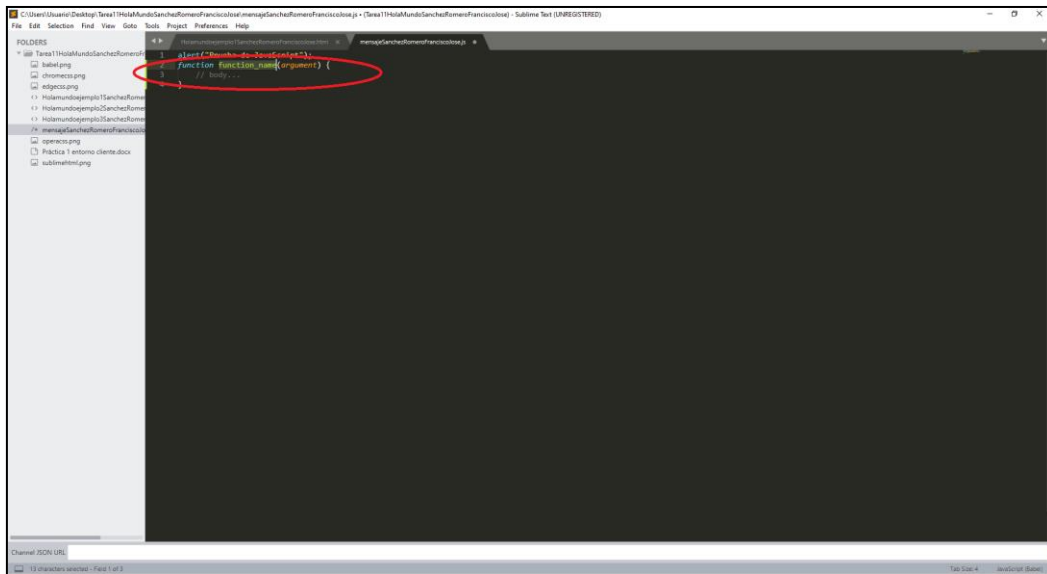


Sin embargo, cuando hablamos de ficheros JavaScript, ha sido necesaria la instalación de un plugin llamado ‘Babel’, el cual nos ofrece ayuda durante la escritura y autocompleta las etiquetas, además de ofrecernos información de ellas.



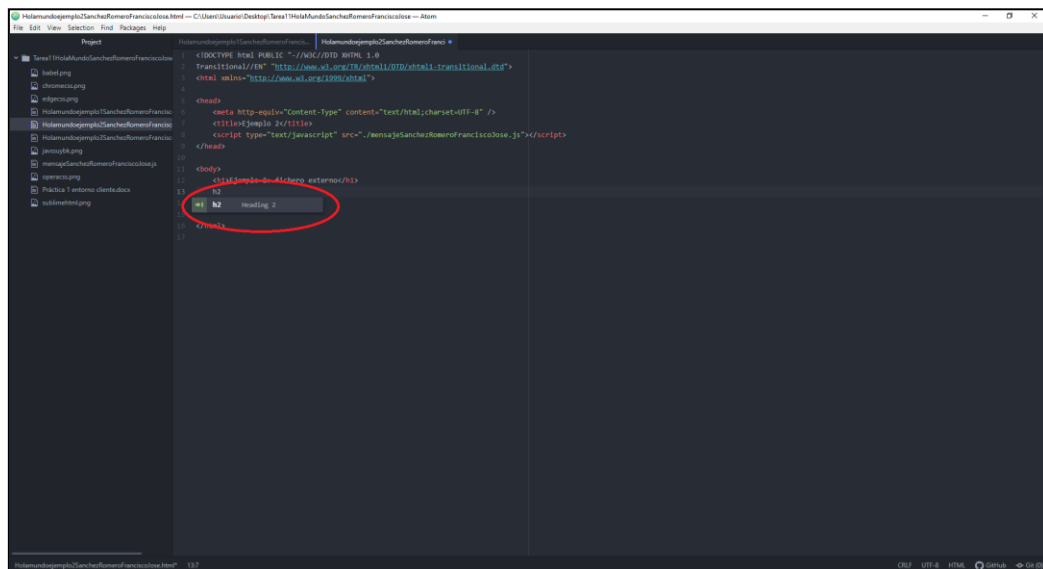
En la imagen superior se muestra como se debe de activar el plugin para que funcione correctamente a la hora de editar fichero JavaScript.

En la imagen inferior se muestra como el propio editor, gracias a la extensión, nos permite autocompletar en este caso la forma que tendría una función.



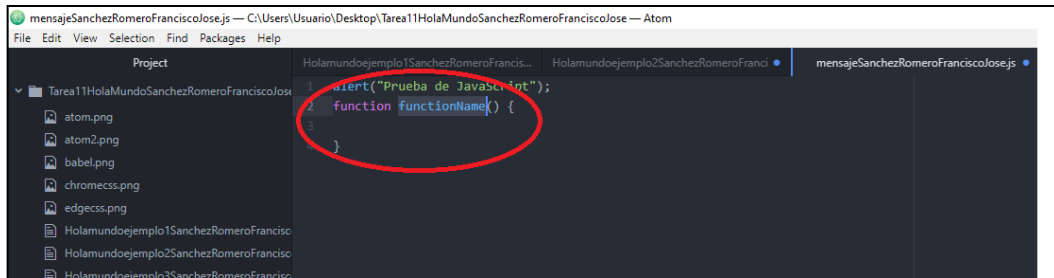
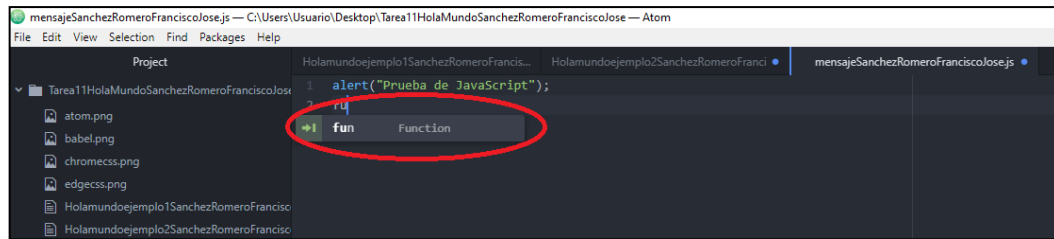
### 1.3 Atom

Respecto a Atom, este editor de texto al igual que Visual Studio Code trae automáticamente incorporadas funciones para reconocer el código html y javascript durante la edición de los ficheros. Como se puede observar en la siguiente imagen, al tener la extensión '.html', el editor nos ayuda a autocompletar las etiquetas de este lenguaje.



Respecto a JavaScript, ocurre lo mismo, el editor es capaz de reconocer y ofrecer ayuda sobre el lenguaje sin la necesidad de la instalación de ningún paquete adicional.

Como se puede observar en las siguientes imágenes.

