

VSCODE

APUNTES:

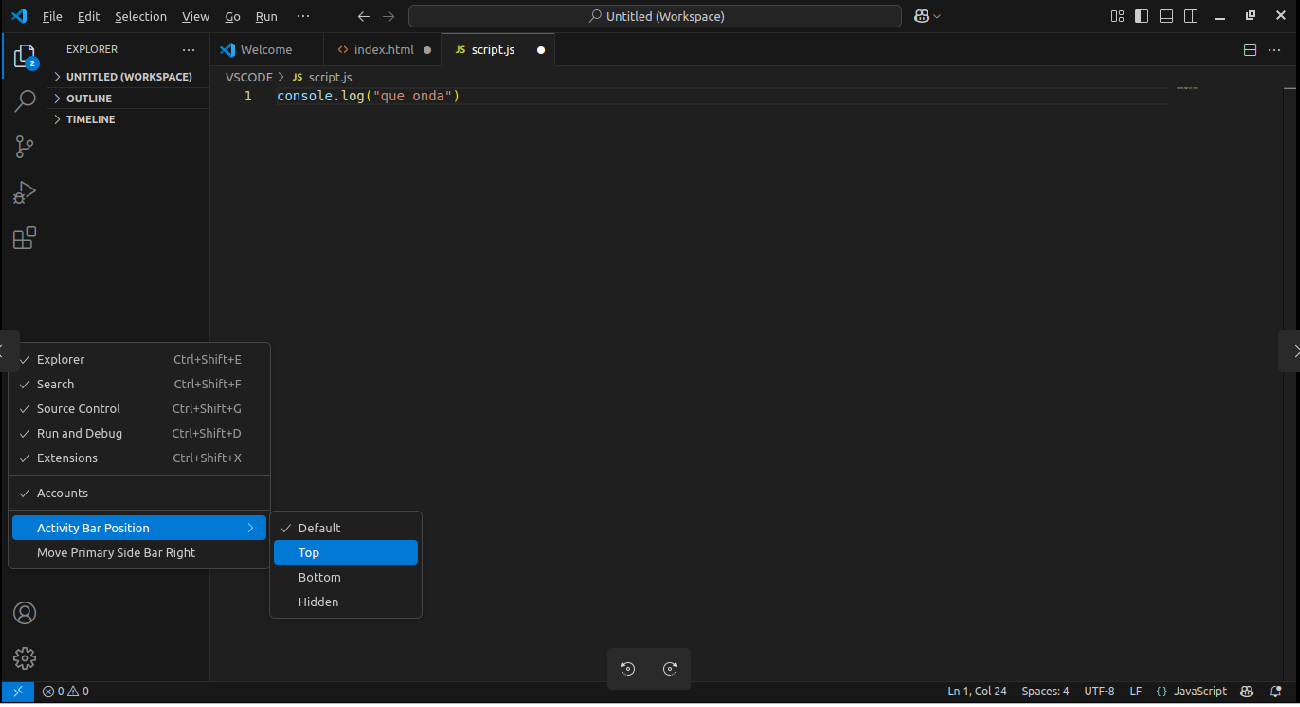
Editor hibrido puede ser un IDE con las extensiones correctas.

CONFIGURACIONES

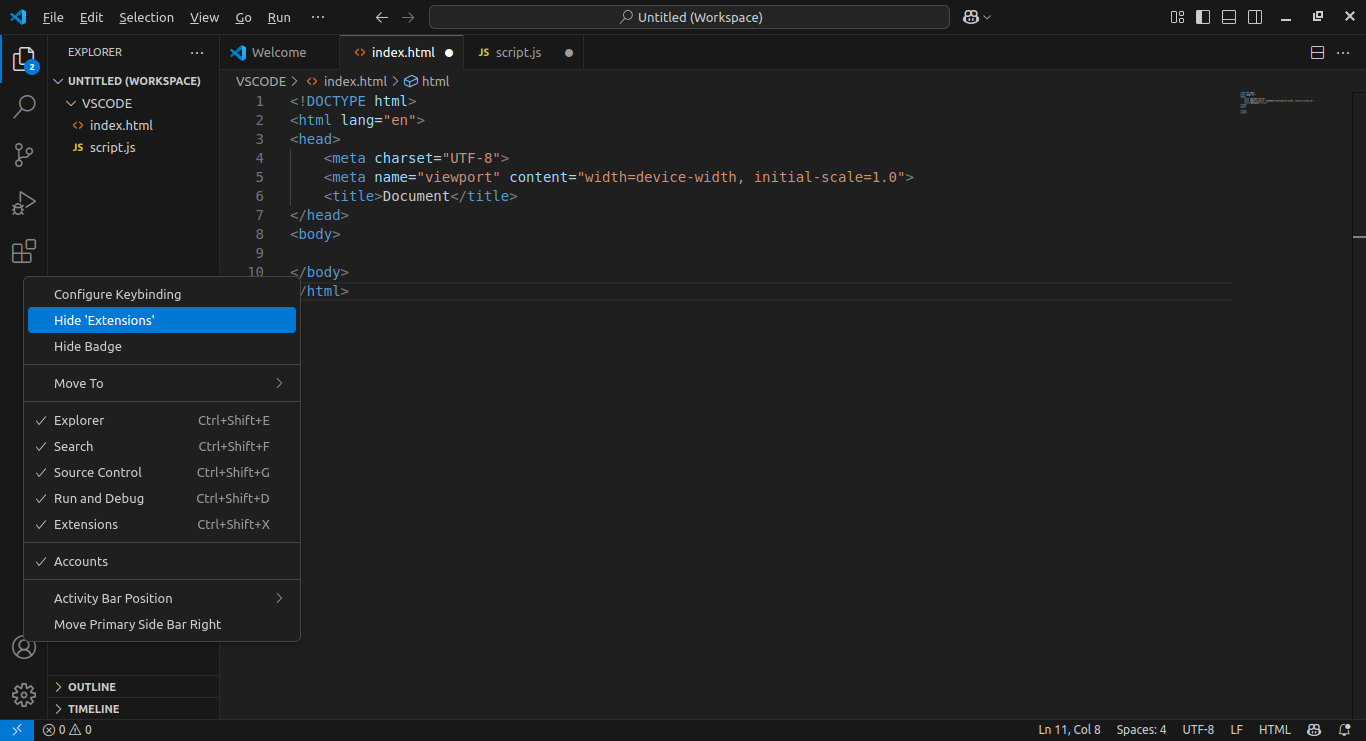
-Restablecer vscode,lo deja como estaba al prinicipio sin ningun cambio: ctrl+p » settings (JSON) opne user » desintalar todas las extennsiones »

APARTADOS:

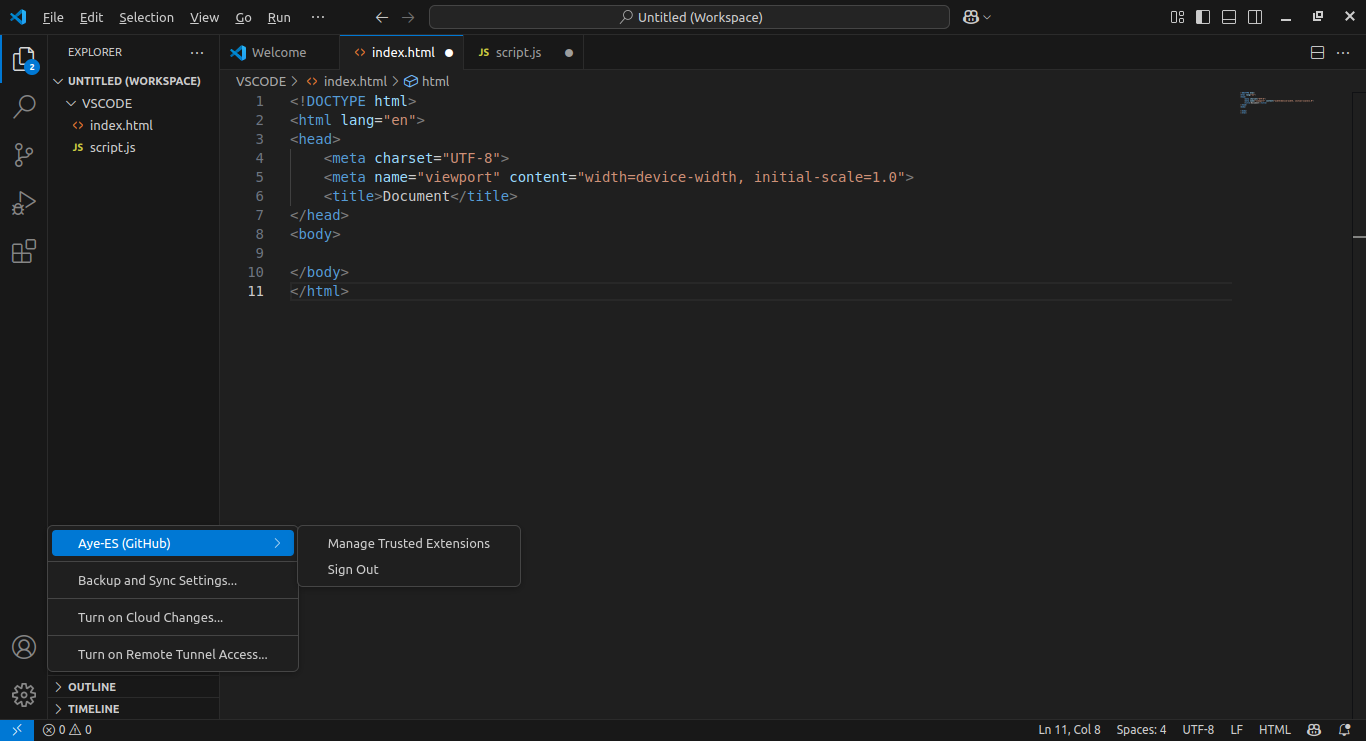
* EXPLORER(AREA)



Podemos cambiar la posicion de la barra de actividad donde querramos.

