

Antes de comenzar con JavaScript vamos a ver una panorámica de las herramientas que nos pueden ser útiles, y las que vamos a utilizar para trabajar.

HERRAMIENTAS DE TRABAJO.

Teóricamente para escribir JavaScript y HTML solo es necesario un editor de texto, como podría ser el block y un navegador. Pero hay herramientas que nos van a hacer más fácil nuestro trabajo. Algunas de estas herramientas son:

- **Editores de texto.**

Hay una gran alternativa a la hora de elegir los editores de texto, por ello veremos cual nos va a ayudar más en nuestro trabajo. Algunos de ellos son:

➤ **Notepad++:** editor gratuito que se puede descargar desde el sitio <http://notepad-plus-plus.org/>.

su interfaz es sencilla y de fácil uso. Dispone de una serie de características muy útiles:

- Resaltado de la sintaxis.
- Agrupación de partes homogéneas de código para poder ocultar o mostrar partes de un documento largo.
- Resaltado de la sintaxis y Sintaxis Folding personalizado por el usuario.
- Resaltado de los paréntesis.
- Búsqueda/reemplazo mediante expresiones regulares.
- Función autocompletar de la sintaxis.
- Marcadores.
- Pantalla con pestañas.
- Visualización de documentos en paralelo para su comparación.

➤ **Atom:** editor gratuito, se puede descargar desde <http://atom.io/> disponible para distintas plataformas (OSX, Windows y Linux). Puede ser completado con distintos paquetes de código abierto y dispone de soporte para el sistema de control de versiones Git.. entre los puntos más fuertes de este editor se encuentran:

- Función de Autocompletar.

- Resaltado de la sintaxis.
- Función de búsqueda y reemplazo entre distintos archivos.
- Posibilidad de abrir varios archivos en paneles para poder compararlos.

➤ **Visual Studio:** este es el editor que vamos a utilizar. Esta desarrollado por Microsoft para diversas plataformas (OS X, Windows y Linux). Se trata de una herramienta gratuita que se puede descargar desde la página: <http://code.visualstudio.com/>. dispone de Git integrado y se puede completar con otros paquetes. Entre sus puntos fuertes se encuentran:

- Función Autocompletar.
- Resaltado de sintaxis.
- Función de búsqueda y reemplazo entre distintos archivos.
- Posibilidad de fijar Breakpoints o puntos de interrupción.
- Trabaja directamente con archivos y carpetas sin necesidad de crear proyecto,

- **Linter.**

Es un programa que por lo general se integra con un editor de código y permite resaltar los errores de sintaxis o en general, de escritura del código.

Uno de los linters para JavaScript más conocido es el ESLint.:

- **Servidores web.**

Al principio los ejercicios los podremos ejecutar desde el mismo sistema de archivos, pero hay casos en los que se necesita hacerlo desde un servidor. Podemos utilizar un servicio online o bien se puede instalar un servidor web en el ordenador local, como puede ser SAMPP.

VISUAL STUDIO CODE.

Editor fácil de utilizar y disponible para distintos sistemas operativos, como Windows, MacOS y Linux.

