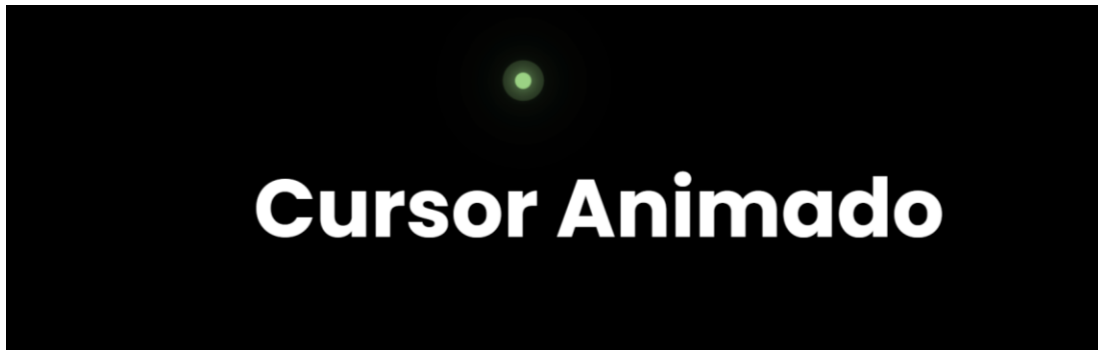


Animación de cursor



Aprendices en esta ocasión aprenderán a realizar una animación sobre el cursor utilizando las propiedades de **mousemove**, vamos al código, nuestro HTML sin novedades:

```
1  <!DOCTYPE html>
2  <html lang="en">
3
4  <head>
5      <meta charset="UTF-8" />
6      <meta http-equiv="X-UA-Compatible" content="IE=edge" />
7      <meta name="viewport" content="width=device-width, initial-scale=1.0" />
8      <title>Animacion del Cursor</title>
9      <link rel="stylesheet" href="style.css" />
10 </head>
11
12 <body>
13     <div class="cursor"></div>
14     <div class="context">
15         <h1>Cursor Animado</h1>
16     </div>
17     <script src="index.js"></script>
18 </body>
19
20 </html>
```

Vamos al CSS:

```
1  @import url("https://fonts.googleapis.com/css2?family=Poppins:wght@100;200;300;400;500;600;700;800;900&display=swap");
2
3  * {
4      margin: 0;
5      padding: 0;
6      box-sizing: border-box;
7  }
8
9  body {
10     display: flex;
11     justify-content: center;
12     align-items: center;
13     min-height: 100vh;
14     font-family: "Poppins", sans-serif;
15     cursor: none;
16     background-color: black;
17 }
18
19 h1 {
20     color: white;
21     font-size: 100px;
22 }
23
```



@hdtoledo

```

24 .cursor {
25   width: 20px;
26   height: 20px;
27   border-radius: 50%;
28   background: skyblue;
29   pointer-events: none;
30   box-shadow: 0 0 20px skyblue, 0 0 60px skyblue, 0 0 100px skyblue;
31   z-index: 999;
32   display: none;
33   position: fixed;
34   transform: translate(-50%, 50%);
35   animation: colors 3s infinite;
36   -webkit-animation: colors 3s infinite;
37 }
38
39 @keyframes colors {
40   0% {
41     filter: hue-rotate(0deg);
42     -webkit-filter: hue-rotate(0deg);
43   }
44   100% {
45     filter: hue-rotate(360deg);
46     -webkit-filter: hue-rotate(360deg);
47   }
48 }
49
50 .cursor::before {
51   content: "";
52   position: absolute;
53   transform: translate(-30%, -30%);
54   width: 50px;
55   height: 50px;
56   background: skyblue;
57   border-radius: 50%;
58   opacity: 0.2;
59 }
60

```

Hablemos un poco de nuestro CSS detalladamente:

1. @import

url("https://fonts.googleapis.com/css2?family=Poppins:wght@100;200;300;400;500;600;700;800;900&display=swap"); Esta línea importa la fuente de Google Fonts llamada "Poppins" con diferentes pesos (100 a 900) y la hace disponible para su uso en el resto del código.

