

# Informe de Laboratorio 03

## Tema: Laboratorio 03

Nota

Estudiante	Escuela	Asignatura
Jhonatan David Arias Quispe jariasq@unsa.edu.pe	Escuela Profesional de Ingeniería de Sistemas	Programación Web 2 Semestre: II Código: 1702122

Laboratorio	Tema	Duración
03	Laboratorio 03	04 horas

Semestre académico	Fecha de inicio	Fecha de entrega
2024 - A	Del 15 Mayo 2024	Al 18 Mayo 2024

## 1. Actividades

### 1.1. Descripción

- Programar en Javascript sobre una pagina web html basica.
- Ejercicio 01: Cree un teclado random para banca por internet.
- Ejercicio 02: Cree una calculadora básica como la de los sistemas operativos, que pueda utilizar la funcion eval().
- Cree una versión del juego "el ahorcado" que grafique con canvas paso a paso desde el evento onclick() de un botón.

### 1.2. Pregunta

- Explique una herramienta para ofuzcar código JavaScript.
- Muestre un ejemplo de su uso en uno de los ejercicios de la tarea..
- Adjunte a su repositorio ambas versiones:
  - script-ejercicio-01.js (development).
  - script-ejercicio-01.min.js (production).

## 2. Equipos, materiales y temas utilizados

- Sistema Operativo ArchCraft GNU Linux 64 bits Kernell
- NeoVim
- Git 2.42.0
- Cuenta en GitHub con el correo institucional.
- HTML5
- CSS3
- JavaScript
- JavaScript Obfuscador
- Latex

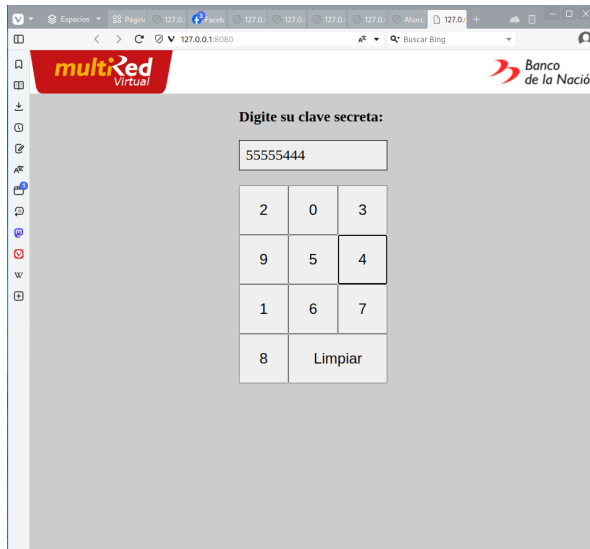
## 3. URL de Repositorio Github

- URL del Repositorio GitHub para clonar o recuperar.
- <https://github.com/JhonatanDczel/pweb2.git>
- URL para el laboratorio 03 en el Repositorio GitHub.
- <https://github.com/JhonatanDczel/pweb2/tree/main/lab03>

## 4. Antecedentes

Javascript es un lenguaje multidinamico usado en su mayoria en el desarrollo web como lenguaje de front end y backend (Gracias a node) y como tal, surgieron muchas herramientas para ofuscar el codigo en JavaScript, una de ellas JavaScript Obfuscator Tool, la podemos encontrar en la siguiente direccion: <https://obfuscator.io/code>

## Ejercicio 01 - Teclado Random



Este informe detalla la creación y configuración de un documento HTML dinámico utilizando JavaScript puro. El código tiene como objetivo generar una interfaz de usuario (UI) que incluye una barra de navegación con dos logotipos y un área principal que contiene un campo de entrada de texto simulado y un teclado numérico aleatorio. A continuación, se explica cada parte del código con porciones de código relevantes.

### Creación de la Barra de Navegación

Primero, se crea un elemento `nav` y se agregan dos imágenes de logotipos a esta barra de navegación.

```
1 let nav = document.createElement("nav");
2
3 let logo = document.createElement("img");
4 logo.src = "./img/logo-multired.jpg";
5 nav.appendChild(logo);
6
7 let logo2 = document.createElement("img");
8 logo2.src = "./img/Logo_BN.jpg";
9 logo2.style.height = "40px";
10 logo2.style.width = "auto";
11 logo2.style.alignSelf = "center";
12 nav.appendChild(logo2);
```

En esta parte del código, se crean los elementos `nav` y `img` utilizando `document.createElement`. Los atributos de las imágenes (como `src` y estilos) se definen y las imágenes se añaden como hijos del elemento `nav`.