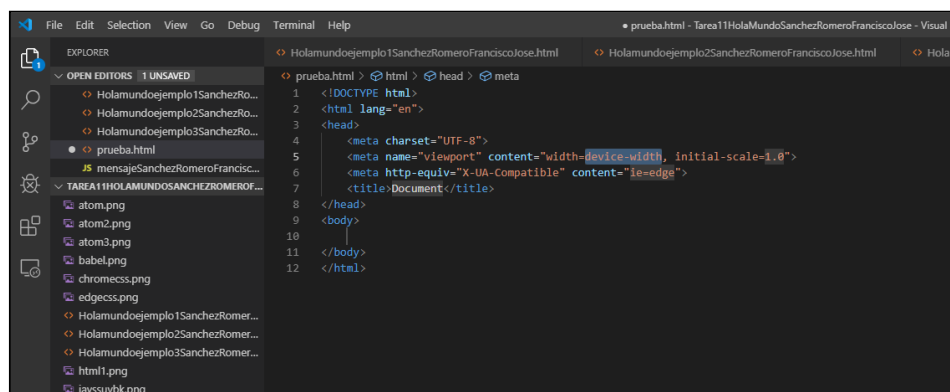
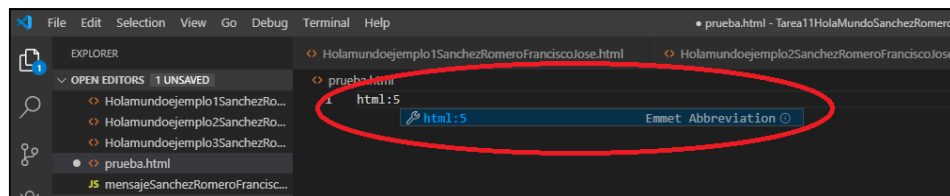


## 2. Atajos en ambos lenguajes

### 2.1 HTML

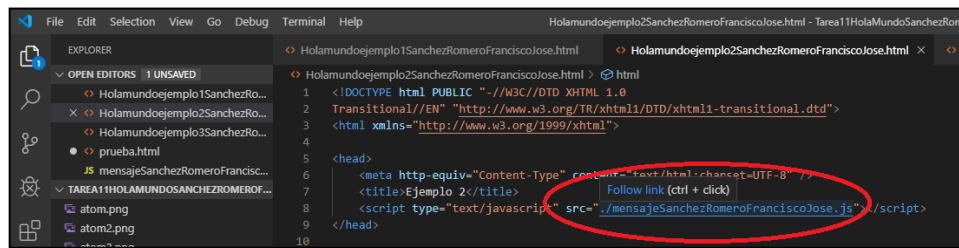
Respecto a los atajos durante la edición, hay que aclarar que en los tres editores de texto se ha instalado un plugin llamado 'Emmet', el cual ayuda a muchos de los atajos que se van a presentar a continuación.

1. **Autoescritura de la plantilla de HTML (Emmet):** gracias al plugin anteriormente mencionado, en los tres editores al escribir el atajo 'html:5' se nos creará una plantilla básica de html, como se puede observar en las siguientes imágenes.

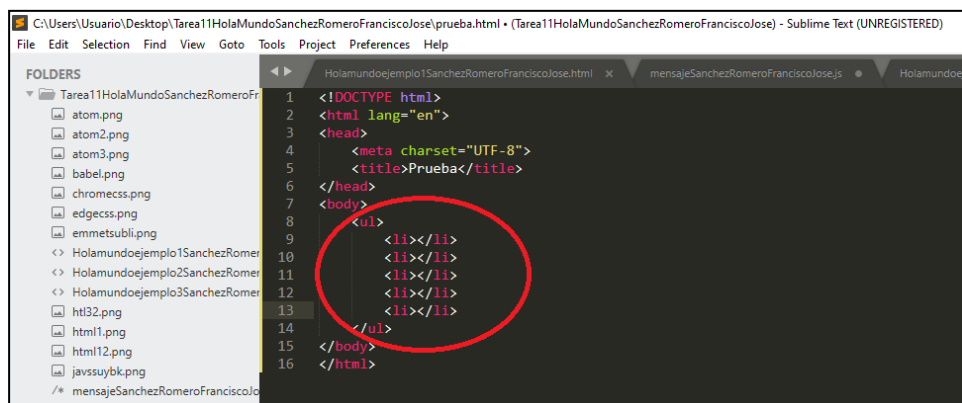


2. **Función de navegado a ficheros externos:** en el caso de Visual Studio Code, no se ha necesitado de ninguna herramienta externa para que esta función haga su trabajo correctamente, sin embargo en los otros dos editores de texto la función se realiza gracias al plugin 'Emmet'. Al colocar el puntero sobre un enlace dentro de un atributo 'src', al hacer la combinación 'Ctrl+Click' el editor navegará

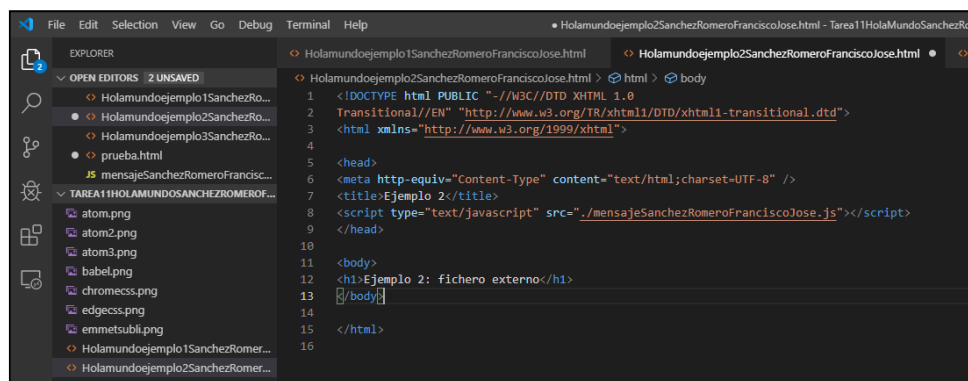
automáticamente hasta el fichero deseado, como se puede ver en la siguiente imagen.

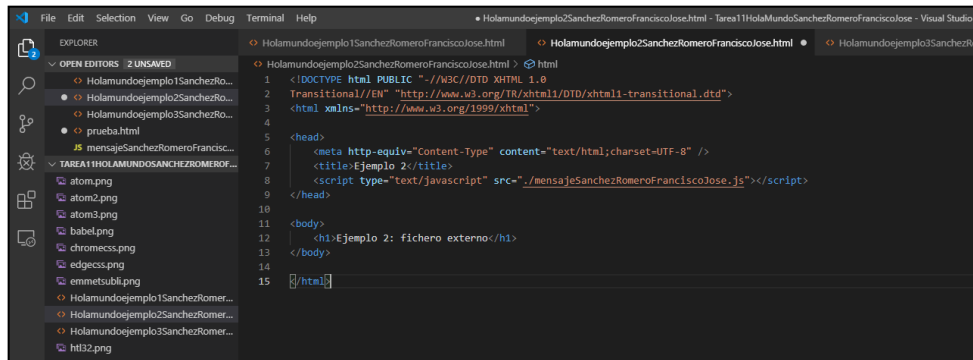


3. **Autocompletado de etiquetas mediante multiplicaciones:** en los tres editores, gracias al plugin 'Emmet', se puede crear una gran plantilla respecto a las etiquetas que se quieran usar, por ejemplo cuando hablamos de la etiqueta `<ul>`. Al escribir en el editor `'ul>li*5'` este nos creará una plantilla con cinco ítems, como se puede ver en las siguientes imágenes.

