Informe de Laboratorio 02

Tema: JavaScript

Nota

Estudiante	Escuela	Asignatura
Rutbel Carlos Ttito Campos rttitoca@unsa.edu.pe	Escuela Profesional de Ingeniería de Sistemas	Programación Web 2 Semestre: III Código: 20231001

Laboratorio	Tema	Duración
02	JavaScript	04 horas

Semestre académico	Fecha de inicio	Fecha de entrega
2023 - A	Del 10 Abril 2023	Al 17 Abril 2023

1. Equipos, materiales y temas utilizados

- Sistema Operativo Ubuntu 22.04.2 LTS
- Git 2.39.2.
- Cuenta en GitHub con el correo institucional.
- HTML, CSS y JavaScript

2. Tarea

- Ejercicio 01: Utilizando JavaScript, genere un teclado aleatorio para Banca por Internet.

- Ejercicio 02: Utilizando JavaScript, genere una calculadora como la siguiente imagen.

- Cada vez que se presiona ENTER se guardan los datos involucrados en un pila.
- Con memoria y botones para recorrer la pila.
- Agregue 4 botones:

- M1: memoria
- M2: memoria
- ¡-: Moverse para abajo en la pila.
- -¿:Moverse para arriba en la pila.

3. Solucion de Ejercicios

3.1. Ejercicio 1

3.1.1. Archivo HTML

Listing 1: banco.html

```html
<!DOCTYPE html>
<html>
<head>
  <title>Banca por Internet</title>
</head>
<body>
  <input type="text" id="clave" readonly>
  <div id="teclado"></div>

  <script src="script.js"></script>
</body>
</html>
```

3.1.2. Archivo JS

Listing 2: script.js

```javascript
function generarTeclado() {
    // Array de numeros del 0 al 9
    var numeros = ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9'];
    // Desordenar el array de numeros aleatoriamente
    numeros.sort(function() { return 0.5 - Math.random() });
    // Crear los botones del teclado con los numeros aleatorios
    numeros.forEach(function(numero) {
        var boton = document.createElement('button');
        boton.textContent = numero;
        boton.addEventListener('click', function() {
          generarClave(numero);
        });
        document.getElementById('teclado').appendChild(boton);
    });
}
// Funcion para generar una clave a partir de los numeros presionados
function generarClave(numero) {
    var clave = document.getElementById('clave').value;
    clave += numero;
    document.getElementById('clave').value = clave;
}
// Llamar a la funcion para generar el teclado al cargar la pagina
```

```
window.addEventListener('load', generarTeclado);
```

3.1.3. Capturas de Ejecucion

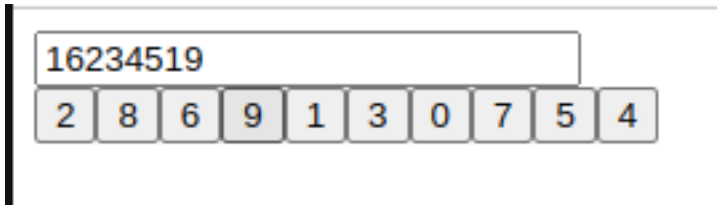

Figura 1: Teclado Aleatorio

3.2. Ejercicio 2

3.2.1. Archivo HTML

Listing 3: calculadora.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Calculadora</title>
  <link rel="stylesheet" href="estilos.css">
</head>
<body>
  <h1>Calculadora</h1>
  <div class="calculator">
    <div class="screen"></div>
    <div class="buttons">
      <div class="button" onclick="handleButton('7')">7</div>
      <div class="button" onclick="handleButton('8')">8</div>
      <div class="button" onclick="handleButton('9')">9</div>
      <div class="button" onclick="handleButton('/')">/</div>
      <div class="button" onclick="handleButton('4')">4</div>
      <div class="button" onclick="handleButton('5')">5</div>
      <div class="button" onclick="handleButton('6')">6</div>
      <div class="button" onclick="handleButton('*')">*</div>
      <div class="button" onclick="handleButton('1')">1</div>
      <div class="button" onclick="handleButton('2')">2</div>
      <div class="button" onclick="handleButton('3')">3</div>
      <div class="button" onclick="handleButton('-')">-</div>
      <div class="button" onclick="handleButton('0')">0</div>
      <div class="button" onclick="handleButton('.')">.</div>
      <div class="button" onclick="handleButton('+')">+</div>
      <div class="button" onclick="handleCalculate()">ENTER</div>
    </div>
    <div class="memory-buttons">
      <div class="button" onclick="saveToMemory(1)">M1</div>
      <div class="button" onclick="saveToMemory(2)">M2</div>
      <div class="button" onclick="moveDownStack()">&#60;-</div>
      <div class="button" onclick="moveUpStack()">->&gt;</div>
    </div>
```