2. * { ... } Este bloque de código establece que se apliquen las siguientes propiedades a todos los elementos de la página:

- **margin: 0;** establece los márgenes en 0.
- **padding: 0;** establece el relleno en 0.
- **box-sizing: border-box;** establece el modelo de caja como "border-box", lo que significa que el tamaño total de un elemento incluye el contenido, el relleno y el borde, pero no el margen exterior.

3. body { ... } Este bloque de código define las propiedades para el elemento **body**, que es el cuerpo principal de la página:

- **display: flex;** hace que el cuerpo sea un contenedor flexible.



@hdtoledo

- **justify-content: center;** alinea los elementos secundarios horizontalmente en el centro.
 - **align-items: center;** alinea los elementos secundarios verticalmente en el centro.
 - **min-height: 100vh;** establece la altura mínima del cuerpo en el 100% de la altura visible del viewport.
 - **font-family: "Poppins", sans-serif;** establece la fuente de la familia "Poppins" como la fuente utilizada para el texto en el cuerpo.
 - **cursor: none;** oculta el cursor del ratón.
 - **background-color: black;** establece el color de fondo del cuerpo en negro.
4. **h1 { ... }** Este bloque de código define las propiedades para los elementos **h1** (encabezados de nivel 1) en la página:
- **color: white;** establece el color del texto en blanco.
 - **font-size: 100px;** establece el tamaño de fuente en 100 píxeles.
5. **.cursor { ... }** Este bloque de código define las propiedades para una clase llamada "cursor", que se utiliza para crear un efecto de cursor personalizado en la página:
- **width: 20px;** establece el ancho del cursor en 20 píxeles.
 - **height: 20px;** establece la altura del cursor en 20 píxeles.
 - **border-radius: 50%;** establece el radio de borde para crear una forma circular.
 - **background: skyblue;** establece el color de fondo del cursor en celeste.
 - **pointer-events: none;** hace que el cursor no reaccione a eventos de puntero, lo que permite que los elementos debajo del cursor sean interactivos.
 - **box-shadow: 0 0 20px skyblue, 0 0 60px skyblue, 0 0 100px skyblue;** crea una sombra alrededor del cursor con diferentes niveles de desenfoque.
 - **z-index: 999;** establece el orden de apilamiento del cursor para que esté en la parte superior de otros elementos.
 - **display: none;** inicialmente oculta el cursor.
 - **position: fixed;** establece la posición del cursor como fija en relación con la ventana del navegador.
 - **transform: translate(-50%, 50%);** mueve el cursor hacia arriba y hacia la izquierda en un 50% de su tamaño.
 - **animation: colors 3s infinite;** aplica una animación llamada "colors" durante 3 segundos en bucle infinito.
 - **-webkit-animation: colors 3s infinite;** la misma animación pero para navegadores basados en WebKit (como Safari y Chrome).



@hdtoledo

6. **@keyframes colors { ... }** Este bloque de código define una animación llamada "colors" que se utiliza en el cursor:
- **0% { ... }** establece los estilos en el inicio de la animación (0% del progreso).
 - **100% { ... }** establece los estilos al final de la animación (100% del progreso).
 - **filter: hue-rotate(0deg);** aplica una rotación de tono de 0 grados al cursor.
 - **-webkit-filter: hue-rotate(0deg);** la misma rotación de tono para navegadores basados en WebKit.
7. **.cursor::before { ... }** Este bloque de código define las propiedades para un pseudo-elemento **::before** del cursor:
- **content: "";** define el contenido del pseudo-elemento como vacío.
 - **position: absolute;** establece la posición del pseudo-elemento como absoluta en relación con el cursor.
 - **transform: translate(-30%, -30%);** mueve el pseudo-elemento hacia arriba y hacia la izquierda en un 30% de su tamaño.
 - **width: 50px;** establece el ancho del pseudo-elemento en 50 píxeles.
 - **height: 50px;** establece la altura del pseudo-elemento en 50 píxeles.
 - **background: skyblue;** establece el color de fondo del pseudo-elemento en celeste.
 - **border-radius: 50%;** establece el radio de borde para crear una forma circular.
 - **opacity: 0.2;** establece la opacidad del pseudo-elemento en 0.2.