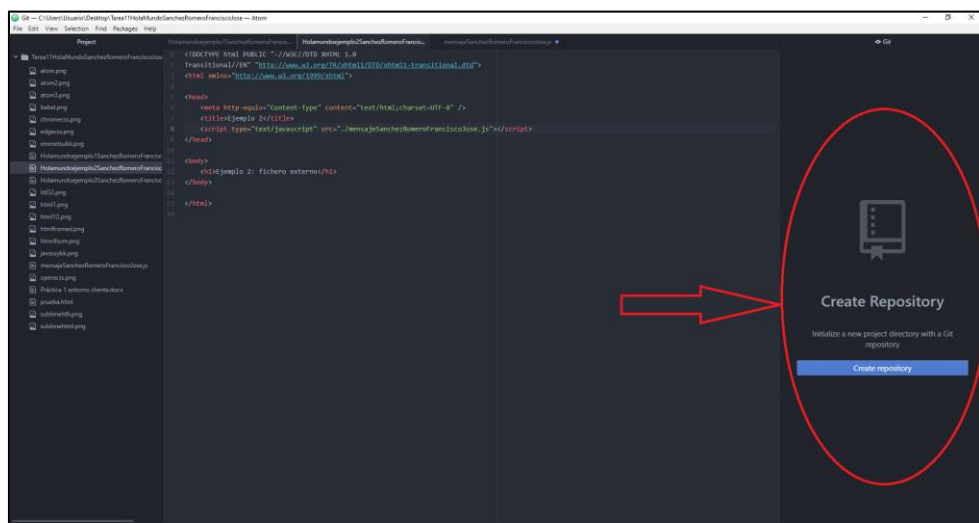


4. **Formateo del documento:** este atajo se encuentra presente de forma automática en Visual Studio Code, al hacer la combinación de teclas 'Shift+Alt+F', el editor nos formateará el documento metiendo las tabulaciones necesarias que se hayan configurado, como se puede ver en la siguiente imagen.



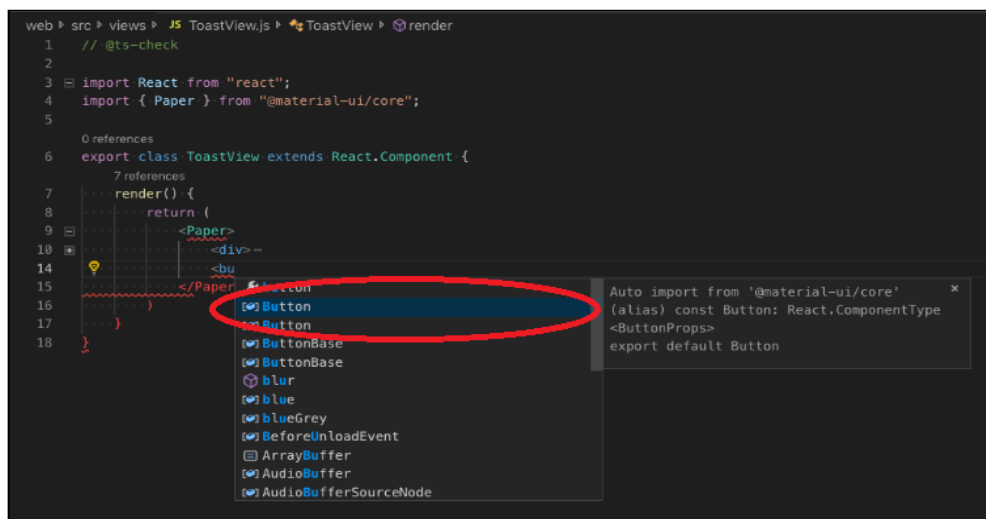


5. **Añadir repositorio en Git:** el editor Atom, desde su propia interfaz permite añadir el proyecto a un repositorio en Git, como se puede ver en la siguiente imagen.



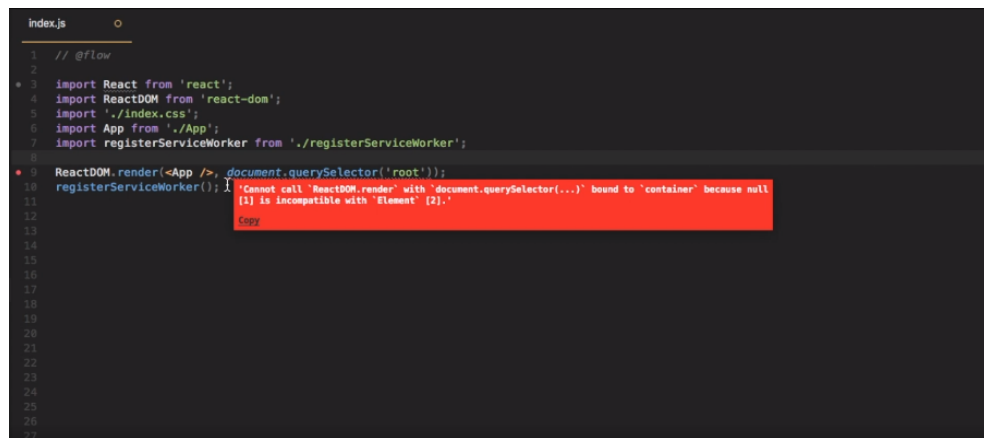
## 2.2 JavaScript

- 1.- **Autoimportado de paquetes:** en Visual Studio Code, el editor nos da la opción de que se importen automáticamente en nuestro proyecto los paquetes necesarios si queremos utilizar alguna librería durante la edición, como se puede observar en la siguiente imagen.

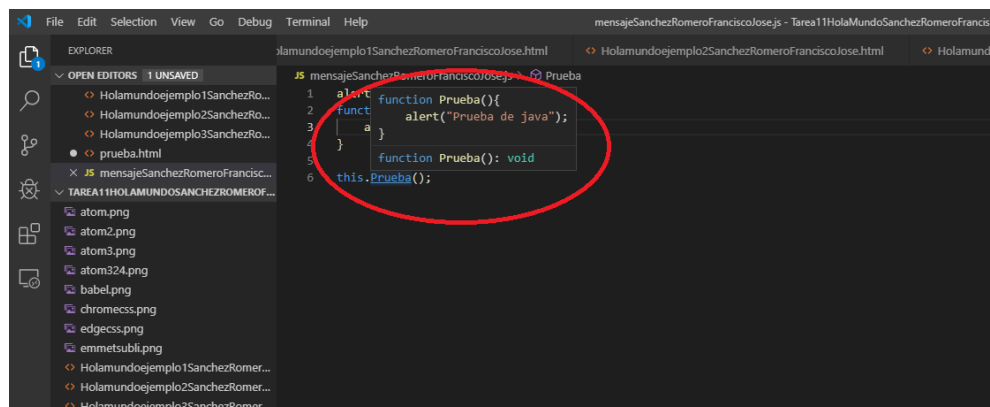




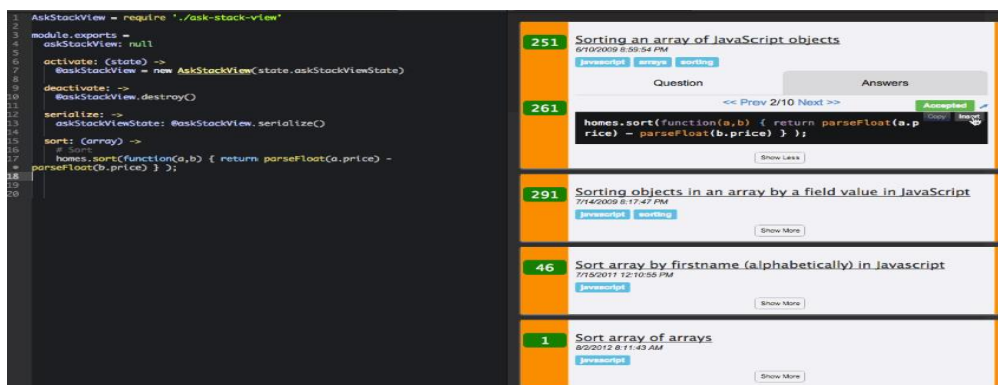
**2.- JavaScript Enhancement Plugin para Sublime Text 3:** este plugin nos ofrecerá mucha ayuda para el editor sublime text si queremos trabajar con ficheros javascript, ya que nos informará de errores proporcionándonos información sobre él y varias sugerencias, como se puede ver en la imagen de abajo.



