

El trabajo entregado cumple con todos los puntos del checklist correspondiente.

El zip contiene la carpeta del proyecto con los siguientes archivos:

- 1 - index.html
- 2 - directorio js con los archivos correspondientes
- 3 - directorio css con el archivo de estilos correspondiente
- 4 - directorio img con las imágenes correspondientes
- 5 - checklist de evaluación

Todos los archivos están enlazados en el HTML

Condiciones para aprobar:

Paleta de colores: La paleta de colores se crea de manera dinámica, manipulando el DOM utilizando la librería de JQuery.

La función cargarPaletaDeColores() recibe como parámetro el array con los colores que deben cargarse en cada iteración del bucle for.

La funcionalidad la hice en JQuery siguiendo los pasos siguientes:

Guarde en un variable el contenedor con id paleta para agregar en las iteraciones un nuevo div. Para referenciar el div actual, lo guarde en la variable \$div_actual y con los métodos attr le asigné el id y la clase correspondiente, así como también el color de background.

Asigné al evento click la función getColor() , la cual modifica las variables indicadorDeColor y colorElegido.

Agregué una función que modifica el DOM informando en formato RGB el color elegido.

Grilla de colores: La grilla de colores se crea de manera dinámica, manipulando el DOM utilizando la librería de JQuery.

La función cargarGrillaDeColores() recibe como parámetro el número de px que deben cargarse en el bucle for.

Asigné dos eventos click con las funciones pintarGrilla() para pintar un sólo px y detectarMouse() que modifica la variable global mousePresionado.

Asigné el evento mouseover con la función pintarContinuado() que detecta la variable mousePresionado en true para pintar deslizándose en la grilla de colores.

Borrado de pantalla: el botón borrar desencadena la función `borrarTodo()` que a su vez con un evento click llama a la función `borrarGrilla()` modificando el background de la grilla a valor White con una animación.

Carga superhéroes: para cargar la plantilla de cada superhéroe asigné una función de callback al evento click para cada Li de la clase `imgs`. La función detecta el id del LI clickado y con un switch modifico la variable `$superHeroe` para posteriormente invocar a la función `cargarSuperheroe()`

Guardado: para el botón guardar asigné el evento click que desencadena la función `guardarPixelArt()`