

Eventos

1. Crear una aplicación que calcule el índice de masa corporal.

Formulario para calcular el Índice de Masa Corporal (IMC). El formulario tiene dos secciones de entrada: 'ESTATURA en centímetros' con un input que contiene '173', y 'PESO en kilogramos' con un input que contiene '62'. Debajo de la sección de peso hay un botón azul que dice 'Calcular'. A la derecha de la sección de peso, hay un campo de texto que muestra el 'Resultado: 20.7'.

Al precionar el boton Calcular debera mostrar el resultado en el input correspondiente.

2. Crear una aplicación de conversion de divisa.

Formulario de conversión de divisa. Muestra dos conversiones: '140,58 peso argentino' y '1 dólar estadounidense'.

Debera tener los valores por defecto, por ejemplo 1 dólar = 140 pesos precio ejemplo, y al hacer algun cambio en alguno de los inputs se debera ver reflejado su correspondiente valor en la moneda a convertir.

3. Realizar una aplicación de notas (el front queda a gusto de cada uno).
 - a. Crear un array donde vamos a guardar las notas
 - b. Agregar un par de notas de prueba como la siguiente

```
{
  id: 1,
  titulo: 'Sacar la basura',
  texto: 'mi mama me va a retar si no lo hago',
  realizada: false
},
```

- c. Crear una variable idGlobal e inicializala en el mismo valor del ultimo id que creaste manualmente, usaremos esto como control de las notas.
- d. Crear un div que va a ser el contenedor de las notas.
- e. Crear una función que pinte las notas en forma de tarjetas dentro del div contenedor.
- f. Sobre el div contenedor, crear una pequeña interfaz para crear nuevas notas: Un input de texto para el titulo, un textarea para el texto y 2 botones, uno para guardar la nota nueva y otro para limpiar los campos.

Formulario para crear una nueva nota. Tiene un título 'Nueva nota'. Debajo hay un input de texto para el 'Titulo' y un textarea para 'Escriba su nota aqui!'. A la derecha hay dos botones: 'Guardar' (verde) y 'Borrar' (rojo).

- g. Crear una función agregarNota la cual necesitara 2 parametros: titulo y texto. La cual deberá crear un objeto de tipo nota como en el punto b y agregarlo al array de notas.
- h. Al presionar el botón guardar deberá guardar en variables los valores de los inputs y verificar si no están vacíos, en cuyo caso deberá llamar a la función que agregara la nueva nota y paso seguido volver a pintar las notas en la pantalla.
- i. Agregar en el template de la tarjeta en la función que pinta un botón con texto “borrar nota” para borrar la nota, para este caso usaremos la propiedad onClick de la etiqueta button y dentro de ella llamaremos a una función que crearemos llamada borrarNota() que recibirá como parámetro el id de la misma.
borrarNota({elemento.id}).
- j. Crearemos la función borrarNota la cual necesitara el parámetro id. La misma deberá eliminar el elemento cuyo id sea igual al recibido por parámetro y volver a pintar las notas para ver reflejado el cambio.
- k. Agregar una validación en la función que pinta las tarjetas, la cual deberá mostrar un mensaje dentro del div contenedor que diga NO HAY NOTAS PARA MOSTRAR en caso de no haber elementos en el array.
- l. Al hacer click en el botón borrar que esta junto con el botón guardar se deberán limpiar los campos de titulo y de texto.
- m. Agregar en el template que pinta las tarjetas un checkbox de la siguiente manera: `<input onClick="marcarRealizada({nota.id})" type="checkbox" {nota.realizada? "checked": ""}>`. El signo de pregunta representa un operador ternario que funciona como un if (https://developer.mozilla.org/es/docs/Web/JavaScript/Reference/Operators/Conditional_Operator), el cual devolverá checked al input y lo colocara como marcado si la propiedad realizada es true y caso contrario no lo hará.
- n. Crear la función marcarRealizada la cual recibirá como parámetro un id y deberá buscar el elemento dentro del array y cambiar la propiedad realizada por el valor contrario al que ya posee y volver a pintar los elementos en pantalla para verlo reflejado.
- o. Crear un par de inputs para realizar filtro por texto y por notas realizadas.

☐ Realizadas

- p. Crear una función que filtre por el estado realizada, la función deberá recibir como parámetro 1 array y devolver lo mismo pero filtrado por los elementos que tengan true en la propiedad realizada.
- q. Crear una función que filtre por texto, la cual recibirá como parámetro un array de notas y un texto. La misma deberá devolver un array filtrado por los elementos que incluyan el texto ingresado en el titulo o el texto de la nota. De no recibir texto deberá retornar el array recibido.
- r. Al cambiar el texto del input de búsqueda o cambiar el valor del checkbox se deberá ver reflejado en pantalla el resultado de los filtros antes mencionados. En ambos casos se deben contemplar los estados de los 2 filtros para poder tener un resultado coherente con lo que se ve en pantalla. Si filtro por texto deberá

contemplar si el checkbox de realizadas esta o no chequeado, y si filtro por realizadas se deberá contemplar el texto que posea el input de búsqueda.

Casos de ejemplo:

1 – Input vacio y checkbox false

☐ Realizadas

☐ **Sacar la basura**

mi mama me va a retar si no lo hago

Borrar nota

☒ **Comer**

~~quedo comida de~~
~~ayer~~

Borrar nota

☐ **Estudiar eventos**

estoy flojo de papeles y no voy a aprobar la task 3

Borrar nota

☒ **Tomar agua**

~~Debo hidratarme~~
~~bien para no~~
~~desmayarme~~

Borrar nota

2 – input vacio y checkbox true

☒ Realizadas

☒ **Comer**

~~quedo comida de~~
~~ayer~~

Borrar nota

☒ **Tomar agua**

~~Debo hidratarme~~
~~bien para no~~
~~desmayarme~~

Borrar nota

3 – Input “ag” y checkbox vacio

ag

☐ Realizadas

☐ **Sacar la basura**

mi mama me va a retar si no lo hago

Borrar nota

☒ **Tomar agua**

~~Debo hidratarme bien para no desmayarme~~

Borrar nota

4 – Input “ag” y checkbox true

ag

☒ Realizadas

☒ **Tomar agua**

~~Debo hidratarme bien para no desmayarme~~

Borrar nota

5 – Input vacío y checkbox true

☒ Realizadas

☒ Comer

~~quedo comida de~~
~~ayer~~

Borrar nota

☒ Tomar agua

~~Debo hidratarme~~
~~bien para no~~
~~desmayarme~~

Borrar nota