Ahora nuestro JS:

```
1  const cursorAni = document.querySelector(".cursor");
2  var timeout;
3
4  document.addEventListener("mousemove", movement);
5
6  function movement(e){
7      console.log("moving")
8      let x = e.pageX;
9      let y = e.pageY;
10     cursorAni.style.left = x + "px";
11     cursorAni.style.top = y + "px";
12     cursorAni.style.display = "block"
13
14     clearTimeout(timeout)
15     timeout = setTimeout(() => {
16         cursorAni.style.display = "none"
17     }, 2000);
18 }
19
20 document.addEventListener("mouseout", ()=>{
21     cursorAni.style.display = "none"
22 })
```



@hdtoledo

Este código JavaScript controla el comportamiento del cursor animado definido en el código CSS anterior. A continuación, te explico cada parte del código:

1. **const cursorAni = document.querySelector(".cursor");** Esta línea selecciona el elemento del DOM con la clase "cursor" y lo almacena en la variable **cursorAni**.
2. **var timeout;** Esta línea declara una variable **timeout** sin asignarle ningún valor inicial.
3. **document.addEventListener("mousemove", movement);** Este código agrega un evento de escucha al documento que espera el evento de movimiento del ratón. Cuando se produce dicho evento, se llama a la función **movement**.
4. **function movement(e) { ... }** Esta función se ejecuta cuando se produce un evento de movimiento del ratón en el documento. Toma el objeto de evento **e** como parámetro.
 - **console.log("moving");** muestra el mensaje "moving" en la consola. Puede ser útil para verificar si el evento de movimiento del ratón se está activando correctamente.
 - **let x = e.pageX;** obtiene la posición horizontal del puntero del ratón en relación con la página.
 - **let y = e.pageY;** obtiene la posición vertical del puntero del ratón en relación con la página.
 - **cursorAni.style.left = x + "px";** establece la propiedad CSS **left** del elemento **cursorAni** para que coincida con la posición horizontal del puntero del ratón.
 - **cursorAni.style.top = y + "px";** establece la propiedad CSS **top** del elemento **cursorAni** para que coincida con la posición vertical del puntero del ratón.
 - **cursorAni.style.display = "block";** muestra el elemento **cursorAni** estableciendo su propiedad CSS **display** en "block".
 - **clearTimeout(timeout);** cancela cualquier temporizador previo que se haya iniciado.
 - **timeout = setTimeout(() => { ... }, 2000);** establece un temporizador que ocultará el elemento **cursorAni** después de 2000 milisegundos (2 segundos). La función de devolución de llamada proporcionada se ejecutará después de que expire el tiempo especificado.
5. **document.addEventListener("mouseout", () => { ... });** Este código agrega un evento de escucha al documento que espera el evento de salida del ratón. Cuando se produce dicho evento, se ejecuta una función de devolución de llamada.
 - **cursorAni.style.display = "none";** oculta el elemento **cursorAni** estableciendo su propiedad CSS **display** en "none". Esto se activa cuando el ratón sale del área de la página.

En resumen, este código controla el posicionamiento y la visibilidad de un cursor animado en la página web en respuesta a los eventos de movimiento del ratón. El cursor se muestra cuando el ratón se mueve y se oculta después de un tiempo de inactividad o cuando el ratón sale del área de la página.



@hdtoledo