

<b>NOMBRE Y APELLIDOS:</b>			<b>FECHA: 01/08/2023</b>		
<b>DOCENTE: MANUEL MACÍAS PÉREZ</b>			<b>NOTA:</b>		
<b>(IFCD0110) CONFECCIÓN Y PUBLICACIÓN DE PÁGINAS WEB</b>			<b>Nº CURSO: 22-35/008902</b>		
MF:	0491	UNIDADES DE APRENDIZAJE A LAS QUE RESPONDE:	UA1	Duración:	2 hrs.
UF:	1842				
PRÁCTICA Nº:	Actividad 23				
<b>DENOMINACIÓN: LOCALSTORAGE</b>					
<p><b>DESCRIPCIÓN</b></p> <p>Se plantea el análisis de la aplicación vista en clase:</p> <ol style="list-style-type: none"> <li>Contesta a las siguientes preguntas: <ol style="list-style-type: none"> <li>Si tengo que reservar un espacio en la página ¿dónde lo haría? <ol style="list-style-type: none"> <li>CSS</li> <li>JavaScript</li> <li>HTML</li> </ol> </li> <li>Si tengo que cambiar los colores por otra paleta ¿en que archivo (html, css o js) y línea lo tenemos que hacer.</li> </ol> </li> </ol> <p>Si te refieres a cambiar de paleta de colores en plena ejecución, lo tienes en el JS entre la línea 4, y la 13:</p> <pre>let hexCodes = ['#e5031d', '#1503e5', '#15e503', '#e5ce03'];  colors.forEach((item, i) =&gt; {   item.addEventListener('click', () =&gt; {     const root = document.querySelector(':root');      root.style.setProperty('--product-img', `url(img/\${item.className}.png)`);     root.style.setProperty('--bg-color', hexCodes[i]);   }) })</pre> <p>Lo que hace este código es rotar entre los divs disponibles en el html (red, Green, etc), y cada vez que se pulse sobre uno de estos, se cambian las variables del css “--product-img” y “--bg-color”, alternando entre 4 valores para cada variable.</p> <p>Si te refieres a cambiar la paleta de colores fuera de ejecución, lo mejor es hacerlo desde el CSS (basta con cambiar el valor de la variable --bg-color. Entre la línea 16 y 19):</p> <pre>:root{   --product-img: url('img/red.png'), url('img/blue.png'), url('img/green.png'), url('img/yellow.png');   --bg-color: #e5031d; }</pre>					

- c. Si tengo que cambiar las imágenes por otras ¿en qué archivo y líneas lo tenemos que hacer?

Si quieres cambiar las imágenes en plena ejecución, también lo tienes entre la línea 4 y 13:

```
let hexCodes = ['#e5031d', '#1503e5', '#15e503', '#e5ce03'];

colors.forEach((item, i) => {
  item.addEventListener('click', () => {
    const root = document.querySelector(':root');

    root.style.setProperty('--product-img',
`url(img/${item.className}.png)`);
    root.style.setProperty('--bg-color', hexCodes[i]);
  })
})
```

Concretamente, la línea que cumple ese rol sería: “root.style.setProperty('--product-img', `url(img/\${item.className}.png)`);” Lo que hace concretamente, es coger el nombre de la clase del div seleccionado, y ponerlo entre “img/” y “.png” para conseguir el nombre de un archivo, que luego se asignará a la variable “--product-img”.

Si lo quieres hacer fuera de ejecución, basta con editar el CSS, y cambiar directamente el valor de “--product-img”.

- d. Las animaciones en que archivo debemos realizar cambios.

En este caso, las animaciones están hechas desde el css. Si queremos hacer algún cambio en una animación, tenemos que cambiar los valores que pasen por los siguientes estilos:

```
.card.active .product-bg{
  clip-path: polygon(0% 0%, 100% 0%, 100% 100%, 0 100%);
}
```

La posición inicial de los bordes dependen de los primeros cuatro valores, mientras que la última posición depende de los cuatro últimos.

Si queremos que la imagen pueda desplazarse, basta con modificar los márgenes máximos.

```
.card.active .product-img{
  margin-top: -100%;
}
```

Y también hay que establecer cuánto tarda a ejecutar la animación de desplazarse, y la duración del desplazamiento.

```
.product-img{  
  width: 250px;  
  height: 300px;  
  background-image: var(--product-img);  
  filter: drop-shadow(10px 10px 10px rgba(0,0,0,0.5));  
  background-size: 100% 100%;  
  transition: margin 1s;  
  transition-delay: .7s;  
}
```

- e. En el supuesto caso de que tengamos que mostrar varios elementos en la misma página (actualmente solo se muestra uno que cambia), que debemos hacer. Razona la respuesta.

Si todos los elementos van a tener la misma animación, basta con utilizar una clase, y aplicar esa animación sobre esta. Si queremos un desplazamiento de imagen, sería así:

```
.nombreClase{  
  transition: margin 1s; //Duración de la animación  
  transition-delay: .7s; //Retraso de la animación  
  margin-top: -100%; //Quitamos margen de la parte superior  
}
```

La práctica se realizará de manera individual.

#### **MEDIOS PARA SU REALIZACIÓN**

- Equipo informático.
- Aplicación Visual Code Studio instalada en el equipo.
- Navegadores actualizados.
- Paquete ofimático.