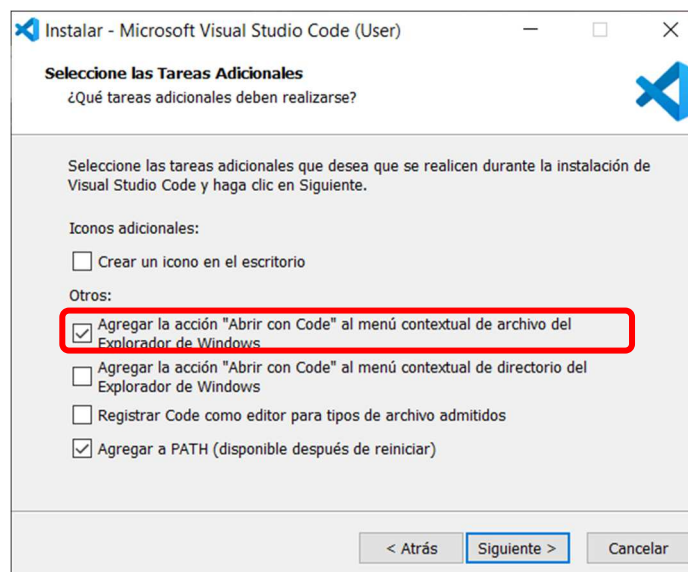
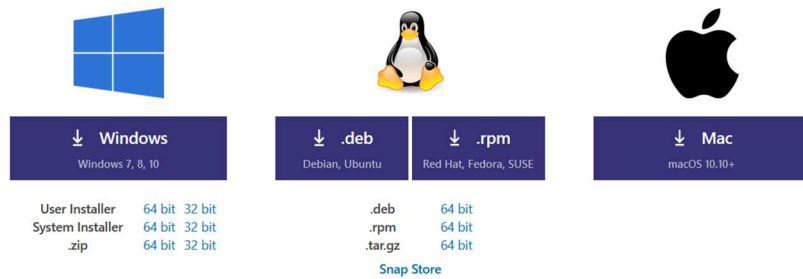
Está dotado de un soporte nativo par JavaScript, TypeScript y Node.js y además dispone de distintas extensiones para otros lenguajes como C++, C#, Python, PHP y Go.

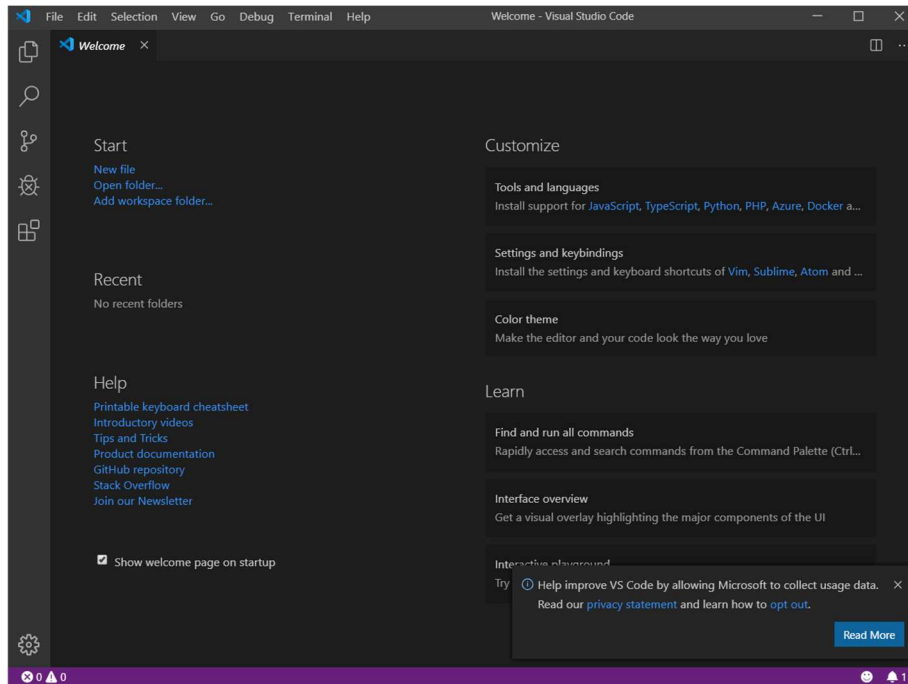
VS Code se actualiza cada mes de forma automática, así el editor siempre se encuentra en el mejor estado.

Se puede descargar desde : <http://code.visualstudio.com/>.

