

Storyboard de simulación

Utilizar JavaScript para añadir interactividad

1. En la herramienta web VS Code, se empieza con un archivo JavaScript vacío, **script.js**, para añadir interactividad a la página web. Seleccione la **flecha siguiente** para continuar.
2. En primer lugar, añadirá un comentario. A continuación, creará dos variables para contener los elementos de entrada y el elemento que visualizará las tareas. Bajo **script.js**, seleccione el campo junto a 1. Esta es la primera línea del código. Escriba una línea de código en cada campo exactamente como se visualiza en el panel de código. A continuación, pulse **Intro**.

Código del panel de código:

```
//constants declared for input button and task list area  
const taskInput = document.querySelector("#newtask input");  
const taskSection = document.querySelector('.tasks');
```

3. Observe que el comentario se refiere a la finalidad de esta sección de código. Está utilizando un **querySelector** para que JavaScript pueda encontrar los elementos HTML y modificarlos. Seleccione la **flecha siguiente** para continuar.
4. Según los requisitos, los usuarios deben poder añadir una tarea pulsando **Intro** en el teclado. Para indicar esto en el código, puede añadir un escucha de sucesos para la tecla **Intro**. Seleccione la **flecha siguiente** para continuar.
5. Escriba una línea de código en cada campo exactamente como se visualiza en el panel de código. A continuación, pulse **Intro**.

Código del panel de código:

```
//listener for the Enter key. Used to add a new task.  
taskInput.addEventListener("keyup", (e) => {  
if (e.key == "Enter") {createTask(); } });
```

6. Observe que el **escucha de sucesos** tiene conocimiento de *cada* pulsación, pero *solo* llama a "createTask", la función que ejecuta el controlador que acaba de añadir (keyup), cuando los usuarios pulsan la tecla **Intro**. Seleccione la **flecha siguiente** para continuar.
7. Según los requisitos, la página web debe funcionar tanto en dispositivos de escritorio como móviles. Debe incluir un botón **Añadir** para que los usuarios puedan añadir tareas

sin teclado. Para indicar esto en el código, puede añadir un **suceso onclick**. Seleccione la **flecha siguiente** para continuar.

8. Escriba una línea de código en cada campo exactamente como se visualiza en el panel de código. A continuación, pulse **Intro**.

Código del panel de código:

```
//the onclick event for the 'Add' button
document.querySelector('#push').onclick = function () {
createTask(); }
```

9. Observe que el **suceso onclick** llama a la misma función que el **escucha de sucesos** que añadió en el paso anterior para el botón **Añadir**. Eche un vistazo al panel de vista previa para comprobar cómo ha avanzado con la página web. Seleccione la **flecha siguiente** para continuar.
10. La mayor parte del archivo JavaScript contiene código que añade tareas al área de tareas en la página. Esta función se ejecuta cuando un usuario pulsa **Intro** o selecciona el botón **Añadir**. Pero esta función primero debe comprobar que el usuario ha especificado una tarea. Seleccione la **flecha siguiente** para continuar.
11. Escriba una línea de código en cada campo exactamente como se visualiza en el panel de código. A continuación, pulse **Intro**.

Código del panel de código:

```
//the function that creates a task
function createTask() {
if (taskInput.value.length == 0) {
alert("El campo de tarea está en blanco. Escriba un nombre de tarea e inténtelo de nuevo."); } }
```

12. Observe que el código que acaba de escribir tiene una **sentencia if** para comprobar si el campo de entrada de tarea contiene una tarea. Si no es así, la página alertará al usuario. Aún no ha cerrado la función createTask. Lo hará más adelante. Seleccione la **flecha siguiente** para continuar.
13. Ahora ha llegado el momento de escribir código para añadir la tarea que especifica el usuario en el campo. Para ello, utilice una **sentencia else**. Debajo del código que acaba de escribir, escriba una línea de código en cada campo exactamente como se visualiza en el panel de código. A continuación, pulse **Intro**.

Código del panel de código:

```
else {  
}
```

14. Entre las últimas llaves, tiene una sección para la **sentencia else**. Para crear una nueva tarea, puede escribir código que utilizará JavaScript para escribir HTML. El código JavaScript coloca el nuevo código HTML en la sección de tareas de la página web. Seleccione la **flecha siguiente** para continuar.
15. Tiene una larga serie de código para escribir. Entre las llaves de la sentencia else, escriba una línea de código en cada campo exactamente como se visualiza en el panel de código. A continuación, pulse **Intro**.

Código del panel de código:

```
//this block inserts HTML that creates each task into the task area div element  
taskSection.innerHTML +=  
`<div class="task">  
  <label id="taskname">  
    <input onclick="updateTask(this)" type="checkbox" id="check-task">  
    <p>${document.querySelector('#newtask input').value}</p>  
  </label>  
  <div class="delete">  
    <i class="uil uil-trash"></i></div></div>`;
```

16. Observe que uno de los elementos div que el código JavaScript crea para el elemento de tarea es un botón **Suprimir** para cada tarea, lo que permite a los usuarios suprimir una tarea. Seleccione la **flecha siguiente** para continuar.
17. A continuación, debe añadir código para el botón **Suprimir** para que este funcione. Debajo de la última etiqueta div de cierre y dentro de la llave de cierre para la **sentencia else**, escriba una línea de código en cada campo exactamente como se visualiza en el panel de código. A continuación, pulse **Intro**.

Código del panel de código:

```
var current_tasks = document.querySelectorAll(".delete");  
for (var i = 0; i < current_tasks.length; i++) {  
  current_tasks[i].onclick = function () {  
    this.parentNode.remove(); } }
```

18. Debe añadir el código que mostrará y eliminará la barra de desplazamiento. A continuación, vea el aspecto de este código. Seleccione la **flecha siguiente** para continuar.

Continuar añadiendo interactividad a la página web con JavaScript

Para este caso de ejemplo de proyecto, añada código para ajustar una barra de desplazamiento. El código mostrará una barra de desplazamiento si el área en la lista de tareas es mayor a 300 píxeles. Si el área es inferior a 300 píxeles, el código eliminará la barra de desplazamiento.

Puede ver el aspecto de este código en el archivo **script .js**.

Seleccione **X** para cerrar esta ventana y continuar.

19. Observe el código que ha añadido. Este código añade una barra de desplazamiento si el área es mayor a 300 píxeles y elimina la barra de desplazamiento si el área es menor a 300 píxeles. Seleccione la **flecha siguiente** para continuar.
20. Para completar el código, aplicará un estilo de tachado a las tareas que los usuarios han marcado como completadas y lo eliminará para aquellas que no han marcado como completadas. Debajo de la función `createTask`, escriba una línea de código en cada campo exactamente como se visualiza en el panel de código. A continuación, pulse **Intro**.

Código del panel de código:

```
function updateTask(task) {  
  let taskItem = task.parentElement.lastElementChild;  
  if (task.checked) {  
    taskItem.classList.add("checked");  
  }  
  else {taskItem.classList.remove("checked"); } }
```

21. Dedique un momento a revisar lo que ha logrado en la página web de lista de tareas interactiva. Seleccione la **flecha siguiente** para continuar.

Ha creado con éxito el archivo JavaScript en VS Code. La página web ahora es interactiva para que los usuarios puedan utilizar la lista de tareas con diversas funciones.