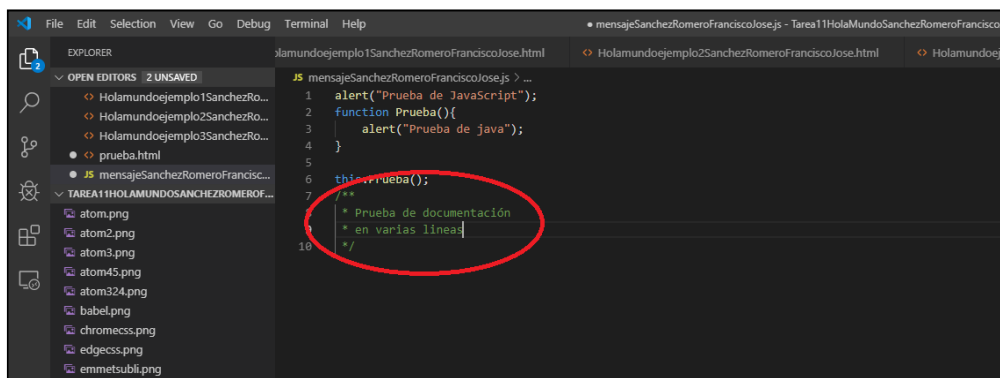
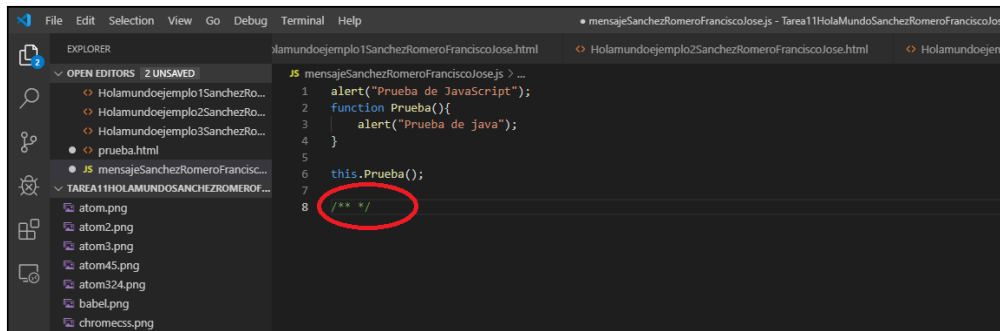
**3.-Navegar hacia las funciones:** si estamos trabajando con varios ficheros y deseamos acceder a una función que se está llamando en uno de ellos, Visual Studio nos ofrece la posibilidad de navegar hacia la función con la combinación 'Ctrl+Click' en el lugar donde se esté llamando a esa función, como se puede ver en la siguiente imagen.



**4.- Ask Stack-Overflow para Atom:** este plugin nos permite desde el mismo editor hacer consultas en la página StackOverflow y utilizar el código similar a la búsqueda que hayamos realizado. Como ejemplo se adjunta la siguiente imagen.



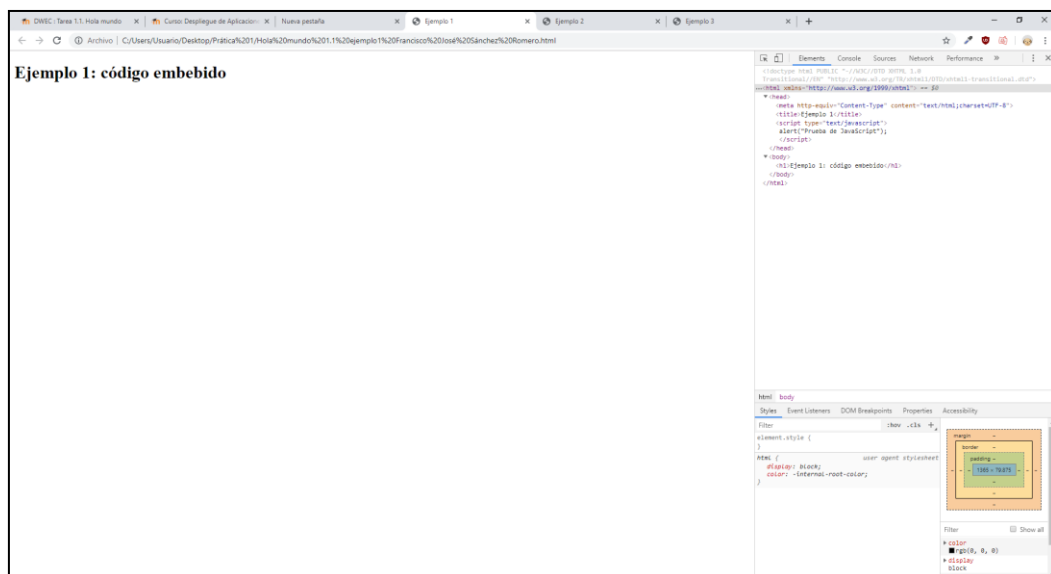
**5.- Autocompletado para la documentación:** en Visual Studio Code, podemos crear nuestra propia plantilla para documentar el fichero correctamente, para ello simplemente debemos de escribir `/**` y el editor nos autocompletará la documentación automáticamente, como se puede ver en las siguientes imágenes.



### 3. Teclas de acceso rápido que muestren las herramientas de desarrollo.

#### 3.1 Google Chrome

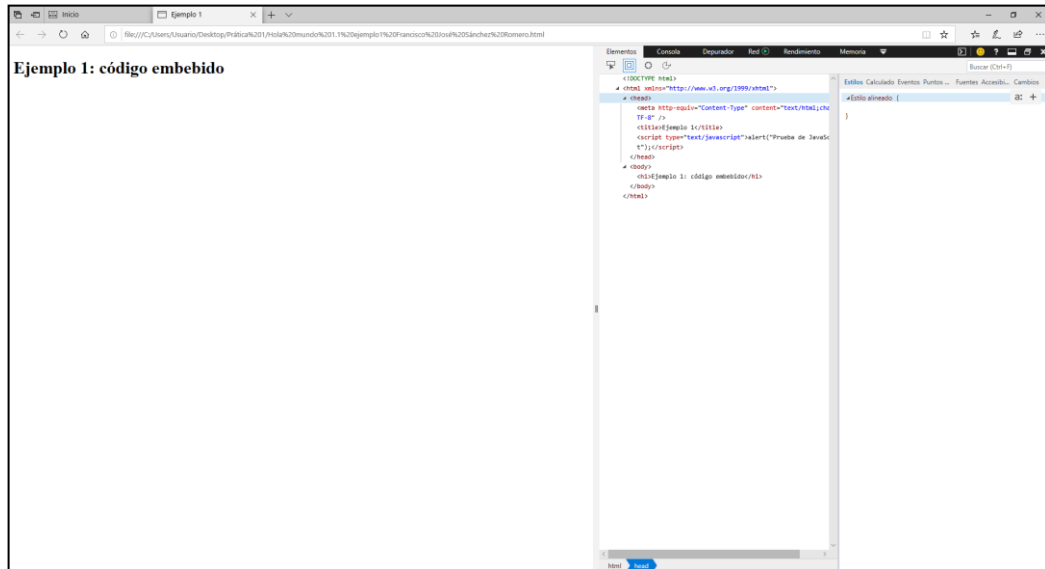
En el caso de este navegador, para poder acceder a las herramientas de desarrollo se debe de pulsar una combinación de teclas, `'Ctrl+Shift+I'` o simplemente pulsando la tecla `'F12'` de nuestro teclado.