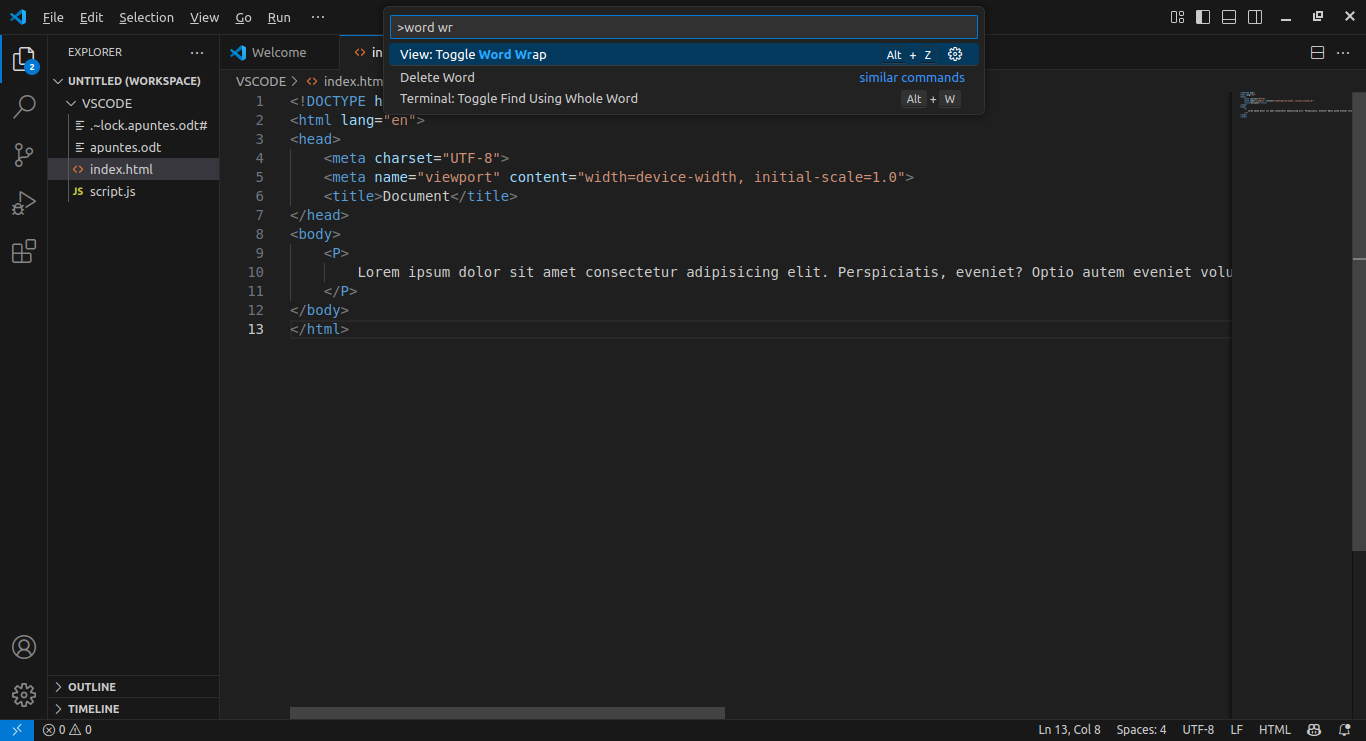


Podemos con hide o destiqueando extensiones, podemos ocultar el icono asi con los demas iconos si se desea ,con la misma mecanica



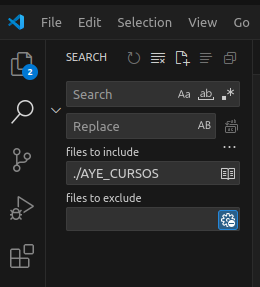
Aqui se inicia sesion con la cuenta de git hub si es pposible para conectar a la cuenta de vscode para mas comodidad y para cuando iniciemos sesion en otro equipo se abra vscode con las mismas configuraciones que nosotros ya hicimos y tenemos en esa cuenta.

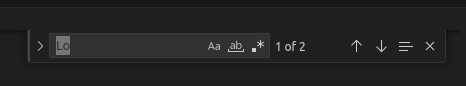
* SEARCH(AREA)



La linea de codigo 10 esta seguida ,pero si quiero que se adapte al tamaño del vscode pues tendria que “ajustar texto” para eso es: ctrl+shift+p » Toogle Word Wrap (Asi automaticamente se ajustara el texto se adapta al tamaño).

La funcionalidad de search,es un panel para escribir texto y buscar palabras de nuestro codigo y asi tambien reemplzarlas por otra , un atajo de teclas es ctrl+f.



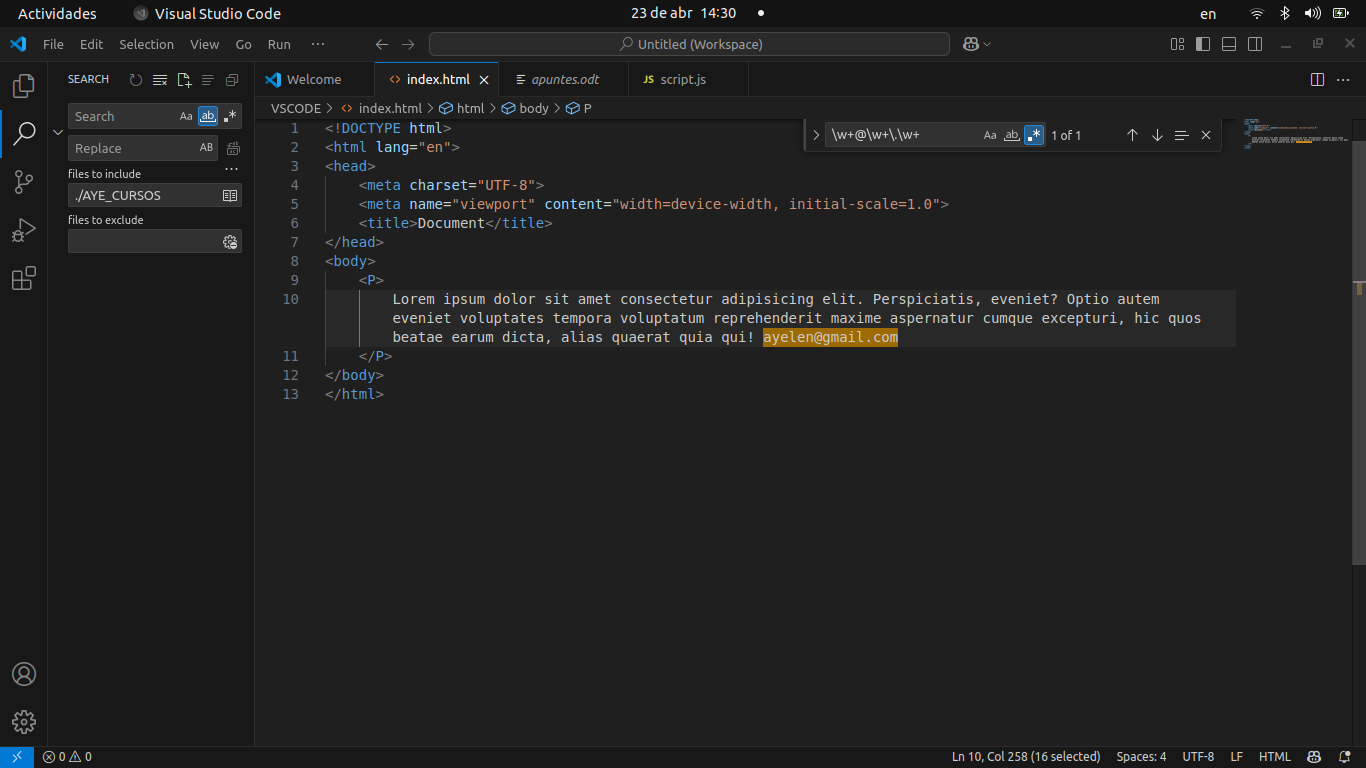


Tiene 3 opciones:

**Match Case(Alt+c).-**Distingue y es sensible de mayusculas y minusculas.

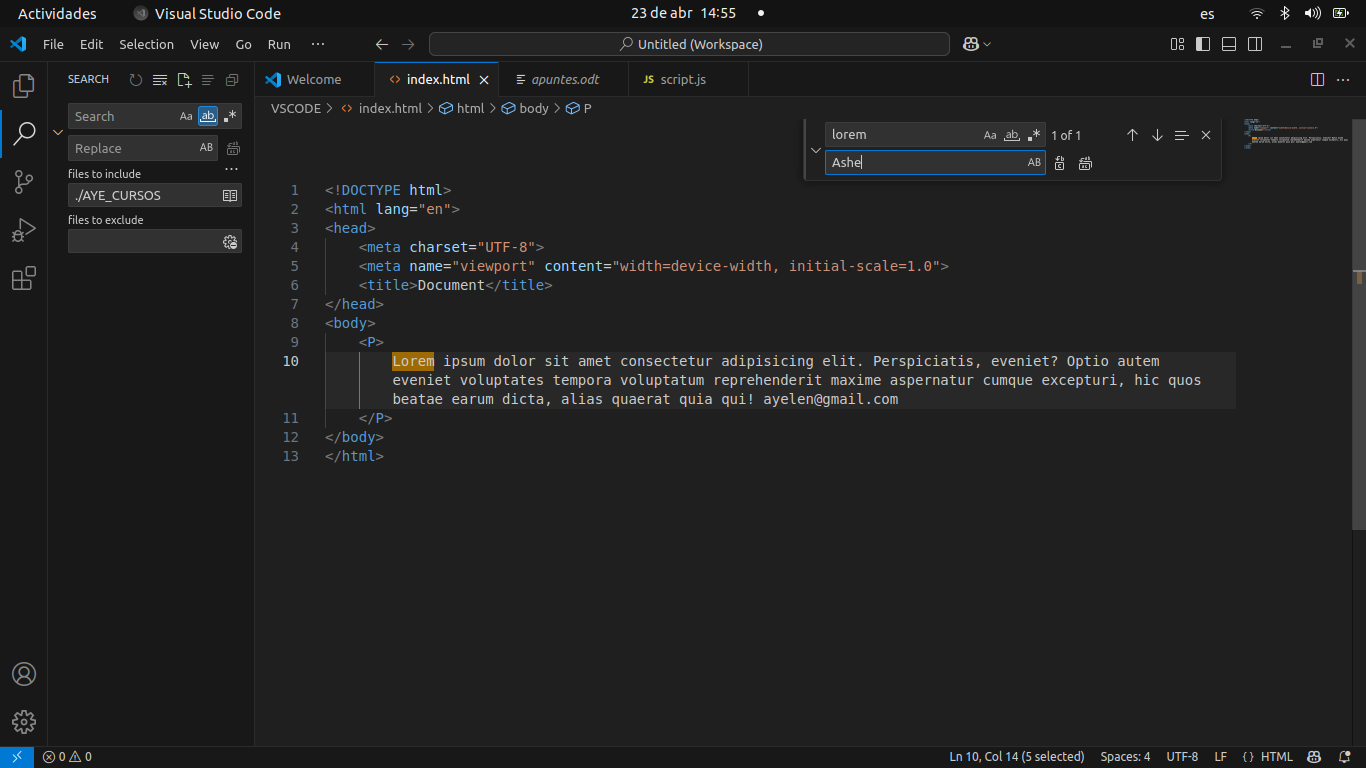
**Match Whole Word(Alt+w).-** Busca palabras especificas completas, no lo busca la apabra adentro de otra palabra.