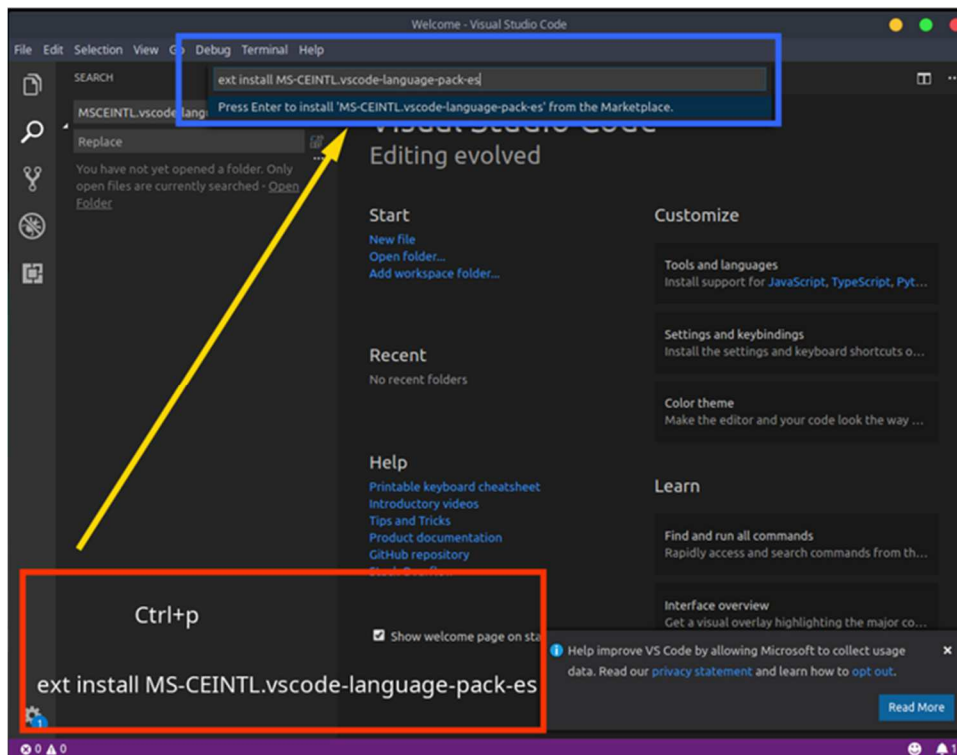
Download Visual Studio Code

Free and built on open source. Integrated Git, debugging and extensions.

