**Introducción**

En este proyecto inicial se ha podido trabajar los elementos básicos para la creación de una página web en local, gracias al uso de JavaScript, HTML y CSS.

Con estas herramientas se ha desarrolla una lista de tareas, en la que se han añadido elementos de forma dinámica mediante un input y un botón. Así como modificar las tareas una vez añadidas a la lista, tachándolas en el caso que estuvieran completadas, o eliminándolas.

De forma adicional en el archivo de JS, se ha añadido la funcionalidad de mantener las tareas, así como su estado una vez actualizado la página.

Dicha información ha sido subida al siguiente repositorio público de github: <https://github.com/iriaprados/to-do_List>

**Estructura principal (HTML)**

En dicho HTML, se puede ver la estructura que ha seguido la página web, compuesta por: el input de texto, que permite introducir al usuario la nueva tarea, junto a un botón que permite que se ejecuté la acción de añadir la tarea.

En la división lista-tareas, se ha añadido la lista en la que dinámicamente se añadirán las tareas una vez introducidas por el usuario.

Texto

El contenido generado por IA puede ser incorrecto.Adicionalmente en la parte superior del código se pueden ver las diversas conexiones realizado con los archivos CSS y JavaScript.

**Estilo y estética (CSS)**

Texto

El contenido generado por IA puede ser incorrecto.En el archivo css que se muestra, se localiza primero el estilo del título, para ello he seleccionado una API de Google que permite añadir fuentes adicionales a las predeterminadas en CSS. Además, de añadir otras características del título como su color, su tamaño de fuente o márgenes.

Texto

El contenido generado por IA puede ser incorrecto.A continuación, se ha determinado el estilo del cuerpo de la página, es decir de la lista de tareas, para ello se ha definido que cada tarea se encuentre centrada en la página, y separada por una línea gris de las otras tareas. En cuanto el formato de la tarea marcada como completada se determina que esta sea tachada.

Por último, se le ha añadido formato a la caja de texto que permite al usuario introducir el contenido de la lista, y el botón que ejecuta este evento.

Para el cuadro de texto, he definido sus medidas, forma y color.

Y para el botón de introducir la tarea he generado un botón con forma semi cuadrada, que resalta su color cuando el usuario pone el cursor encima de el, usando para esto último un hoover.

Texto

El contenido generado por IA puede ser incorrecto.

**Reacción a eventos y funciones generales de la página (JavaScript)**

Para definir la funcionalidad general de la página se ha usado un archivo JavaScript, que permite reaccionar al botón, y que esta añada dinámicamente una tarea a la lista, el liminar dentro de la lista las tareas, marcarlas como completadas. Y que estos elementos se mantengan gracias a un LocalStorage.

En primer lugar, se han definido los elementos para que puedan ser interpretados por el DOM.

Texto

El contenido generado por IA puede ser incorrecto.

La función saveTask, permote que se puedan mantener los elementos estáticos al refrescar la página, gracias que las tareas serán añadidas a una lista, una vez introducidas por el usuario en la lista.

Texto

El contenido generado por IA puede ser incorrecto.

Pantalla de computadora con letras

El contenido generado por IA puede ser incorrecto.Para que se puedan mostrar esta información guardada una vez actualizada la página, se ha creado la función loadTask, que permite obtener las tareas del Json generado al realizar el refresco, y mostrarla por pantalla. Además de capturar su estado en el momento del refresco (completada, añadida …)

Para la creación de los elementos de la lista, se ha usado la función CreateTaskElement. En esta función se han marcado dos elementos principales, las tareas y el estado que tiene (completada, sin completar). Para marcar el estado de la tarea primero se crea una checkbox, que determina el estado de la tarea, y lo guarda por si se actualiza la página.

Además, se ha añadido un botón con forma de X, que permite borrar la tarea una vez pulsado. Esto también es guardado como el estado de la tarea.

Texto

El contenido generado por IA puede ser incorrecto.Adicionalmente se han añadido los elementos con los que el usuario ha interactuado, siendo estos el botón de eliminar, el input, la checkbox.

Por último, se ha creado la función addTask, que permite introducir la tarea a la lista una vez introducida por el usuario en el span y tras pulsar el botón, esta tarea es guardada el en localStorge. Además, en el caso que el usuario intente introducir una tarea vacía la página mostrará una alerta indicándole que necesita introducir información.

Texto

El contenido generado por IA puede ser incorrecto.