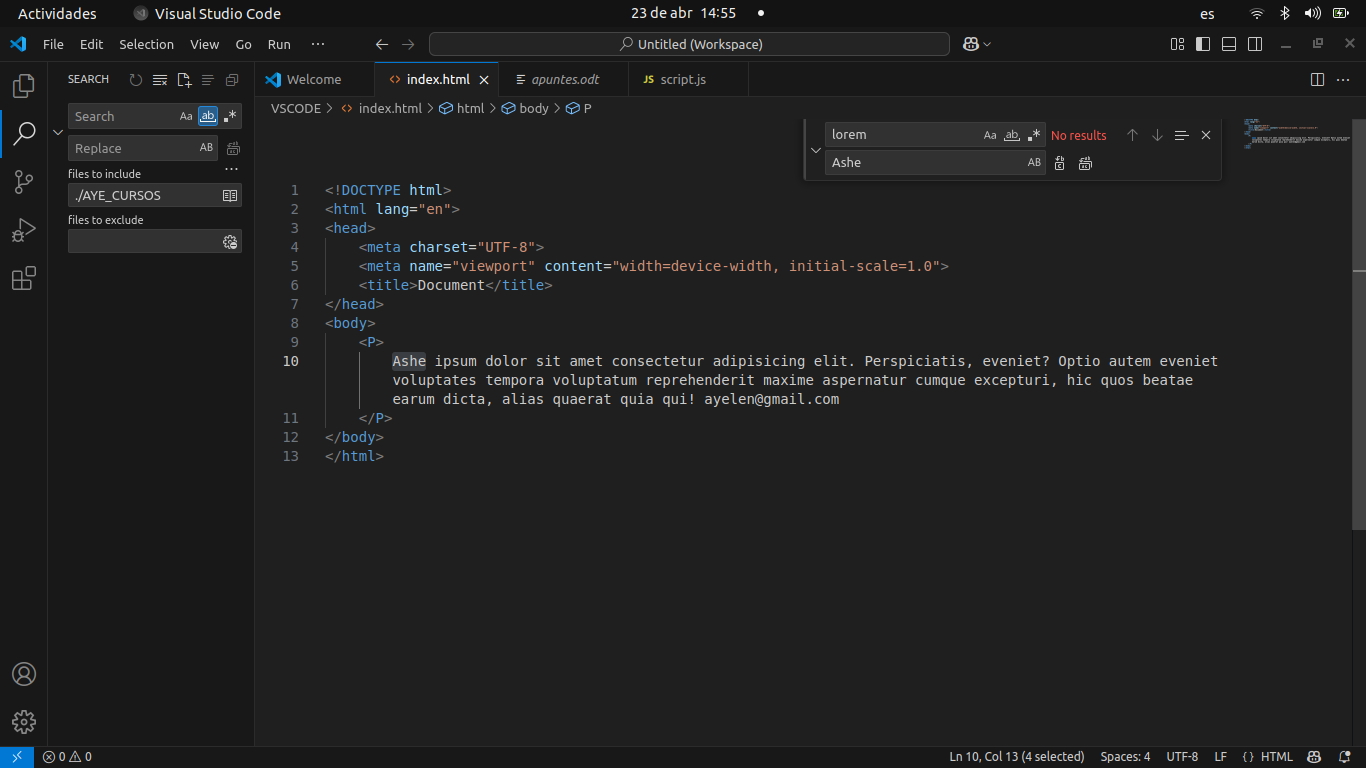
**Use Regular Expression(Alt+r).- Las expresiones regulares son patrones que se utilizan para hacer coincidir combinaciones de caracteres en cadenas.** Esta tercer opcion, permite encontrar cosas en un texto por ejemplo si quisiera encontrar un email, seria indicandole en el buscador que es lo que debe buscar y hallar en el texto, asi: \w+@\w+\.\w+ y hacer click en la 3 opcion de search para que encuentere el email.



O decir que busque esta expresion regular: **https?://\S+”** , para encontrar enlaces y asi sucesivamente usar las expresiones regulares para encontrar texto o varias cosas en un texto aun mas grande.

**Replace: Ctrl + h.-** Reemplaza una palabra por otra.Ahora si se tiene que reemplazar mas de una misma palabra por otra con el atajo ctrl+alt+enter se reemplazara automaticamnte todas las palabras a reemplazar sin tener que ir reemplzando una por una.



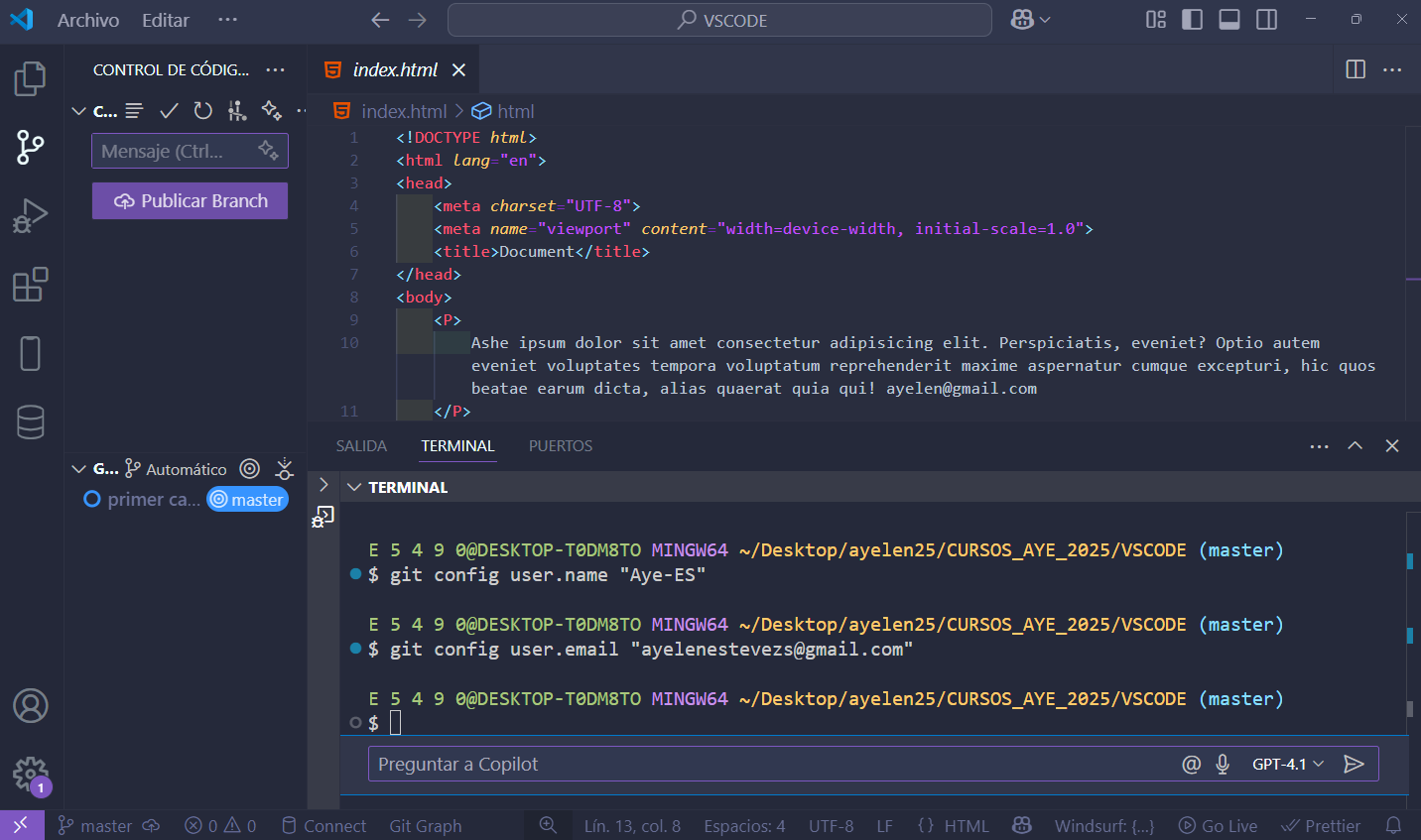




* SOURCE CONTROL (AREA)-“GIT”

