

HTML

2.9 Practicamos

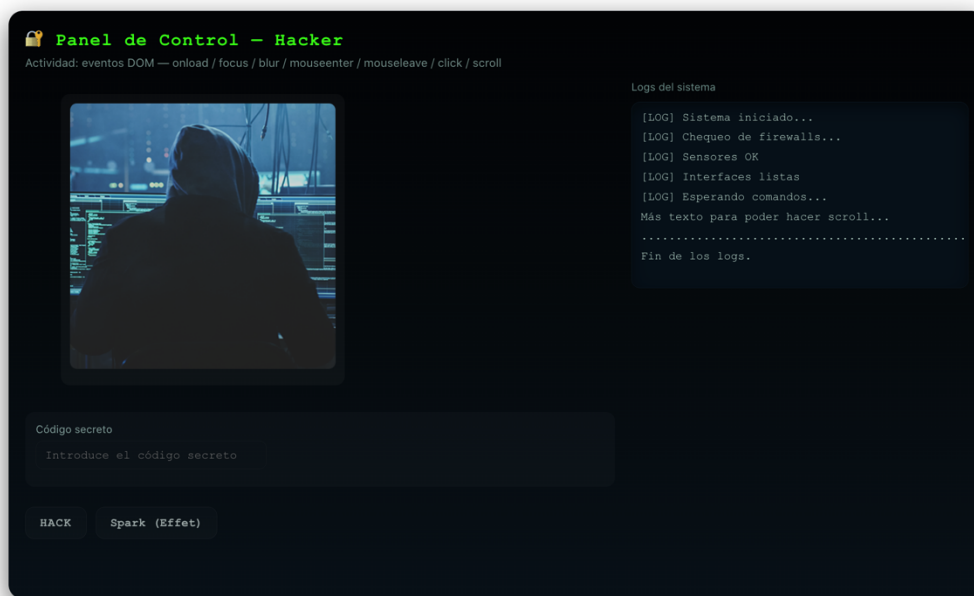


Autor: Lisa ERIKSEN

Fecha: 2025 / 2026

Panel de Control Hacker

Objetivo: practicar eventos del DOM (onLoad, focus, blur, mouseenter, mouseleave, click, scroll) creando un panel de control interactivo. Tu misión: escribir el *código JavaScript* para crear la interacción.



Actividad:

1. Descarga la carpeta hacker con el index.html y el styles.css.
2. Crea un archivo app.js donde escribirás las funciones JavaScript como en este ejemplo:

```
function iniciarSistema() {  
  document.getElementById("bootMsg").innerText = "🔥 Sistema inicializado";  
}
```

3. Usa los eventos aprendidos en la actividad anterior para generar efectos visuales e interacciones, siguiendo las instrucciones indicadas en la ficha:

1.1 Inicio del sistema — Evento load

Elemento: `l#core` (imagen del núcleo del sistema)

Tarea: cuando la imagen termine de cargarse, mostrar el mensaje:

🔥 Sistema inicializado



Pista:

- Crea una función con `document.getElementById("bootMsg").innerText = "...";`
- Llama esa función a dentro del imagen, con un evento `onLoad`.

1.2 Campo del código secreto — Eventos focus y blur

Elemento: `#codigo`

Tareas:

- Cuando el usuario entre en el campo → mostrar:  Introduce el código...
- Cuando salga → mostrar:  Has salido del campo.

Pista:

- Crea dos funciones con `document.getElementById` y `innerText`
- Llama esas funciones a dentro del input, con eventos `onFocus` y `onBlur`.

1.3 Botón sensible al ratón — `mouseenter` / `mouseleave`

Elemento: `#hackBtn`

Tareas:

- Cambiar el color de fondo cuando el ratón entra en el botón.
- Restaurarlo cuando el ratón sale del botón.

Pista:

- `mouseenter` ≠ `mouseover`. Usa `mouseenter` para evitar activaciones múltiples.
- Crea una función con `document.getElementById` y `style.backgroundColor`
- `style.backgroundColor = ""` reinicia el color

1.4 Acción al hacer clic — Evento click

Elemento: #hackBtn

Tarea: Al hacer clic, lanzar una alerta o algún efecto visual de tu elección.

Pista:

- *Prueba usando alert() para validar el funcionamiento antes de añadir estilos dinámicos.*

1.5 Zona de logs — Evento scroll

Elemento: #logs

Tarea: Cuando el usuario haga scroll, mostrar: Desplazamiento detectado en los logs.

Pista:

- *Crea una función con document.getElementById y innerText*