

PRUEBA PRÁCTICA

Buscador Web

NOMBRE ALUMNO:

CONTENIDOS: bloque 2, bloque 3, bloque 4, bloque 5

Partes a evaluar (Pista: **Pasos a seguir):**

BUSCADOR DE AUTOS

Marca
Audi

Año
Seleccione

Precio Min
40,000

Precio Max
Seleccione

Puertas
Seleccione

Transmisión
Seleccione

Color
Seleccione

RESULTADOS

AUDI A4 - 2020 - 4 PUERTAS - TRANSMISIÓN: AUTOMATICO - PRECIO: 40000 - COLOR: NEGRO

AUDI A3 - 2017 - 2 PUERTAS - TRANSMISIÓN: MANUAL - PRECIO: 55000 - COLOR: NEGRO

IES ALFONSO XI

- **Primeros pasos:**

- Tenemos 2 archivos .js , el db.js simula ser una base de datos(estos no hay que tocarlos). El otro lo tenéis que crear, y es el principal.
- Crear un evento para comprobar que el documento está cargado

-
- Dividimos el código de nuestro js con comentarios en 3 grandes bloques: variables, eventos y funciones
 - Creamos un elemento con el DIV que tiene id = resultado y querySelector
 - Creamos la función mostrar autos(), la cual llamaremos cuando el DOM este listo.
 - En la función vamos a recorrer el array autos con un foreach y vamos a crear un párrafo con cada coche y toda su información. Después lo añadimos como hijo de resultado.
-
- **Generar un Select de años:**
 - Crea la función llenarSelect
 - La llamaremos después de mostrarAutos()
 - Creamos el elemento año con el id #year y querySelector
 - Creamos dos variables, añoMax y añoMin, para introducir el rango de valores (nota: usando date().getFullYear obtenemos el año actual)
 - En la función llenarSelect: usamos un bucle, ayudándonos de un contador i, para crear las diferentes opciones del select. Primero, creamos un nuevo elemento 'opcion' con createElement, después añadimos su value y su textContent. Finalmente lo agregamos como hijo del elemento año.
-
- **Leer valor de la marca del automóvil:**
 - Ahora vamos a crear un objeto que se llame datosBusqueda, que contenga la diferente información de los automóviles pero vacía, es decir, marca, año, precioMin, etc.
 - Creamos los elementos en el bloque de variables con su id, por ejemplo querySelector('#marca'), etc.. para todos los que hemos añadido al objeto.
 - Ahora creamos un bloque de eventListener para los formularios, un evento para el cambio de valor es 'change'. Entonces cuando cambié el valor del select, llamamos a una arrow function y le pasamos el valor del evento a la propiedad que corresponda en el objeto datosBusqueda, creado anteriormente.

- **Filtrar autos por marca:**

- Creamos una función para filtrar en base a la búsqueda, porque la idea es si seleccionas la marca BMW por ejemplo, que solo aparezcan los autos de esa marca. Esta función la llamamos `filtrarAuto()`.
- Cuando estemos en el arrow función anterior llamamos a esta función y usamos el método `.filter(filtrarMarca)`, para filtrar el array de autos por el value que tiene el evento en ese momento. Mirar un ejemplo en w3school o MDN, ya que estamos llamando a una función dentro del filter y puede ser confuso.
- Crear función `filtrarMarca` y realizar la comparación con un `if,else`. Si la marca del objeto no está vacía devolvemos los elementos del array que su marca, coincide con la marca del objeto `datosBusqueda`. Si no seleccionas nada y está vacío el select, debe devolver todos los autos, esto debes ponerlo en tu `else`.

- **Filtrar autos por año:**

- Ahora llamamos a la función `filtrarAuto()` igual que anteriormente
- lo siguiente es usar el encadenamiento o chaining, si usamos `autos.filter(filtrarMarca).filter(filtrarAño)`, estamos encadenando 2 filtros sobre el mismo array, esto es lo que debes hacer.
- crea la función `filtrarAño()`, es igual que la de filtrar marca pero con el contenido de la propiedad `año` o `year`.
- CUIDADO, los datos obtenidos del objeto puede que estén como `STRING` y no como número, usa el `parseInt`.

- **Mostrar los resultados del Filtrado en el HTML:**

- Pero tenemos que pasarle los valores filtrados, primero a una variable y luego a la función `mostrarAutos()`, esto desde la función `filtrarAutos()`

-
- Hay que modificar la función para que pida un parametro de entrada
 - Crear función limpiarHtml(), que vamos a llamar dentro de la función mostrarAutos() , al inicio antes de recorrer el array
-
- **Filtrar por precio:**
 - Similar a filtrar por marca y filtrar por año
 - Crea 2 funciones filtrarMinimo y filtrarMaximo
-
- **Filtrar por puertas:**
 - Similar al anterior
 - Añade el resto de filtros que queden, es decir, transmisión, etc...
-
- **Mostrar mensaje cuando no hay resultados:**
 - Dentro de la función mostrarAutos, crea un condicional que compruebe el array de resultados, si esta vacío que muestre un mensaje y sino que muestre los vehiculos del array.
 - Cuando esta vacío, llamamos a una función nueva que vamos a crear con el nombre mostrarMensaje().
 - Dentro de la función mostrarMensaje(), Creas un elemento DIV, añades 2 clases: alerta y error. Añade texto a este contenedor.
 - Añade este como hijo de Resultado. Pero antes limpia el HTML.

BUSCADOR DE AUTOS

Marca

Audi

Año

Seleccione

Precio Min

40,000

Precio Max

20,000

Puertas

Seleccione

Transmisión

Seleccione

Color

Seleccione

RESULTADOS

NO HAY RESULTADOS

IES ALFONSO XI