Como se puede observar en la imagen superior, en la parte derecha de nuestra pantalla se abrirá una nueva pestaña con el árbol DOM de nuestro documento, además de poder acceder a diversas herramientas.

### 3.2 Microsoft Edge

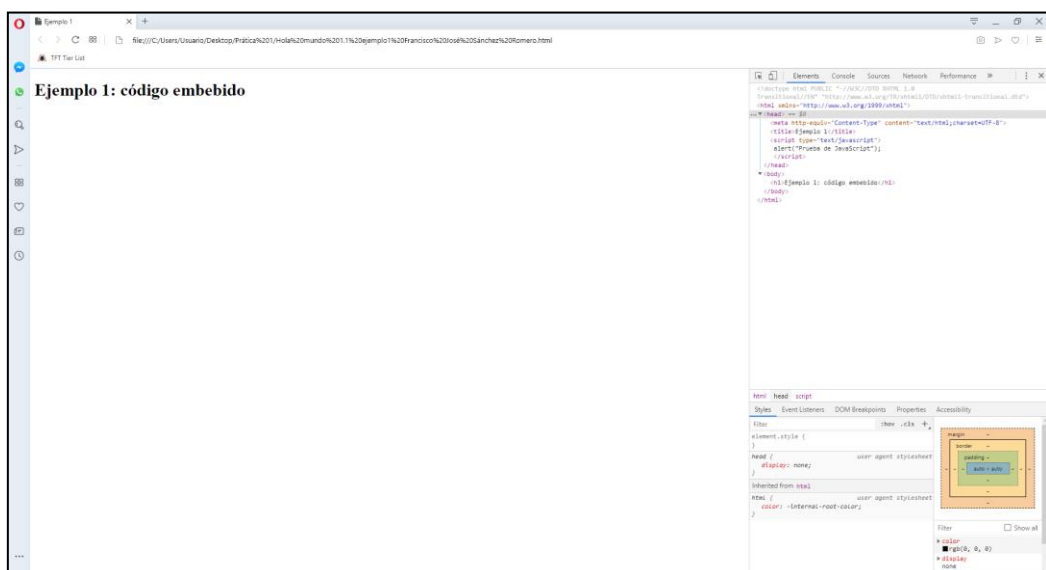
Este navegador ofrece las mismas formas que el anterior para acceder a las herramientas de desarrollo, simplemente pulsado la combinación 'Ctrl+Shift+I' o la tecla 'F12', la herramienta de desarrollo se abrirá automáticamente.



Como se puede observar en la imagen, la herramienta de desarrollo aparecerá en la parte derecha de nuestra pantalla.

### 3.3 Opera

Al igual que los dos navegadores anteriores, Opera abrirá la herramienta de desarrollo con la combinación de teclas 'Ctrl+Shift+I' o 'Ctrl+Shift+C', sin embargo, no te da la opción de poder abrirla pulsando la tecla 'F12'.

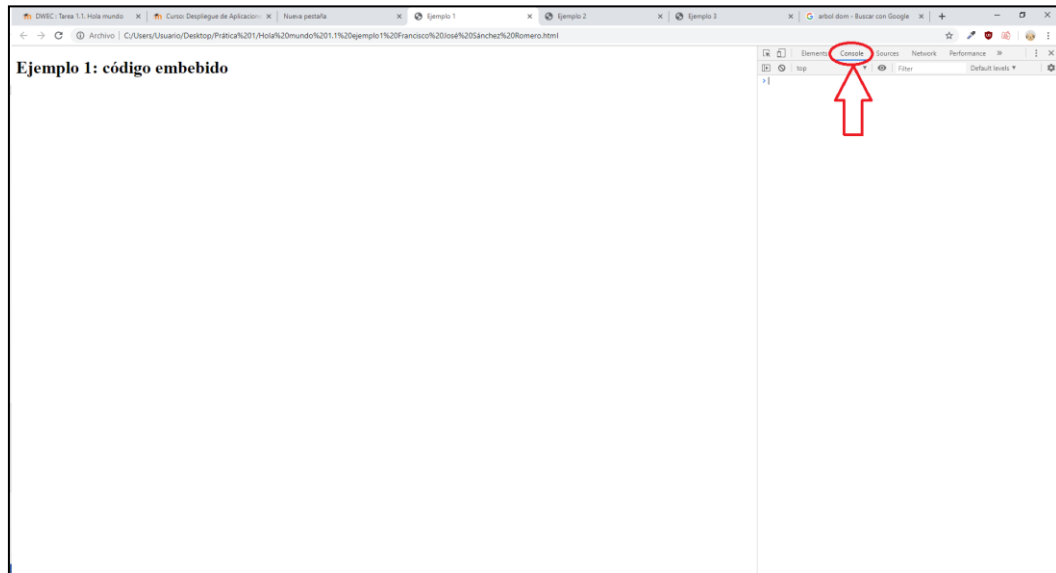


Como se puede observar en la imagen, la herramienta de desarrollo aparecerá a la derecha de nuestra pantalla.

## 4. Acceso a la consola

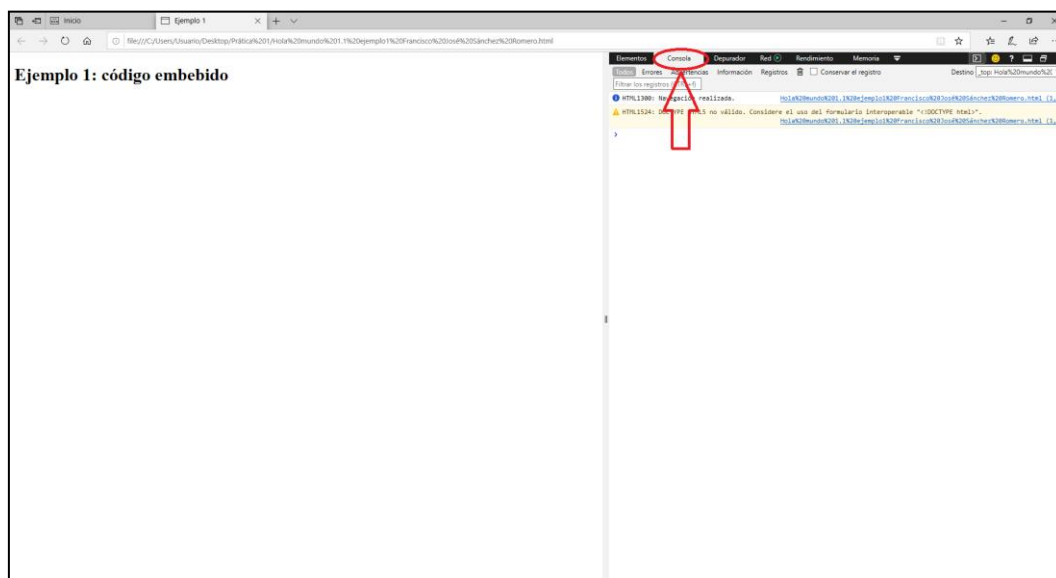
### 4.1 Google Chrome

Para poder acceder la consola, debemos de tener abierta la herramienta de desarrollo tal y como se indica en el punto 3.1 y simplemente debemos de pulsar sobre la pestaña 'Console' que aparecerá, como podemos ver en la siguiente imagen.