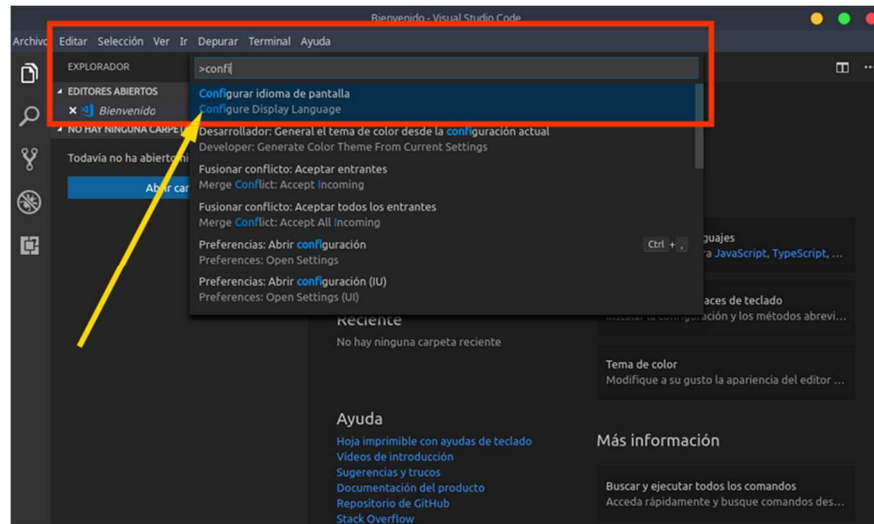




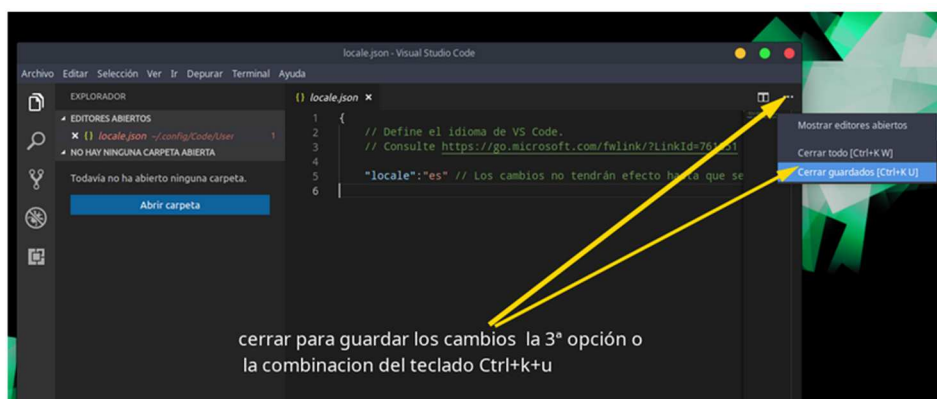
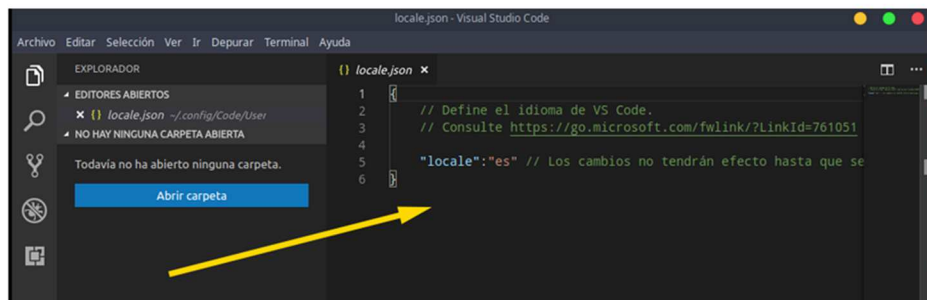
Abrimos el Visual Studio pack, para ello pulsamos las teclas: **CTRL + p**, lo que hace que se abra un ventano de comandos, en la que escribiremos lo siguiente: **ext install MS-CEINTL.vscode-language-pack-es**



Una vez que ya se ha cambiado el idioma a español, para que se conserven los cambios para siempre se debe modificar **locale.json**, para ello pulsamos las teclas **CTRL +SHIFT + p** para activar el “**Command Palette**”, comenzamos a teclear “**config**” para filtrar la lista de comandos disponibles.



Y luego seleccionamos el comando **Configure Display Language** y elegimos “**es**”. Guardamos



VS Code no dispone de herramientas de control del código durante la escritura, menos que se las instale de forma explícita. Es conveniente tenerlas instaladas. En primer lugar, necesitamos Node.js. Node.js es un entorno donde se puede ejecutar aplicaciones escritas en JavaScript. De estas, necesitamos la aplicación ESLint, que es la que vamos a utilizar con VS Code.

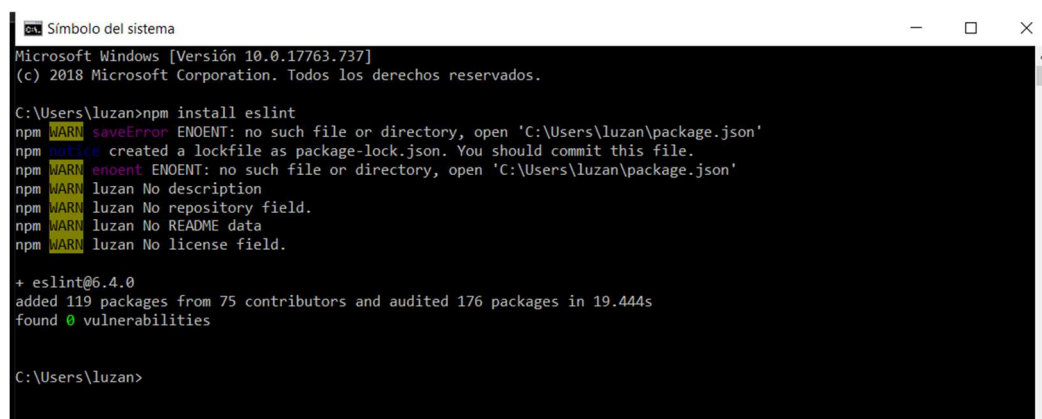
Para instalarla deberemos hacer lo siguiente:

- 1.- Descargar e instalar Node.js, lo cual lo haremos desde la página <https://nodejs.org>. Instalaremos la aplicación con los parámetros adecuados: damos a todo siguiente hasta el final.
- 2.- Para instalar el módulo ESLint de Node.js utilizando el programa NPM incluido en Node.js. esta instalación se lleva a cabo desde el símbolo del sistema de Windows. En **cmd** pondremos **npm install eslint** y pulsamos **ENTER**.



```
Símbolo del sistema
C:\Users\luzan>npm install eslint
```

- 3.- Instalar el pluggin ESLint dentro del VS Code. Pulsamos sobre el botón **Extensiones** (el último en la parte inferior de la barra de botones de la izquierda) y buscamos la extensión **ESLint**; a continuación, hacer clic sobre **Instalar**.