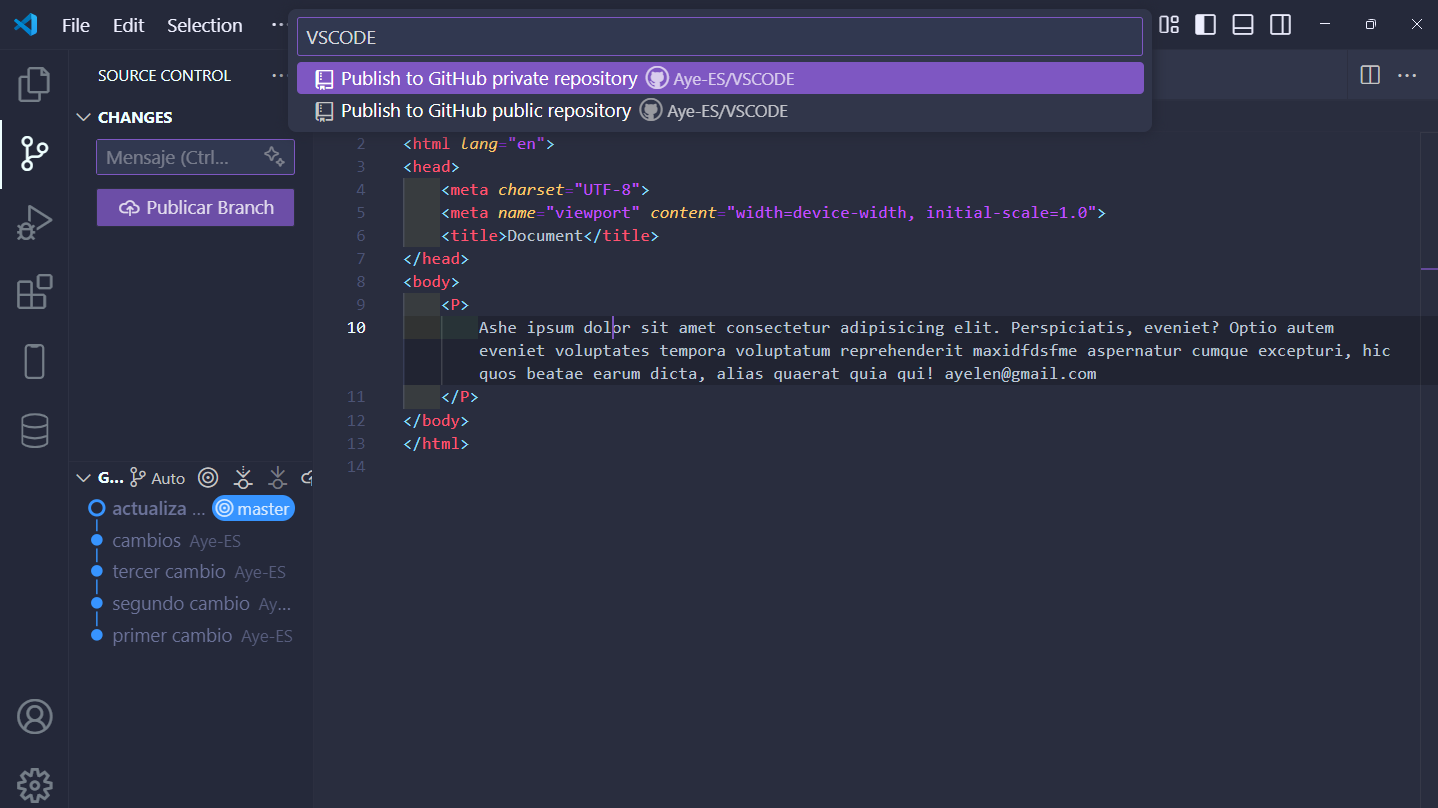
Antes siempre poner la configuración del usuario y contraseña de la cuenta de GitHub que se está utilizando (esto desde terminal con los comandos de “git config user.name” y “git config user.email”)

Este espacio de trabajo en VCODE es cuando conectamos **git con Vscode** para subir archivos o proyectos a un repositorio y realizar cambios, etc. Se crea el repositorio y se guardan los cambios con un “commits”



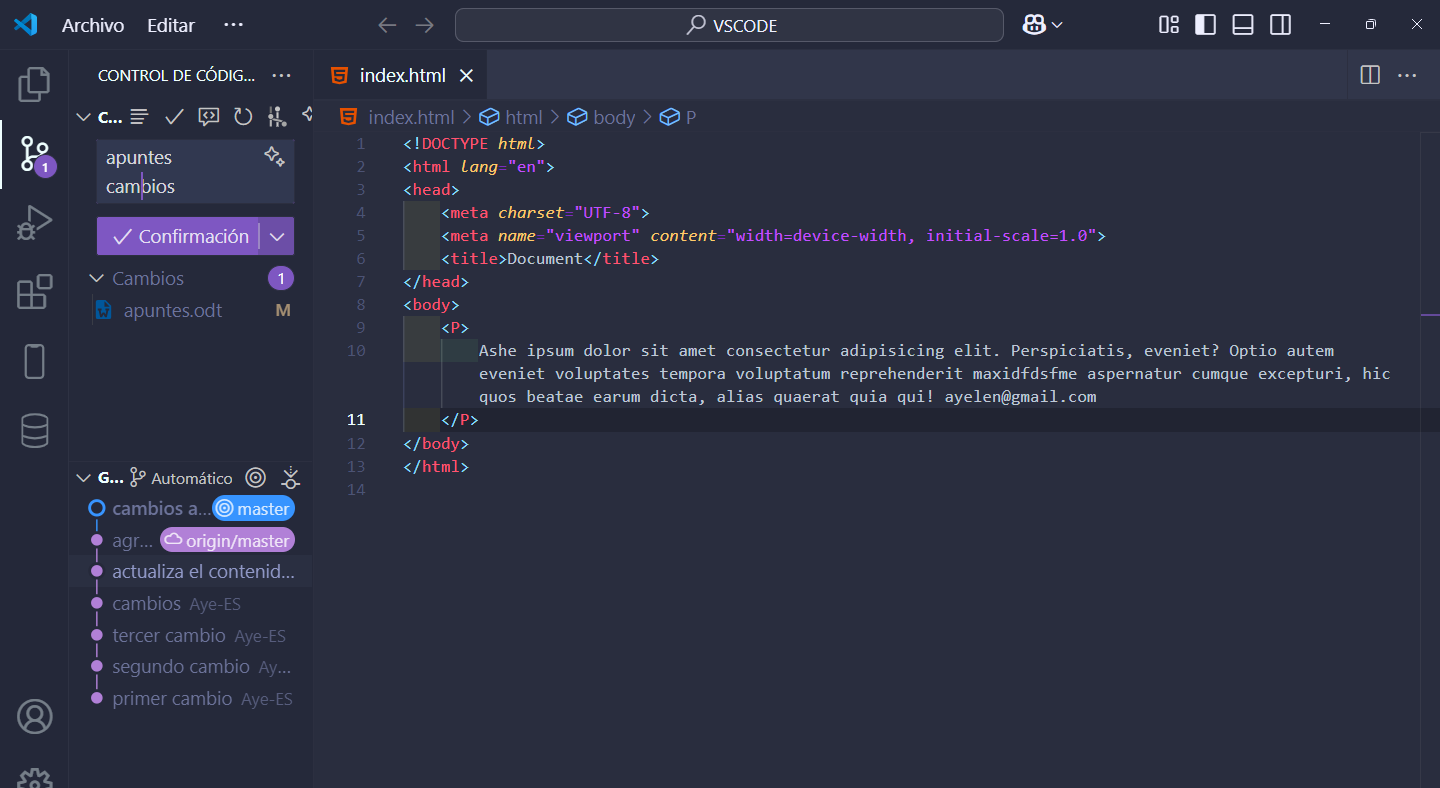
En el circulo encerrado es donde se ponen las descripciones para confirmar los commits y otras acciones más como visualizar los cambios en git graph.

Aparte se puede publicar el proyecto a un repositorio remoto ya sea privado o público desde Vscode(siempre y cuando conectemos nuestra cuenta de GitHub a Vscode ):

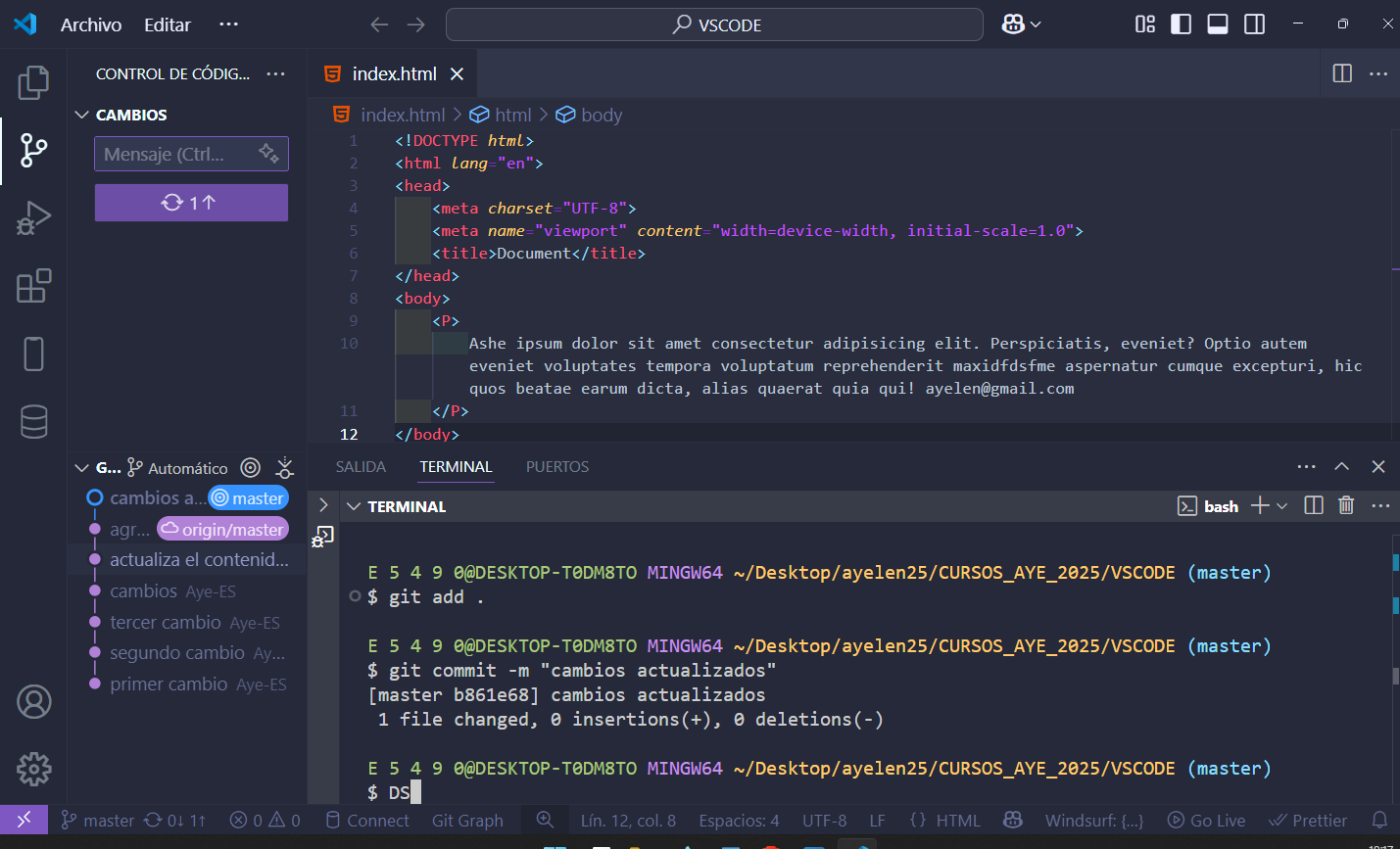


Podemos realizar los cambios desde la interfaz de vcode o desde terminal(bash)