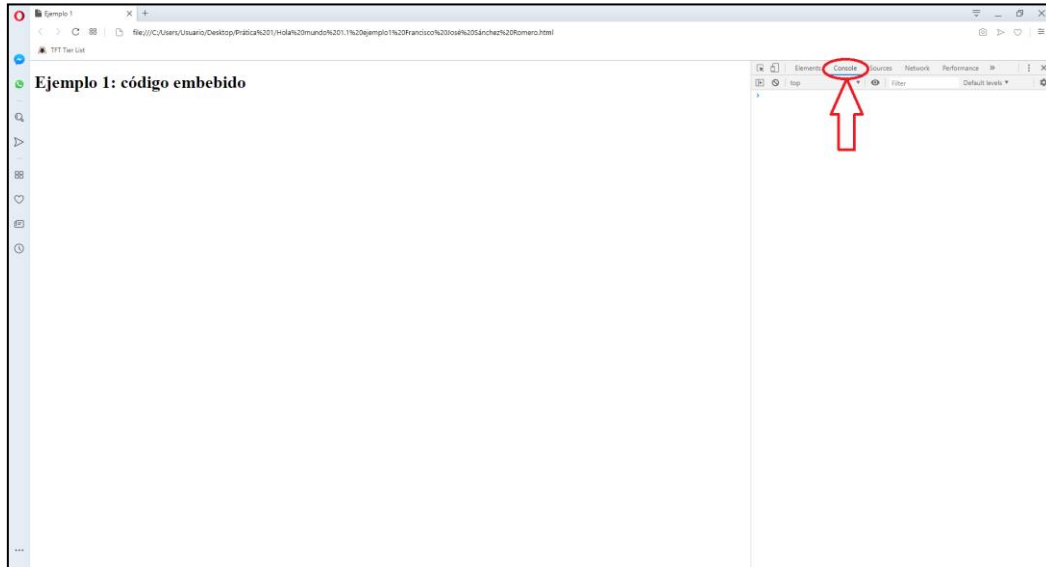
### 4.2 Microsoft Edge

En el caso de este navegador, para acceder a la consola de desarrollo debemos de seguir los pasos indicados en el punto 3.2 y abrir la herramienta de desarrollo, una vez abierta debemos de pulsar sobre la pestaña 'Consola' que aparece como se puede observar en la siguiente imagen.



### 4.3 Opera

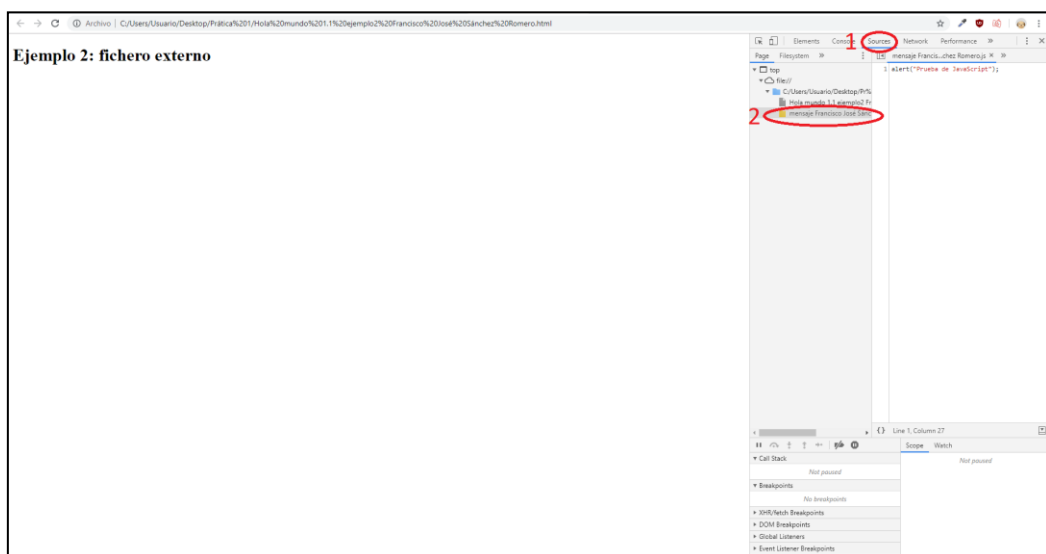
Para poder acceder a la consola en este navegador, deberemos de seguir los pasos del punto 3.3 y abrir la herramienta de desarrollo, una vez realizado, deberemos de pulsar en la pestaña 'Console' como se observa en la siguiente imagen.



## 5. Acceso a los ficheros

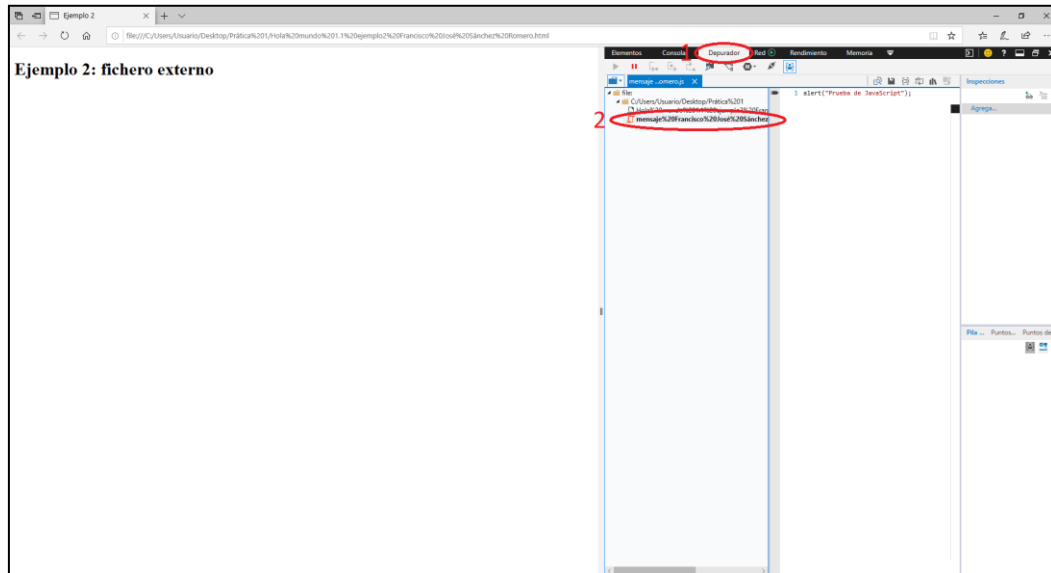
### 5.1 Google Chrome

Para poder acceder a un fichero javascript externo, debemos de tener abierta la herramienta de desarrollo y pulsar sobre la pestaña 'Sources', en ella aparecerán los archivos que usa este documento, en este caso debemos de seleccionar el fichero .js que aparece, tal y como se muestra en la siguiente imagen.



