**Proceso instalación Visual Studio Code**

**Paso 1:** Escribimos en Google Visual Studio Code y seleccionamos donde dice “Download”.



**Paso 2:** Seleccionamos el sistema operativo que tenemos y lo descargamos.



**Paso 3:** Al darle clic nos descargará un .exe, al cual le daremos clic encima.



**Paso 4:** Lee y acepta el acuerdo de licencia. Haz clic en Next para continuar.



**Paso 5:** Puedes cambiar la ubicación de la carpeta de instalación o mantener la configuración predeterminada. Haz clic en Next para continuar.



**Paso 6:** Elige si deseas cambiar el nombre de la carpeta de accesos directos en el menú Inicio o si no deseas instalar accesos directos en absoluto. Haz clic en Next.



**Paso 7:** Selecciona las tareas adicionales, por ej. crear un icono en el escritorio o añadir opciones al menú contextual de Windows Explorer. Haz clic en Next.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

**Paso 8:** Haz clic en Install para iniciar la instalación.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

**Paso 9:** El programa está instalado y listo para usar. Haz clic en Finish para finalizar la instalación y lanzar el programa.



**Creación de carpetas**

Creamos una carpeta raíz. En nuestro caso la llamaremos “taskApp”.

Captura de pantalla de un celular

Descripción generada automáticamente

Dentro de esta, crearemos dos archivos. Uno llamado index.html y el segundo llamado app.js (Es importante tener exactamente esa extensión en el archivo).

Captura de pantalla de un celular

Descripción generada automáticamente

**Task App – Project**

En el archivo index.html empezaremos creando el html raíz.



Ahora, vamos a Google para buscar “getbootstrap”.

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

Lo buscamos, ahora le daremos al primer resultado que nos arroja Google.

Captura de pantalla de un celular

Descripción generada automáticamente

Dentro de esta página, procederemos a darle donde dice “Docs” en la barra de navegación de esta.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Ahora podremos ver toda la documentación de bootstrap5, lo único que tendremos que hacer en estos momentos es bajar un poco hasta encontrar lo siguiente:

Interfaz de usuario gráfica, Texto, Aplicación, Teams

Descripción generada automáticamente

Para terminar, lo único que tendremos que hacer es copiar la línea de código que está debajo de “title”, exactamente esta:

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

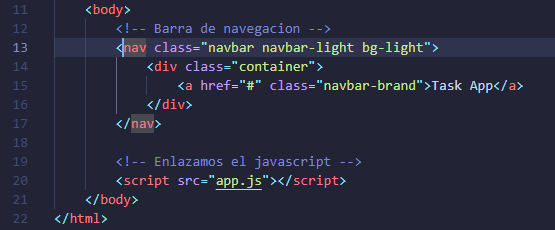
Descripción generada automáticamente

Esta línea la copiamos y pegamos exactamente en el mismo lugar, solo que ahora en nuestro html. De igual manera linkeamos el javascript en nuestro html (El link sigue hacia la derecha, por eso es importante cogerlo de la página antes dicha).

Texto

Descripción generada automáticamente

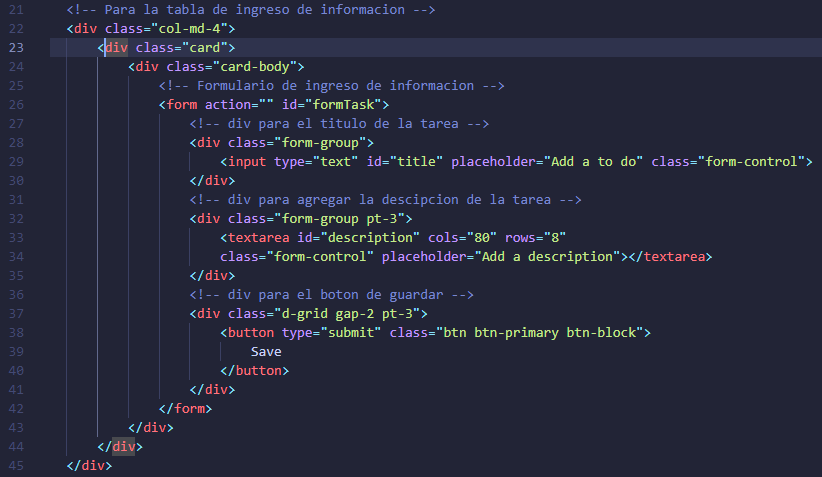
Creamos la barra de navegación dentro del body.