```
  </div>
  <script src="script2.js"></script>
</body>
</html>
```

3.2.2. Archivo CSS

Listing 4: estilos.css

```
.calculator {
    width: 240px;
    margin: 0 auto;
}

.screen {
    width: 100%;
    height: 60px;
    background-color: #f2f2f2;
    display: flex;
    align-items: center;
    justify-content: flex-end;
    padding: 10px;
    box-sizing: border-box;
}

.buttons {
    display: grid;
    grid-template-columns: repeat(4, 1fr);
    gap: 10px;
    padding: 10px;
}

.button {
    height: 40px;
    display: flex;
    align-items: center;
    justify-content: center;
    background-color: #f2f2f2;
    cursor: pointer;
    user-select: none;
}

.button:hover {
    background-color: #d9d9d9;
}

.memory-buttons {
    display: flex;
    justify-content: space-between;
    margin-top: 10px;
}
```

3.2.3. Archivo JS

Listing 5: script2.js

```
const screen = document.querySelector('.screen');
const memory = {
    m1: null,
    m2: null
};
let stack = [];
function handleButton(value) {
    screen.textContent += value;
}

function handleCalculate() {
    const result = eval(screen.textContent);
    stack.push(result);
    screen.textContent = '';
}
function saveToMemory(memoryIndex) {
    const result = screen.textContent;
    if (memoryIndex === 1) {
      memory.m1 = result;
    } else if (memoryIndex === 2) {
      memory.m2 = result;
    }
    screen.textContent = '';
}
function moveDownStack() {
    if (stack.length > 0) {
      const currentValue = screen.textContent;
      stack.push(currentValue);
      const valueFromStack = stack.shift();
      screen.textContent = valueFromStack;
    }
}

function moveUpStack() {
    if (stack.length > 0) {
      const currentValue = screen.textContent;
      stack.unshift(currentValue);
      const valueFromStack = stack.pop();
      screen.textContent = valueFromStack;
    }
}
```

3.2.4. Capturas de Ejecucion

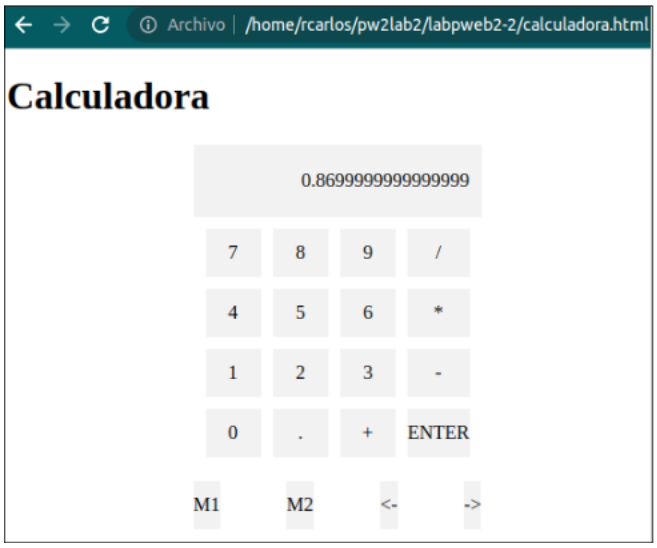

Figura 2: Ejemplo Calculadora

Figura 3: Ejemplo de operacion en Calculadora

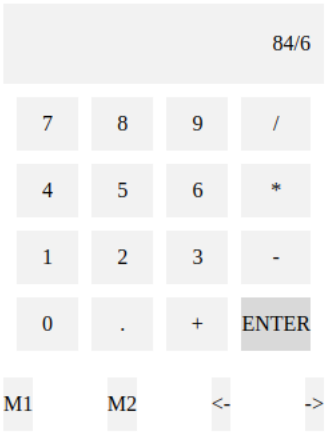

Figura 4: Ejemplo operacion con Memoria Calculadora

4. URL de Repositorio Github

- URL del Repositorio GitHub para clonar o recuperar:
- `https://github.com/RutbelCarlosTC/pw2-lab-d-23a.git`
- URL para el laboratorio 04 en el Repositorio GitHub.
- `https://github.com/RutbelCarlosTC/pw2-lab-d-23a/tree/main/Lab2`

5. Actividades con el repositorio GitHub

5.1. Commits

- Commits Ejercicio 1