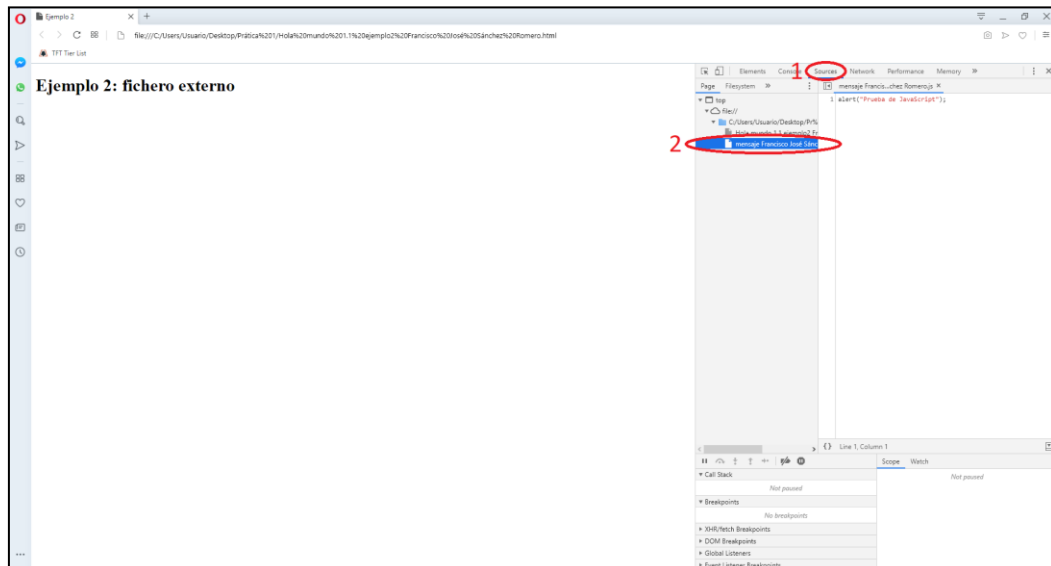
## 5.2 Microsoft Edge

En este navegador, para poder acceder a ficheros javascript externo debemos de tener abierta la herramienta de desarrollo y pulsar sobre la pestaña ‘Depurador’, en esta pestaña debemos de elegir el fichero con extensión ‘.js’ deseado tal y como se puede observar en la siguiente imagen.



## 5.3 Opera

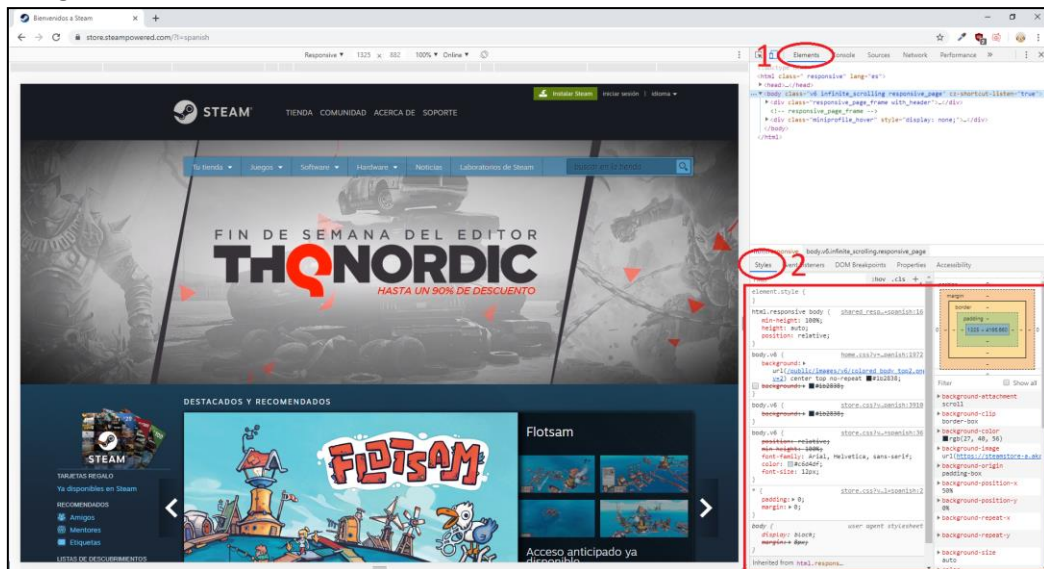
En este navegador, debemos de tener la herramienta de desarrollo abierta y acceder a la pestaña ‘Sources’ y seleccionar el fichero externo que deseemos visualizar, como se puede en observar en la imagen de a continuación.



## 6. Reglas CSS

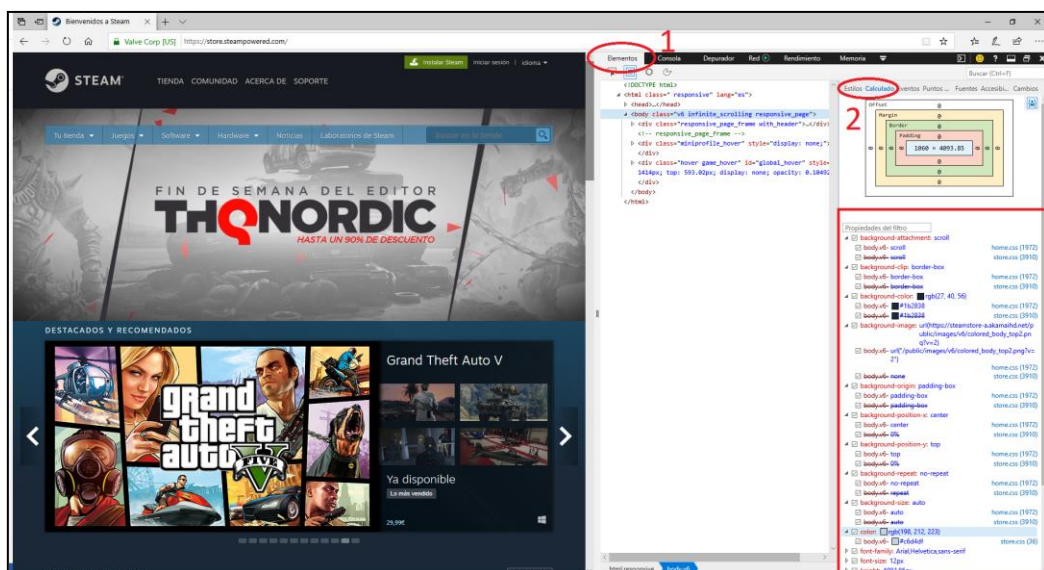
### 6.1 Google Chrome

Para poder acceder a nuestra hoja de estilos CSS, debemos de tener la herramienta de desarrollo abierta y pulsar sobre la pestaña 'Elements', en la parte inferior podremos observar que hay otra pestaña llama 'Styles' sobre la cual debemos de pulsar y asi poder acceder a los estilos de nuestro documento, como se puede apreciar en la siguiente imagen.



