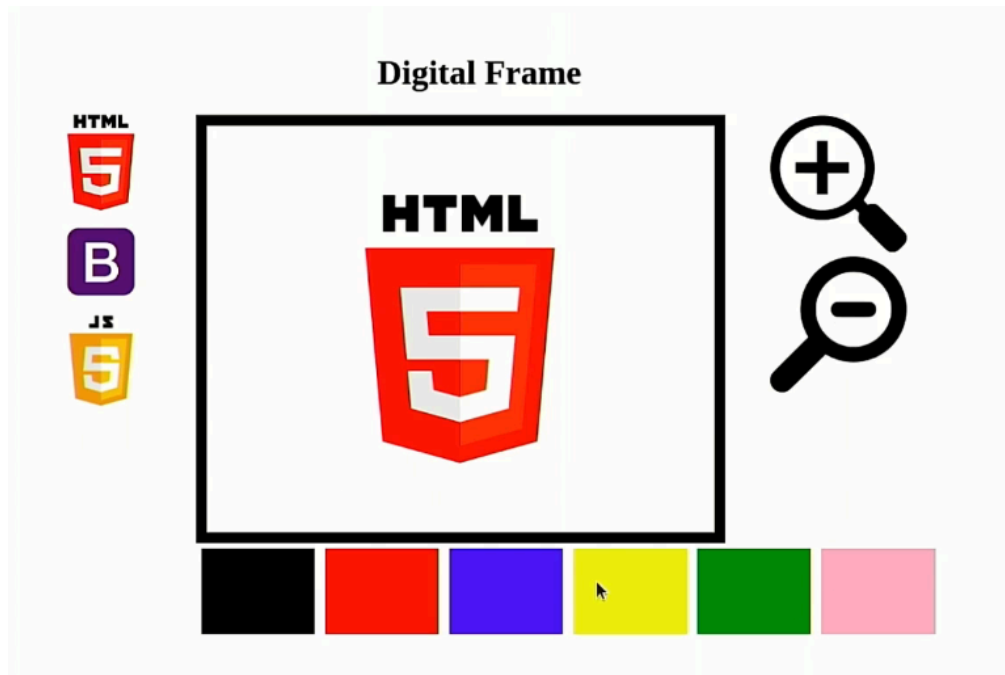


Uso de eventos y traversing en el dom

En los siguientes retos pondremos a prueba lo aprendido respecto a manejo de eventos y traversing.

Reto 1: Realiza el siguiente panel de control de una página html. Los elementos son:

- Unas **miniaturas** de imagen en la parte izquierda
- Una **capa central con borde** que contiene una **imagen**
- Unas imágenes de **lupa + y lupa -** en la derecha.
- Una fila con capas de **distintos colores** en una capa inferior.



Crea esta estructura con html y genera la siguiente funcionalidad:

- ☐ Al hacer **clic izquierdo o arrastrando** las miniaturas dentro de la capa central, el src de la imagen central cambiará, mostrando la seleccionada.
- ☐ Al hacer **clic en alguna de las lupas**, el tamaño de la imagen contenida en la capa central aumentará o disminuirá un valor de pixeles que se parametrizará por código. Controla que **jamás supere el tamaño máximo** la altura de la propia capa que la contiene.
- ☐ Al hacer **clic en los colores**, el **color del borde de la capa central** cambiará al seleccionado.
- ☐ Al hacer **clic derecho dentro de la imagen**, **simularemos** que hemos pinchado en el botón de disminuir el tamaño (lupa -) 3 veces, y lo mismo con el clic izquierdo, simulando que hemos pinchado en el botón de aumentar (lupa +) 3 veces.
- ☐ Al **presionar la imagen central con el botón de scrolling**, restauraremos la imagen al tamaño original.
- ☐ Al pulsar en **cualquiera de los colores**, **mientras** tenemos la **tecla control pulsada**, cambiaremos el color del borde de la imagen central a uno aleatorio generado en el acto.

Reto 2: Realiza las siguientes tareas:

- Dado el siguiente código, utiliza traversing para seleccionar todos los elementos `<p>` dentro del `div` con id "contenedor".

```
<div id="contenedor">
  <p>Hijo 1</p>
  <p>Hijo 2</p>
  <p>Hijo 3</p>
</div>
```

- Encuentra el elemento padre del `` con id "elemento".

```
<ul>
  <li id="elemento">Elemento 1</li>
  <li>Elemento 2</li>
</ul>
```

- Selecciona el elemento hermano inmediato que sigue al span con id "inicio".

```
<div>
  <span id="inicio">Primero</span>
  <span>Segundo</span>
  <span>Tercero</span>
</div>
```

- Encuentra el hermano anterior del `` con id "medio".

```
<div>
  <span>Primero</span>
  <span id="medio">Segundo</span>
  <span>Tercero</span>
</div>
```

- Recorre todos los nodos hijos del `div` (incluyendo nodos de texto) y muestra su tipo y contenido en consola.

```
<div id="lista">
  Texto antes
  <p>Parrafo 1</p>
  <p>Parrafo 2</p>
  Texto después
</div>
```

Para los siguientes ejercicios, utiliza los métodos para crear elementos, modificar contenido, añadir y borrar nodos hijos:

createElement: Crea un nuevo nodo del tipo especificado.

textContent: Permite añadir o modificar texto dentro de un nodo.

appendChild: Añade un nodo como último hijo de un elemento.

removeChild: Elimina un nodo hijo de un elemento padre.

- Crea un nuevo elemento `` con el texto "Elemento 3" y añádelo al final de la lista.

```
<ul id="lista">
  <li>Elemento 1</li>
  <li>Elemento 2</li>
</ul>
```

- Elimina el `<p>` con id "eliminar" usando JavaScript.

```
<div id="contenedor">
  <p>Parrafo 1</p>
  <p id="eliminar">Parrafo 2</p>
  <p>Parrafo 3</p>
</div>
```