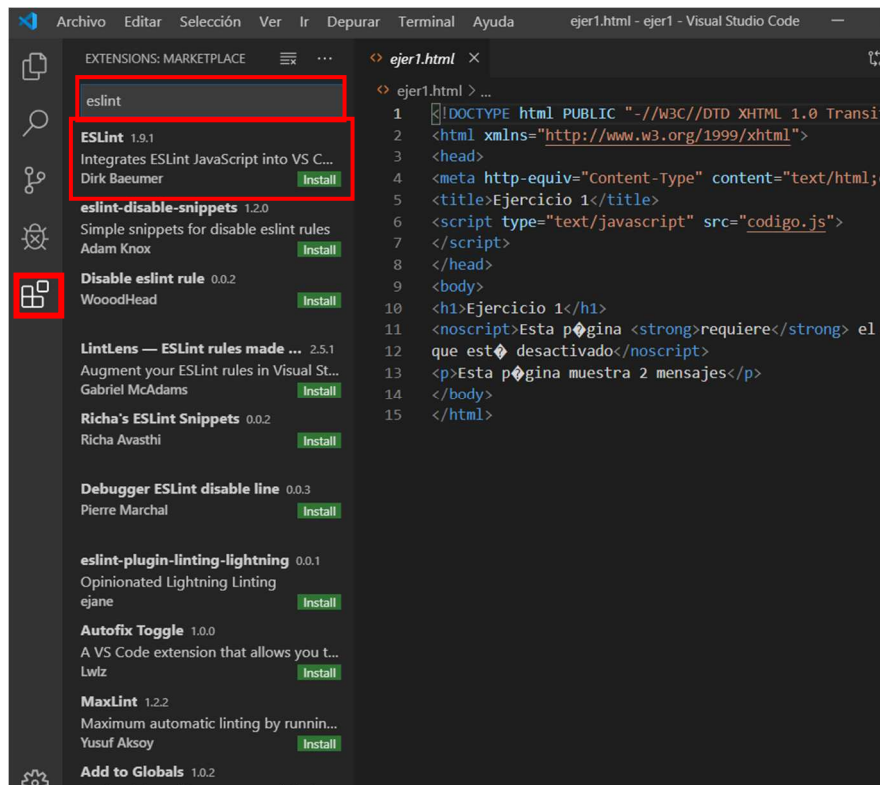
```
Símbolo del sistema
Microsoft Windows [Versión 10.0.17763.737]
(c) 2018 Microsoft Corporation. Todos los derechos reservados.

C:\Users\luzan>npm install eslint
npm WARN saveError ENOENT: no such file or directory, open 'C:\Users\luzan\package.json'
npm notice created a lockfile as package-lock.json. You should commit this file.
npm WARN enoent ENOENT: no such file or directory, open 'C:\Users\luzan\package.json'
npm WARN luzan No description
npm WARN luzan No repository field.
npm WARN luzan No README data
npm WARN luzan No license field.

+ eslint@6.4.0
added 119 packages from 75 contributors and audited 176 packages in 19.444s
found 0 vulnerabilities

C:\Users\luzan>
```

Al finalizar la instalación, el botón **Instalar** pasa a ser **Recargar**, hacemos clic sobre él para volver a cargar VS Code con el pluggin que acabamos de instalar.



4.- Crear el archivo de configuración ESLint en la carpeta/proyecto en el cual se está trabajando. Una vez abierta la carpeta/proyecto seleccionar **Ver → Paleta de comandos** y, en el campo que se abre, buscamos ESLint. a continuación, localizamos la opción **Create `.eslintrc.json` File** y pulsamos sobre ella, se añadirá a los archivos de la carpeta el archivo .eslintrc.json, que permite gestionar el control del código.