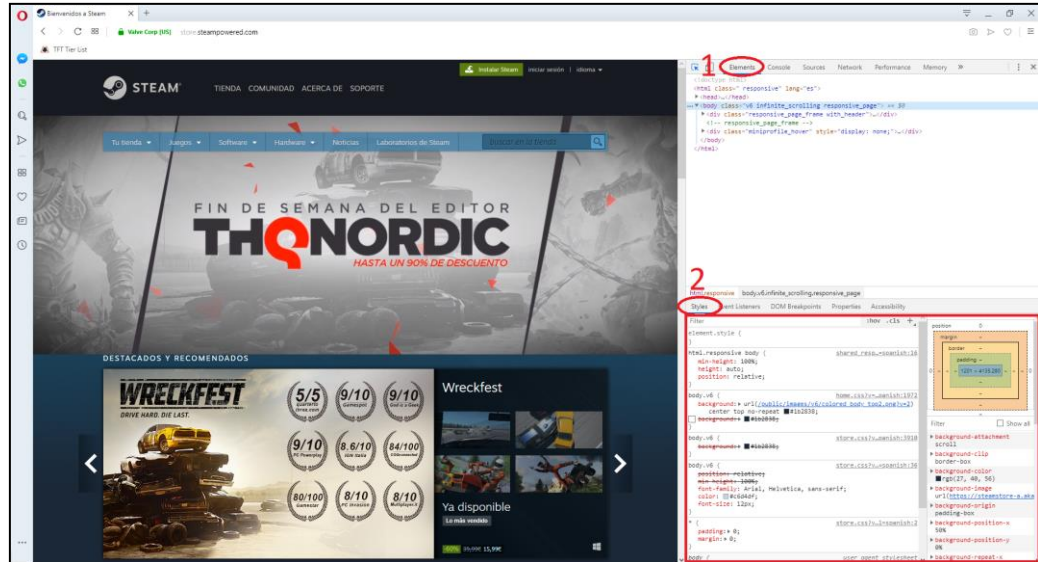
### 6.2 Microsoft Edge

En este navegador, si deseamos acceder a nuestra hoja de estilos debemos de tener la herramienta de desarrollo abierta y pulsar sobre la pestaña 'Elementos' y dentro de ella en la parte derecha, pulsar sobre la pestaña 'Estilos' pudiendo acceder asi a nuestra hoja de estilos deseada, tal y como se puede observar en la siguiente imagen.



### 6.3 Opera

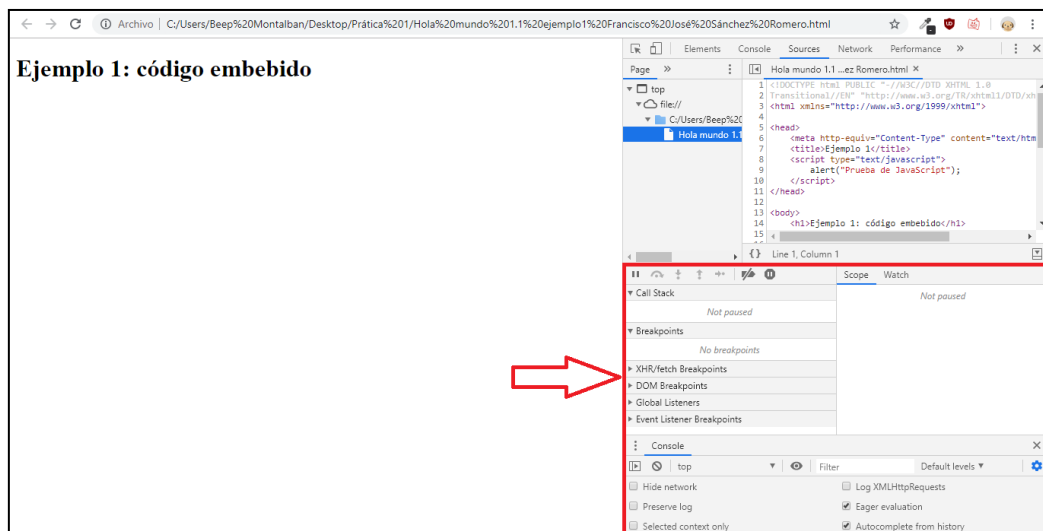
En opera, al igual que en los demás navegadores debemos de tener la herramienta de desarrollo abierta para poder acceder a nuestra hoja de estilos, en ella debemos de pulsar sobre la pestaña 'Elements' y en la parte inferior pulsar en la pestaña 'Styles' observando asi nuestra hoja de estilos, como se muestra en la siguiente imagen.



## 7. Depurador

### 7.1 Google Chrome

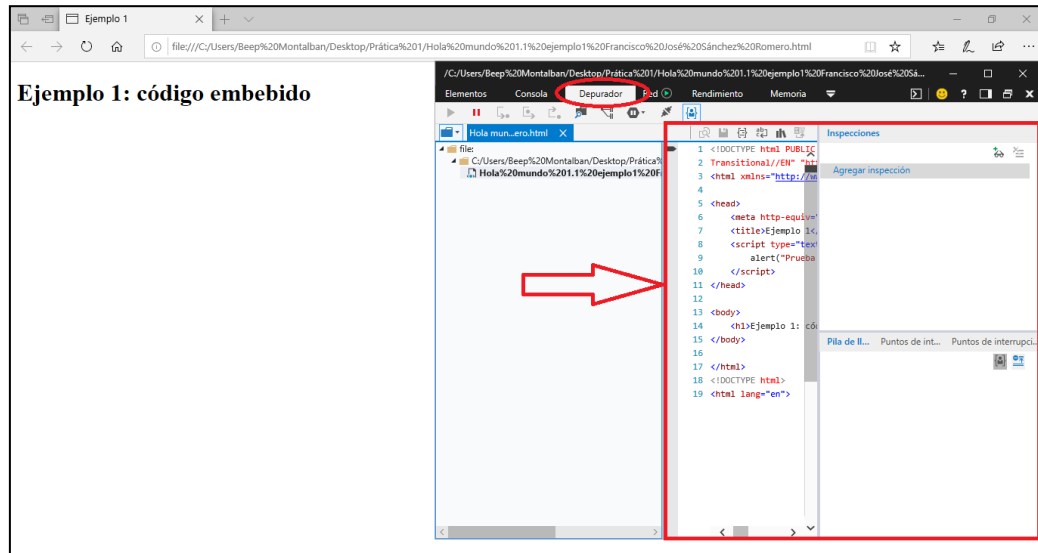
En este navegador, para poder acceder al depurador se debe de abrir la herramienta de desarrollo y acceder a la pestaña de 'Sources' y en la parte inferior aparece el depurador con los 'breakpoints'/puntos de interrupción que hemos marcado, como se puede observar en la siguiente imagen.





## 7.2 Microsoft Edge

Para acceder al depurador en este navegador debemos de abrir la herramienta de desarrollo y acceder a la pestaña 'Depurador', en ella podremos ver el árbol DOM de nuestro documento además de los puntos de interrupción creados, como se observa en la siguiente imagen.



## 7.3 Opera

Al igual que Chrome, para acceder al depurador se debe de tener abierta la herramienta de desarrollo y acceder a la pestaña de 'Sources', en ella, situado en la parte inferior podemos ver el depurador junto con los puntos de interrupción o 'breakpoints' que hemos creado, como se muestra en la siguiente imagen.