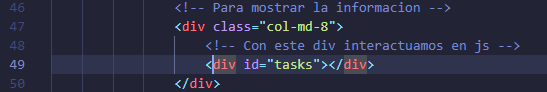
Ahora crearemos un contenedor principal, donde estará todo el cuerpo (container), luego separamos la página en dos (col-md-4, col-md-8), Bootstrap separa en 12 columnas el cuerpo de la página, lo que hacemos es coger 4 para el formulario donde el usuario va a ingresar la información y los otros 8 los tomamos para mostrar la información.



Ahora, crearemos el formulario donde el usuario va a digitar la información.



Dentro de la otra sección (col-md-8) solo pondremos un div, pero esto es porque con este es el que vamos a interactuar en js, porque ahí es donde mostraremos y borraremos la información por medio del LocalStorage.



Hemos terminado de codificar en html, ahora vamos al archivo app.js para empezar a crear la interacción. Empezamos escuchando el evento del formulario, que tendrá que ser submit y cuando eso suceda se correrá la función saveTask.



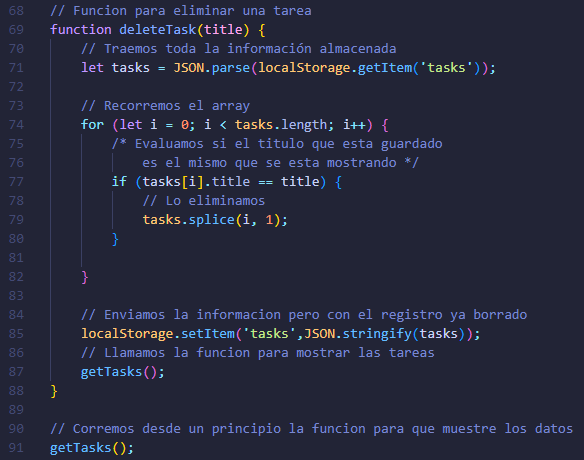
Creamos la función saveTask, esta nos servirá para guardar nuevas tareas. Para hacer esto utilizamos la API de localStorage, donde almacenaremos todos los datos que el usuario ingrese. ¿Para qué es necesario? Es necesario ya que si sólo utilizamos javascript, al recargar la página, se pierde toda la información, en cambio, utilizando esta API, todo permanecerá alojado localmente dentro de la página.



Ahora, creamos otra función llamada “getTasks”, este nos servirá para traernos toda la información que esté almacenada en el localStorage y por consiguiente mostrarla toda en la página, insertando html dentro del javascript.



Por último, creamos la función para eliminar datos, primero nos traemos todos los datos almacenados en el localStorage, recorremos el arreglo y por último, evaluamos si los datos coinciden, si todo esto se cumple, se debe eliminar el registro satisfactoriamente; luego volvemos a enviar todo al localStorage para que se pueda actualizar la vista. Y llamamos afuera del todo la función getTasks para poder que al iniciar se muestren los datos ya ingresados.



Hemos terminado el desarrollo de este aplicativo, y se debería ver así:

Interfaz de usuario gráfica, Aplicación, Word

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación, Teams

Descripción generada automáticamente

Aquí podemos ver el localStorage, donde están almacenados dos registros.

Interfaz de usuario gráfica

Descripción generada automáticamente