### Estilos para la Barra de Navegación

Luego, se aplican estilos CSS al `nav` para estructurarlo visualmente.

```
1 nav.style.display = "flex";
2 nav.style.justifyContent = "space-between";
3
4 document.body.appendChild(nav);
```

Se utiliza `style` para establecer la propiedad `display` como `flex`, lo que permite una distribución flexible de los elementos hijos, y `justifyContent` para espaciar los elementos (los logotipos) uniformemente.

## Creación del Área Principal

A continuación, se crea el elemento `main` y un contenedor `div` dentro de él.

```
1 let main = document.createElement("main");
2 document.body.appendChild(main);
3
4 let mainDiv = document.createElement("div");
5 main.appendChild(mainDiv);
```

## Adición de la Etiqueta y Campo de Entrada

Dentro del `div` principal, se agrega un `p` para la etiqueta y otro `p` para simular el campo de entrada de texto.

```
1 let label = document.createElement("p");
2 label.textContent = "Digite su clave secreta:";
3 label.style.fontSize = "1.5rem";
4 label.style.fontWeight = "bold";
5 mainDiv.appendChild(label);
6
7 let camp = document.createElement("p");
8 camp.style.fontSize = "1.5rem";
9 mainDiv.appendChild(camp);
10 camp.style.minHeight = "1.5rem";
11 camp.style.padding = "10px 10px";
12 camp.style.border = "1px solid black";
13 camp.style.backgroundColor = "#eee";
14 camp.style.display = "flex";
15 camp.style.alignContent = "center";
16 camp.style.overflow = "hidden";
```

Estos elementos de párrafo (`p`) se estilizan para parecer un campo de entrada de texto. El campo de entrada (`camp`) tiene estilos adicionales para simular un área de entrada segura.

## Función para Mezclar los Números

Se define una función `shuffle` para mezclar un array de números, que se usará para crear el teclado numérico aleatorio.

```
1 function shuffle(array) {
2   for (let i = array.length - 1; i > 0; i--) {
3     const j = Math.floor(Math.random() * (i + 1));
4     [array[i], array[j]] = [array[j], array[i]];
5   }
6   return array;
7 }
8
9 let numbers = Array.from({ length: 10 }, (_, i) => i);
10 numbers = shuffle(numbers);
```

La función `shuffle` utiliza el algoritmo de Fisher-Yates para mezclar los elementos de un array.

## Creación del Teclado Numérico

El teclado numérico se crea como un `div` y se añaden botones numerados, incluyendo un botón de "Limpiar".

```
1 let keyboard = document.createElement("div");
2
3 for (let i = 0; i <= 10; i++) {
4   let key = document.createElement("button");
5   let name = i < 10 ? numbers[i] : "Limpiar";
6   if (i == 10) {
7     key.style.gridColumn = "span 2";
8   }
9   key.textContent = name;
10  key.style.height = "80px";
11  keyboard.appendChild(key);
12  key.addEventListener("click", () => {
13    if (name == "Limpiar") {
14      camp.textContent = "";
15      return;
16    }
17    camp.textContent += name;
18  });
19  key.style.fontSize = "1.5rem";
20 }
21
22 mainDiv.appendChild(keyboard);
```

Cada botón tiene un `event listener` que actualiza el campo de entrada (`camp`) con el número correspondiente o lo limpia si se presiona "Limpiar".

## Estilos Adicionales para el Documento y los Elementos

Se aplican estilos globales y específicos a varios elementos para asegurar una presentación consistente.

```
1 document.documentElement.style.margin = 0;
2 document.documentElement.style.padding = 0;
3 document.body.style.margin = 0;
4 document.body.style.padding = 0;
5
6 document.body.style.height = "calc(100vh - 70px)";
7 main.style.width = "100vw";
8 main.style.height = "calc(100vh - 70px)";
9 main.style.display = "flex";
10 main.style.backgroundColor = "#ccc";
11 main.style.justifyContent = "center";
12 main.style.alignContent = "center";
13
14 mainDiv.style.width = "240px";
15 mainDiv.style.height = "100px";
16
17 keyboard.style.display = "grid";
18 keyboard.style.gridTemplateColumns = "1fr 1fr 1fr";
```

Estos estilos aseguran que el documento ocupe toda la ventana del navegador, que el contenido principal esté centrado y que el teclado numérico esté distribuido en una cuadrícula de tres columnas.

## Conclusión

El código genera dinámicamente una interfaz de usuario utilizando JavaScript, creando una barra de navegación con logotipos, un área principal con un campo de entrada de texto simulado y un teclado numérico con números aleatorios y un botón de "Limpiar". Los elementos y estilos se gestionan mediante el DOM y CSS en línea para lograr la funcionalidad y apariencia deseada.