Desde interfaz vcode:

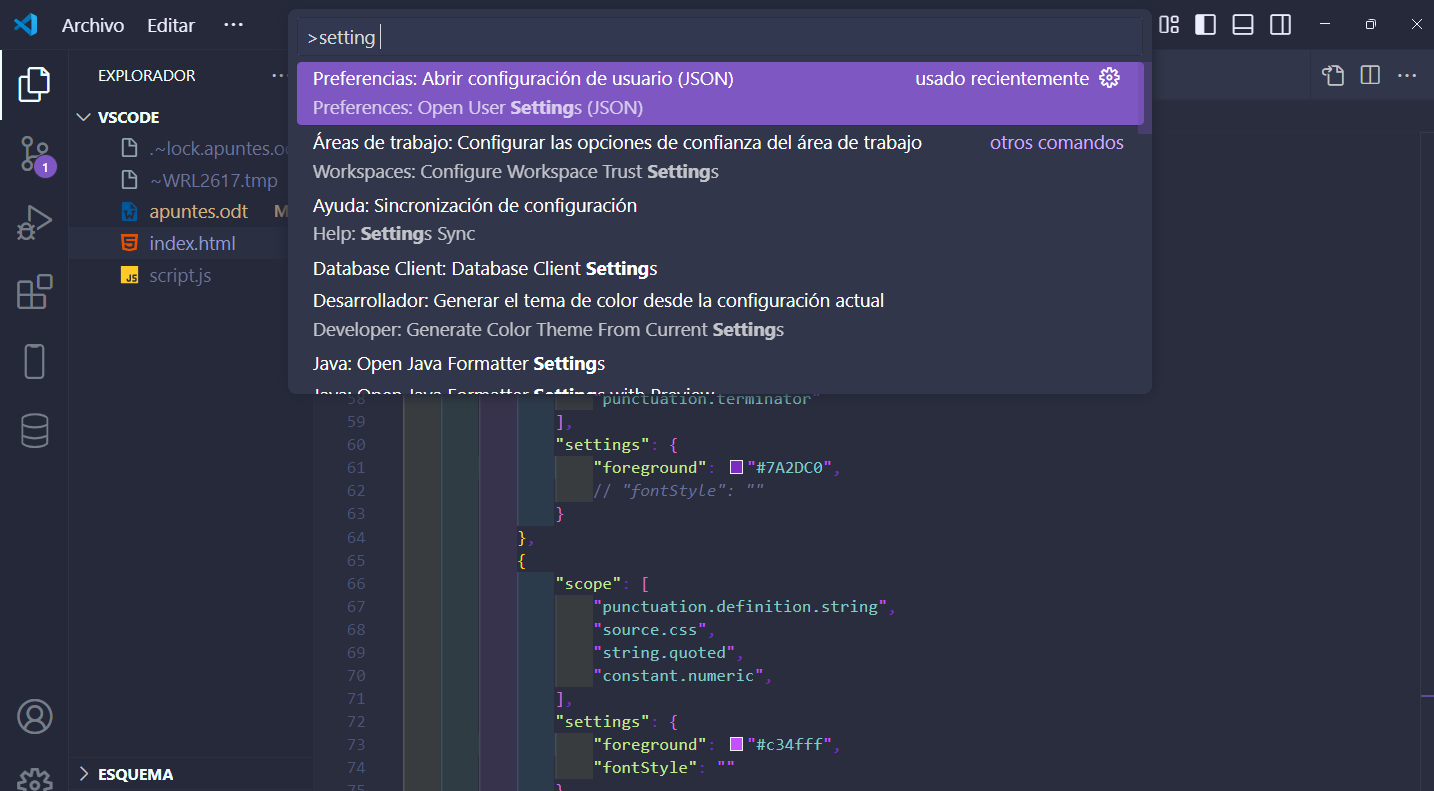


Desde terminal (bash):

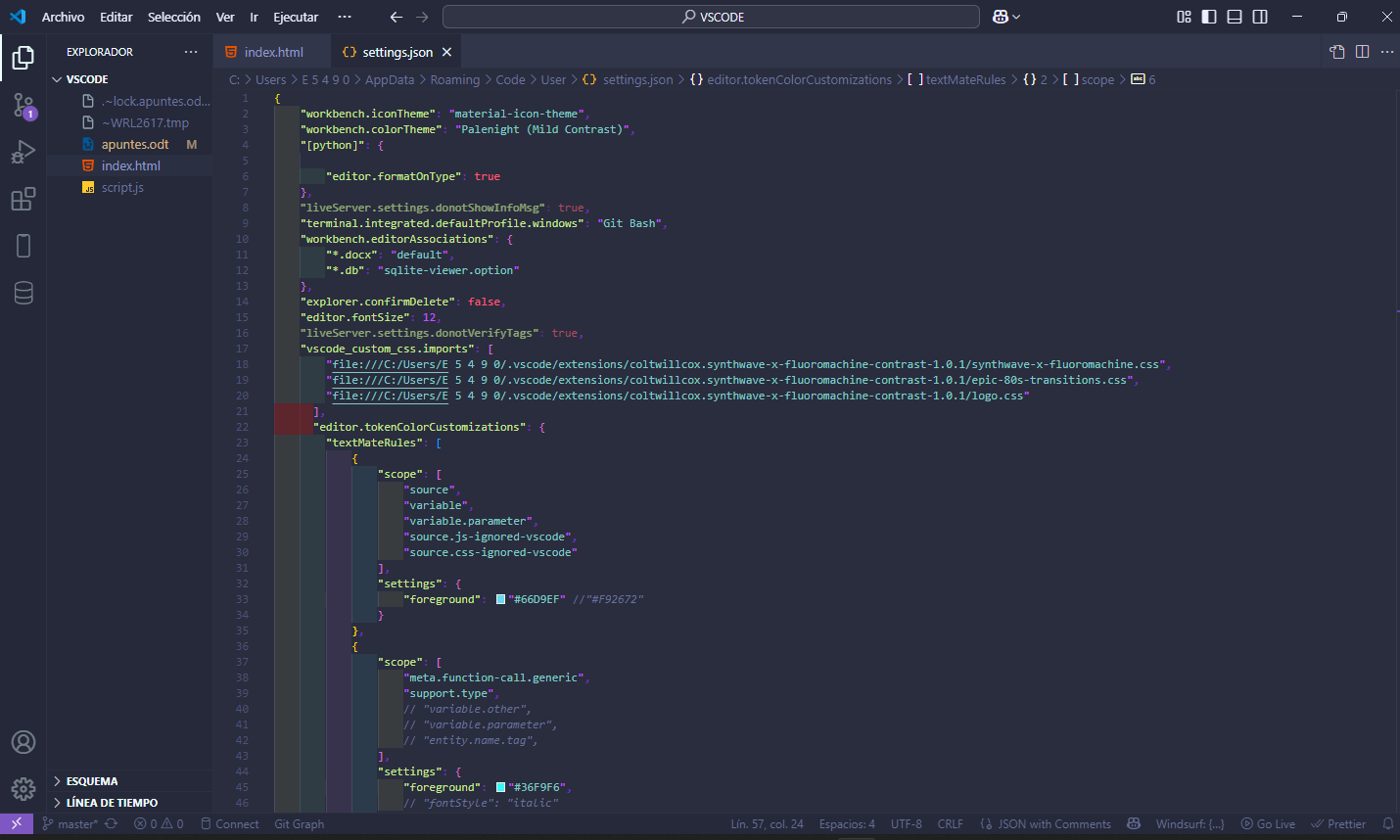


Y por último este es el lugar donde podemos ir agregando manualmente si se quiere “configuraciones relacionadas con VSCODE”, SE TYPEA OPEN USER SETTINGS JSON:

Atajo usado:Ctrl+shift+p



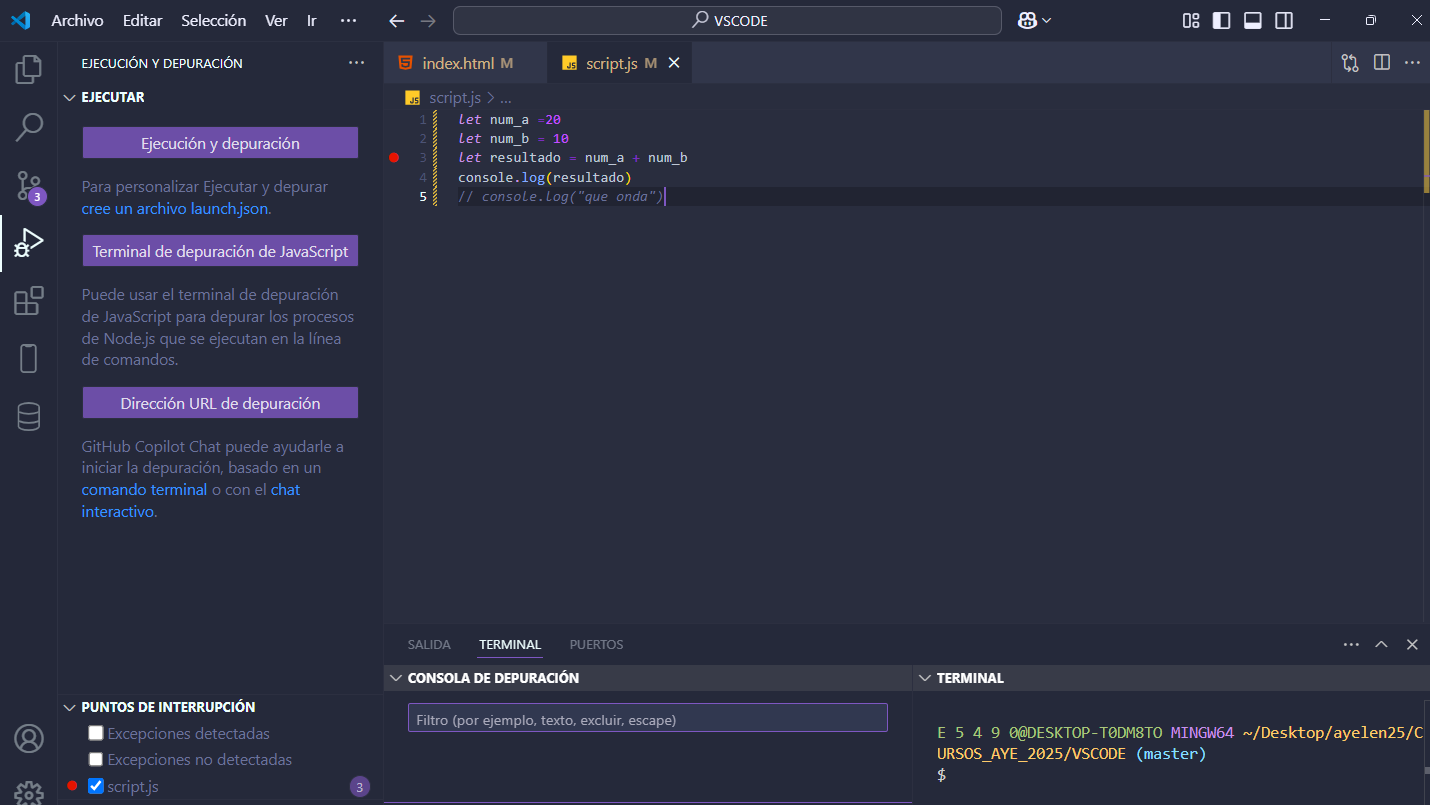
Aquí se muestran todas las configuraciones que se tiene por el momento:



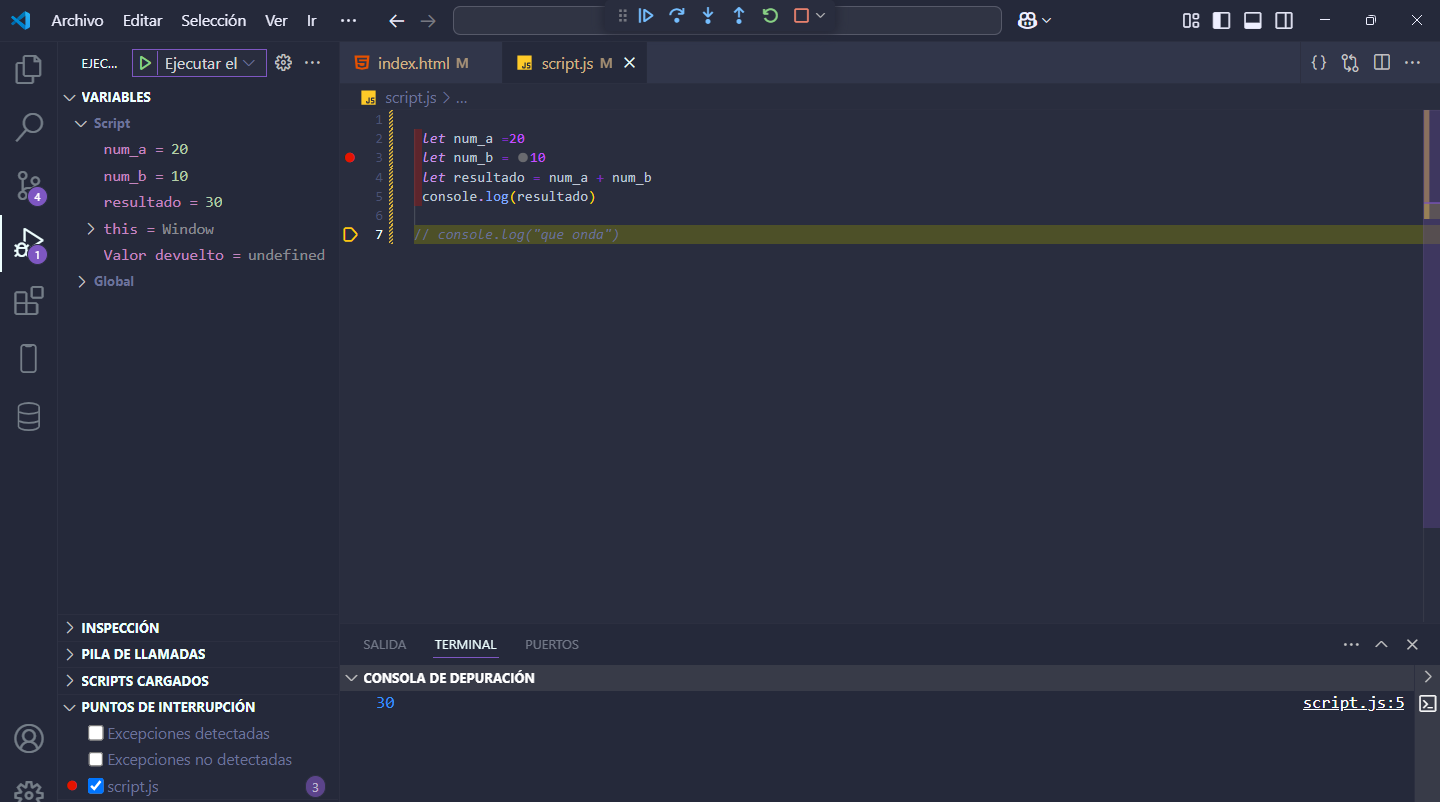
* RUN AND DEBUG (AREA)

Es una herramienta que nos permite depurar(Depurar es buscar y corregir errores en un programa.)

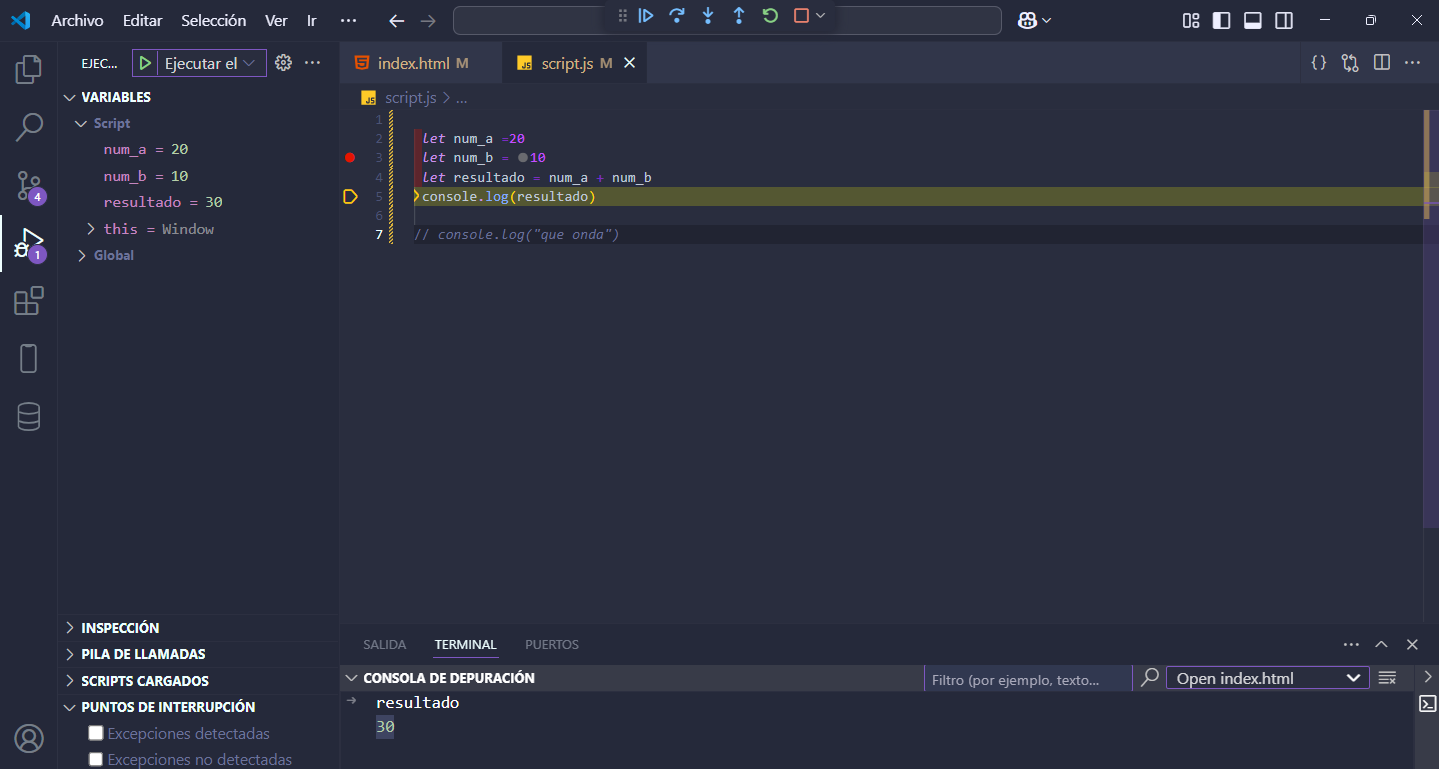
Tyepo “ejecutar y depurar pero que este seleccionado el archivo “archivo.html” ya que se necesita realizar unas configuraciones extas para depurarlo desde un “archivo.js”



Aquí ya empieza a ir línea a línea del código :

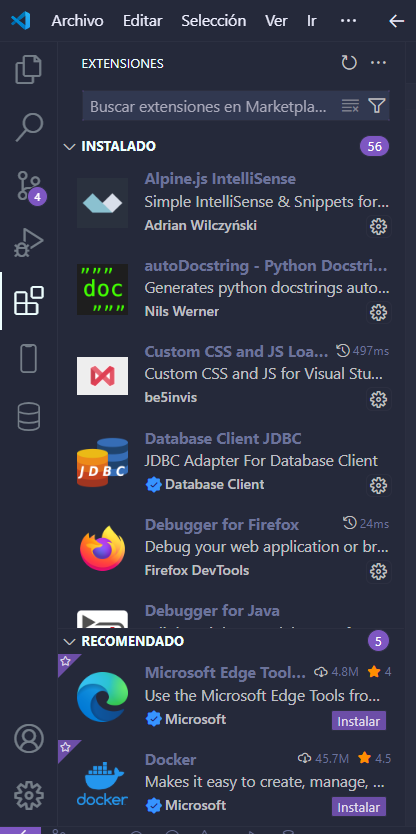


Ese punto rojo para poner el límite de hasta dónde queremos nosotros que se depure el código. Y las flechas de abajo ya son para seguir depurando línea por línea nosotros mismos y las flechas hacia adelante es para salir de la depuración.



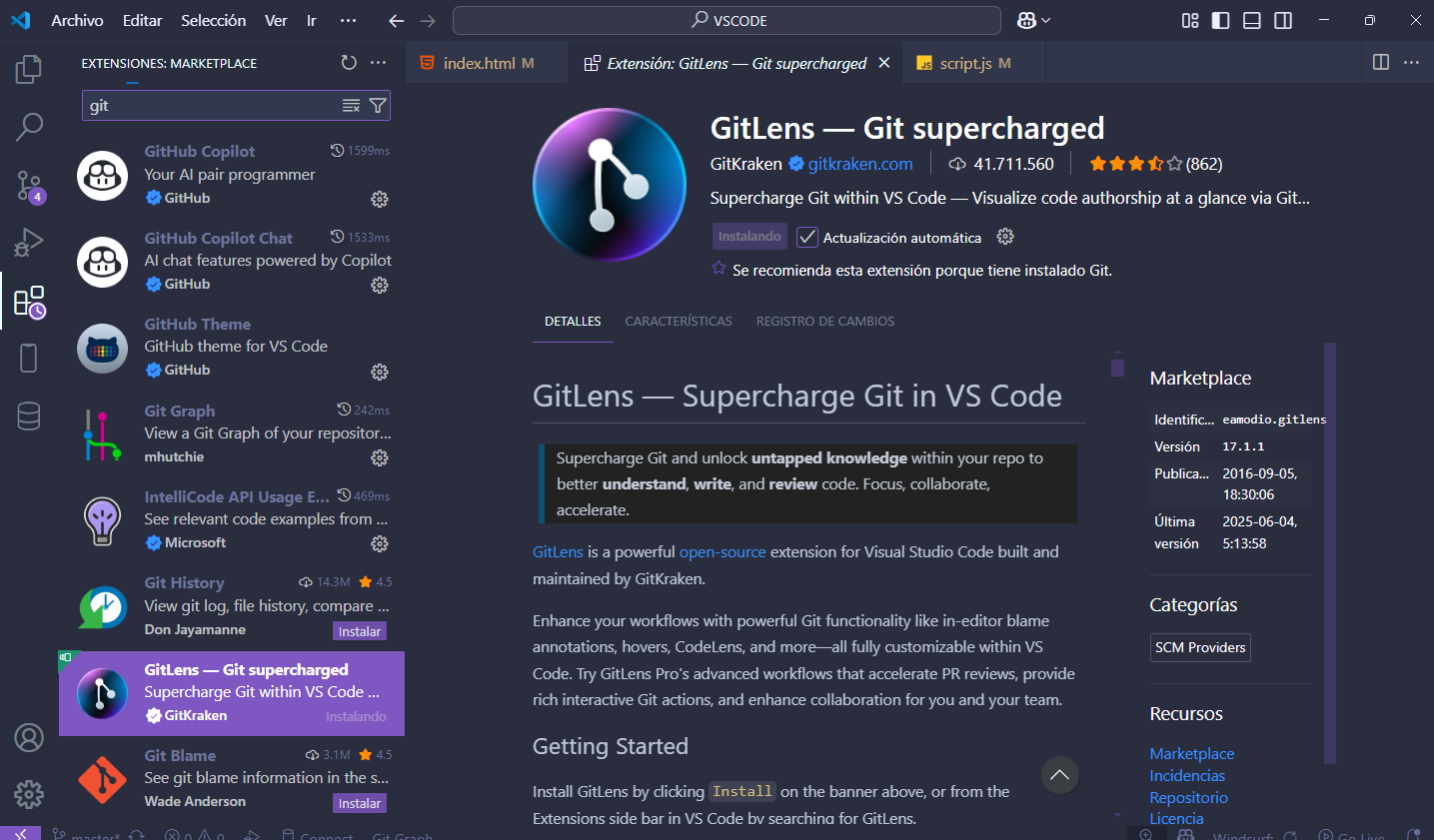
Y en esta foto ya se muestra el resultado final del código.

Entonces este apartado de Vscode nos sirve para entender que es lo que pasa y cuales son los valores de las variables en determinado momento del programa

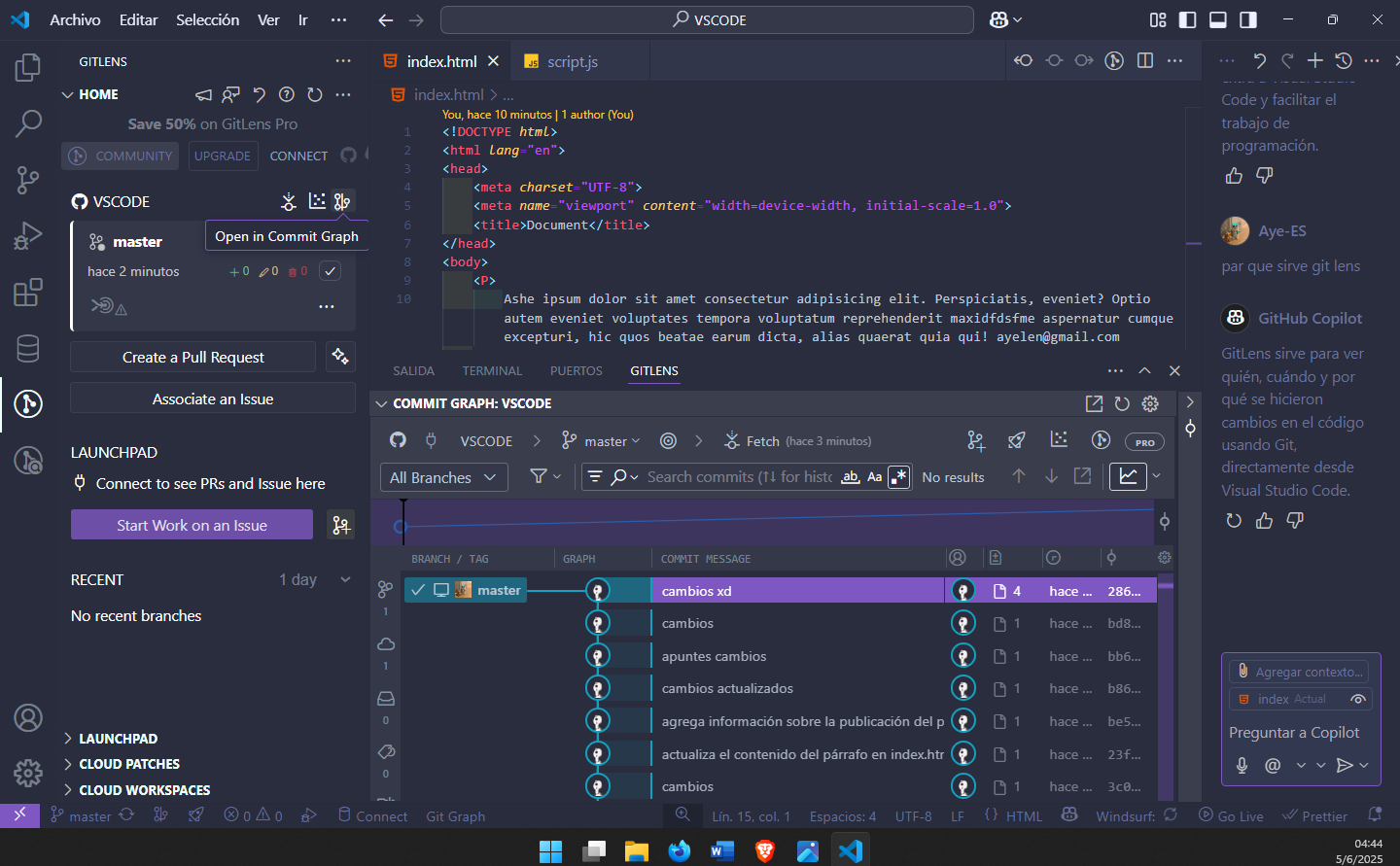
* EXTENSIONS (AREA)

Las extensiones sirven para agregar funciones extra a Visual Studio Code y facilitar el trabajo de programación.

Bueno a continuación se instalará la extensión de GitLens que sirve para ver quién, cuándo y por qué se hicieron cambios en el código usando Git, directamente desde Visual Studio Code.



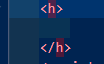
Y aquí podemos visualizar igualmente el grafico de commits de nuestro proyecto, es buenísimo.



Y así podemos ir descargando otras extensiones que nos sean útiles para nosotros.

TIPS: AHORA PARA EDITAR AMBAS ETIQUETAS JUNTAS EN HTML SOLO SE A LA VONFIGURACION DE Ctrl +shift +p > settings.json > "editor.linkedEditing": true,

Y ahora con este tip, si yo borro la etiqueta de apertura igualmente se borrará la etiqueta de cierre en los tags de html. Me ahorra tiempo.



HASTA AQUÍ ELENTENDIMIENTO DE LA BARRA DE ACTIVIDADES.

* **AHORA SE VERA COMO FUNCIONA LOS AJUSTES PARA QUE SE PUEDA CAMBIAR CUALQUIER COSA EN VSCODE.**
* USER SETTINGS

Pensaremos en objetos, a estos objetos yo puedo acceder a sus propiedades y modificarlo desde User settings json.

>Entro con el atajo crtl+shift+p: Con esto voy a acceder a la paleta de comandos

> User settings json: El archivo \*\*User settings.json\*\* sirve para personalizar y configurar el comportamiento y la apariencia de Visual Studio Code según tus preferencias. Aquí puedes ajustar temas, atajos, fuentes, colores, extensiones y muchas otras opciones para que el editor se adapte a tu forma de trabajar.

Entonces es el archivo de configuración que permite realizar todas las configuraciones que Vscode no tiene, asi mismo podemos acceder a las extensiones y modificarlas.

**AQUÍ SE MUESTRAN ALGUNAS CONFIGURACIONES QUE SE REALIZARON:**

{

    "workbench.iconTheme": "material-icon-theme",

    "workbench.colorTheme": "Tomorrow Night Blue",

    "[python]": {

        "editor.formatOnType": true

    },

    "liveServer.settings.donotShowInfoMsg": true,

    "terminal.integrated.defaultProfile.windows": "Git Bash",

    "workbench.editorAssociations": {

        "\*.docx": "default",

        "\*.db": "sqlite-viewer.option"

    },

    "explorer.confirmDelete": false,

    "editor.fontSize": 12,

    "liveServer.settings.donotVerifyTags": true,

    "vscode\_custom\_css.imports": [

        "file:///C:/Users/E 5 4 9 0/.vscode/extensions/coltwillcox.synthwave-x-fluoromachine-contrast-1.0.1/synthwave-x-fluoromachine.css",

        "file:///C:/Users/E 5 4 9 0/.vscode/extensions/coltwillcox.synthwave-x-fluoromachine-contrast-1.0.1/epic-80s-transitions.css",

        "file:///C:/Users/E 5 4 9 0/.vscode/extensions/coltwillcox.synthwave-x-fluoromachine-contrast-1.0.1/logo.css"

    ],

    "editor.tokenColorCustomizations": {

        "textMateRules": [

            {

                "scope": [

                    "source",

                    "variable",

                    "variable.parameter",

                    "source.js-ignored-vscode",

                    "source.css-ignored-vscode"

                ],

                "settings": {

                    "foreground": "#66D9EF" *//"#F92672"*

                }

            },

            {

                "scope": [

                    "meta.function-call.generic",

                    "support.type",

*// "variable.other",*

*// "variable.parameter",*

*// "entity.name.tag",*

                ],

                "settings": {

                    "foreground": "#36F9F6",

*// "fontStyle": "italic"*

                }

            },

            {

                "scope": [

                    "constant.character.escape",

                    "variable.language",

                    "keyword.operator.assignment",

                    "keyword.operator.arithmetic",

                    "entity.name",

                    "support.other",

                    "punctuation.separator",

                    "punctuation.terminator"

                ],

                "settings": {

                    "foreground": "#7A2DC0",

*// "fontStyle": ""*

                }

            },

            {

                "scope": [

                    "punctuation.definition.string",

                    "source.css",

                    "string.quoted",

                    "constant.numeric",

                ],

                "settings": {

                    "foreground": "#c34fff",

                    "fontStyle": ""

                }

            },

            {

                "scope": [

*// the following elements will be in italic*

*// "source.java",*

                    "comment",

                    "storage.modifier", *// static keyword*

                    "storage.type.php", *// typehints in methods keyword*

                    "keyword.other.new.php", *// new*

                    "entity.other.attribute-name", *// html attributes*

                    "fenced\_code.block.language.markdown", *// markdown language modifier*

                    "keyword.control",

                    "keyword.operator.logical",

                    "storage.type", *//class keyword*

                    "entity.other.attribute-name",

                    "entity.name.method",

                    "keyword.control.import.ts",

                    "keyword.control.import.tsx",

                    "keyword.control.import.js",

                    "keyword.control.flow.js",

                    "keyword.control.from.js",

                    "keyword.control.from.ts",

                    "keyword.control.from.tsx"

                ],

                "settings": {

                    "fontStyle": "italic"

                }

            },

            {

                "scope": [

*// the following elements will be displayed in bold*

                    "entity.name.type.class" *// class names*

                ],

                "settings": {

                    "fontStyle": "bold"

                }

            },

            {

                "scope": [

*// the following elements will be displayed in bold and italic*

                    "entity.name.section.markdown" *// markdown headlines*

                ],

                "settings": {

                    "fontStyle": ""

                }

            },

        ]

    },

    "explorer.confirmDragAndDrop": false,

    "workbench.editor.empty.hint": "hidden",

    "[html]": {

        "editor.defaultFormatter": "esbenp.prettier-vscode"

    },

    "codeium.enableConfig": {

        "\*": true,

        "assembly": true

    },

    "editor.minimap.size": "fit",

    "editor.minimap.enabled": false,

    "git.enableSmartCommit": true,

"editor.cursorBlinking": "expand",

"editor.cursorStyle": "line",

"editor.linkedEditing": true,

"workbench.editor.enablePreview": false,

"files.autoSave": "afterDelay", *//configurar el autosave para que se guarde cada cierto tiempo programado*

"files.autoSaveDelay": 1000,*//El tiempo en el que se guardara el o los archivos es de 1000 milisegundos (1 segundo)*

"files.exclude": {

    "\*\*/.git": false,*//false para que no se oculten los archivos de git y  true para que se oculten los archivos de git*

    "\*\*/.svn": true,

    "\*\*/.hg": true,

    "\*\*/.DS\_Store": true,

    "\*\*/Thumbs.db": true

},

}

* ATAJOS DE TECLADO(Shortcuts)

CRTL +S = Guardar archivo(s)

CRTL +O =Abrir un archivo

CRTL +SHIFT+P =Abrir la paleta de comandos

CRTL +SHIFT+N =Abrir una nueva ventana o instancia de Vscode

CRTL +W =Cerrar pestañas abiertas en Vscode

CRTL +K+C= Comentar línea de código

SHIFT+ALT= Para copiar la misma línea de código ya sea arriba o debajo de una forma más rápida

ALT (tenerlo presionado ya sea con la flecha para \*arriba o abajo\*)= Para mover esa línea de código donde yo quiera, según mantenga presionado con la flecha de arriba para mover hacia arriba o abajo para moverlo hace abajo

CRTL +, =Nos abre una ventana de configuración de Vscode (lo mismo que la paleta de comandos, pero aquí ya muestra todo lo que esta para configurar más sus selecciones múltiples para que el usuario elija que desea modificar de los objetos y así también sus propiedades)

CRTL +ALT+ flecha de abajo=Para identar un grupo o conjunto de líneas de código una debajo de la otra, de una manera más rápida. (Identar código es organizarlo usando espacios o tabulaciones al inicio de cada línea para que sea más claro y fácil de leer, mostrando la estructura y jerarquía del programa.): Con el atajo este, se crea múltiples cursores y por ende puedo escribir, tabular, borrar, etc. en las líneas seleccionadas al mismo tiempo

ALT+ click del mouse en donde queremos que se cree el cursor=Para el mismo caso de arriba, solo que esto nos sirve para cuando queremos seleccionar líneas de código que no estén juntas o una debajo de otra, sino que estén separadas por unas cuantas líneas de código ya sea por arriba o debajo.

CRTL +F=Buscar una palabra

CRTL +H=Reemplazar la palabra buscada

“Seleccionar la palabra a reemplazar” + CRTL +SHIFT+L=Reemplaza de una forma más rápida “una palabra” que está en diferentes líneas de código.

CRTL +SHIFT+ [ = Para plegar etiquetas de html o bloques de código.

CRTL +SHIFT+] = Para desplegar etiquetas de html o bloques de código.

CRTL+K(suelto)+Z= Para poner Vscode en “ZENMODE” (obviamente esto puedo configurar como quiero que me aparezca el zenmode a mí(personalizarlo), en settings. json.

CRTL +SHIFT+ M= Para abrir el editor de problemas, donde se ven los errores que se tiene en el código.

* SNIPPETS

En Visual Studio Code, un \*\*snippet\*\* es un fragmento de código predefinido que puedes insertar rápidamente en tus archivos para ahorrar tiempo y escribir código más rápido. Puedes usar snippets ya incluidos o crear los tuyos personalizados.

ESTE APARTADO VERLO E INVESTIGAR DESPUES DE APRENDER Y TENER MAS PRACTICA EN CODIFICAR EN LENGUAJES Y FRAMEWORK, YA QUE LOS SNIPPETS SON DE MUCHA AYUDA.

* EMMET

\*\*Emmet\*\* es una herramienta integrada en Visual Studio Code que permite escribir abreviaturas cortas y expandirlas automáticamente en código HTML, CSS y otros lenguajes, ahorrando tiempo y facilitando la escritura de código repetitivo.

Aquí tienes un ejemplo sencillo de un snippet usando Emmet en HTML:

Si escribes esto en un archivo HTML en VS Code:

```

ul>li\*3

```

y presionas \*\*Tab\*\*, Emmet lo expande automáticamente a:

```html

<ul>

<li></li>

<li></li>

<li></li>

</ul>

```

Esto te ahorra tiempo al crear listas u otras estructuras repetitivas.

Otro ejemplo:

Si pongo esto: ul>li\*3, le doy Enter entonces pasa esto:

    <ul>

        <li></li>

        <li></li>

        <li></li>

    </ul>

Otro ejemplo:

ul>li\*3>a

queda asi:

    <ul>

        <li><a *href*=""></a></li>

        <li><a *href*=""></a></li>

        <li><a *href*=""></a></li>

    </ul>

Otro ejemplo

header.header>nav.header-nav>ul.header-el>li.list-item\*5>a.link-item{Enlace $}

Queda asi:

    <header *class*="header">

        <nav *class*="header-nav">

            <ul *class*="header-el">

                <li *class*="list-item"><a *href*="" *class*="link-item">Enlace 1</a></li>

                <li *class*="list-item"><a *href*="" *class*="link-item">Enlace 2</a></li>

                <li *class*="list-item"><a *href*="" *class*="link-item">Enlace 3</a></li>

                <li *class*="list-item"><a *href*="" *class*="link-item">Enlace 4</a></li>

                <li *class*="list-item"><a *href*="" *class*="link-item">Enlace 5</a></li>

            </ul>

        </nav>

    </header>

Aun se puede realizar más emmets, investigar más sobre este apartado.

AHORA DE ESTO , TAMBIEN SE PUEDE CREAR UN SNIPPET PARA HACERLO MUCHO MAS CORTO.

// Ejemplo de snippet para menú header mostrado anteriormente.

"Menu Header": {

"prefix": "menuheader",

"body": [

"<header class=\"header\">",

" <nav class=\"header-nav\">",

" <ul class=\"header-el\">",

" <li class=\"list-item\"><a class=\"link-item\" href=\"\">Enlace 1</a></li>",

" <li class=\"list-item\"><a class=\"link-item\" href=\"\">Enlace 2</a></li>",

" <li class=\"list-item\"><a class=\"link-item\" href=\"\">Enlace 3</a></li>",

" <li class=\"list-item\"><a class=\"link-item\" href=\"\">Enlace 4</a></li>",

" <li class=\"list-item\"><a class=\"link-item\" href=\"\">Enlace 5</a></li>",

" </ul>",

" </nav>",

"</header>"

],

"description": "Estructura de menú header con 5 enlaces"

}

PRACTICAR ESTO Y AVERIGUAR MAS PORQUE AYUDA A LA PRODUCTIVIDADY ES DINAMICO.