```
commit 88033a26fe5b53f89540e30ab172895b492cd086
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date:   Sun May 14 20:26:31 2023 -0500

    se llama a la funcion generarTeclado

commit 212bf5bbd848e56dcc1a71b736255634f8995a15
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date:   Sun May 14 20:25:30 2023 -0500

    se agreo funcion para generar clave a partir de los botones presionados

commit 76f787ee4605fe60977345bee4feb88bafa82746
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date:   Sun May 14 20:24:02 2023 -0500

    se crearon los botones con numeros aleatorios

commit a09a9723a6ca6146cc4d8bcec058abd53f761c14
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date:   Sun May 14 20:23:01 2023 -0500

    se desordeno el array de numeros

commit ae12b2cc60138905efa8006422b3295bfc5b55e3
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date:   Sun May 14 20:22:13 2023 -0500

    se agrego funcion inicial generar teclado

commit 6003b9775dfda808cff274ddea777657d2e7631f
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date:   Sun May 14 20:21:00 2023 -0500

    se agrego html inicial
```

- Commits Ejercicio 2

```
commit 80041214ae42f2b1c7e391c27f063d9c913dc2d8
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date:   Sun May 14 20:41:38 2023 -0500

    se agrego funcion para guardar en M1 y M2

commit da7c9957c6a0a69db6d13d4c3737f22ddbc92fe8
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date:   Sun May 14 20:40:56 2023 -0500

    se agrego funcion para ver la operaciones efectuadas en la pantalla

commit e459ad37d274b5f968e1a3c26bbb8063f4e71168
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date:   Sun May 14 20:39:45 2023 -0500

    se agrego la pantalla, objeto memoria y la pila

commit 185c876560aed96eda5249e7e2b94072668ebca3
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date:   Sun May 14 20:36:40 2023 -0500

    mostrando los botones

commit 1300ed181e189cca97edb97751b62a2adc75e4fd
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date:   Sun May 14 20:32:37 2023 -0500

    se agreo funcion para actulizar la pila

commit e6f8736f4f4c1b9f6c98b26c6e302a78b0f04eeb
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date:   Sun May 14 20:31:35 2023 -0500

    se agrego funcion para agregar elementos a la pila

commit 35a6e1dc8b9105be4c19f1006a390bf4fd820dc1
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date:   Sun May 14 20:30:42 2023 -0500

    se agrego funcion para el ecento tecla presionada

commit 267ef5f1ba53afca8f5b2e413442440ecce121e2
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date:   Sun May 14 20:29:52 2023 -0500

    se agrego script2 incial
```

6. Pregunta: ¿Para qué sirve el directorio pycache?

Se utiliza para almacenar los archivos de código compilado en formato de bytecode de Python (*.pyc) para mejorar el rendimiento en futuras ejecuciones.

7. Rúbricas

7.1. Entregable Informe

Tabla 1: Tipo de Informe

Informe	
Latex	El informe está en formato PDF desde Latex, con un formato limpio (buena presentación) y facil de leer.

7.2. Rúbrica para el contenido del Informe y demostración

- El alumno debe marcar o dejar en blanco en celdas de la columna **Checklist** si cumplio con el ítem correspondiente.
- Si un alumno supera la fecha de entrega, su calificación será sobre la nota mínima aprobada, siempre y cuando cumpla con todos lo items.
- El alumno debe autocalificarse en la columna **Estudiante** de acuerdo a la siguiente tabla:

Tabla 2: Niveles de desempeño

	Nivel			
Puntos	Insatisfactorio 25 %	En Proceso 50 %	Satisfactorio 75 %	Sobresaliente 100 %
2.0	0.5	1.0	1.5	2.0
4.0	1.0	2.0	3.0	4.0

Tabla 3: Rúbrica para contenido del Informe y demostración

Contenido y demostración		Puntos	Checklist	Estudiante	Profesor
1. GitHub	Hay enlace URL activo del directorio para el laboratorio hacia su repositorio GitHub con código fuente terminado y fácil de revisar.	2	X	2	
2. Commits	Hay capturas de pantalla de los commits más importantes con sus explicaciones detalladas. (El profesor puede preguntar para refrendar calificación).	4	X	3	
3. Código fuente	Hay porciones de código fuente importantes con numeración y explicaciones detalladas de sus funciones.	2	X	2	
4. Ejecución	Se incluyen ejecuciones/pruebas del código fuente explicadas gradualmente.	2	X	2	
5. Pregunta	Se responde con completitud a la pregunta formulada en la tarea. (El profesor puede preguntar para refrendar calificación).	2	X	2	
6. Fechas	Las fechas de modificación del código fuente estan dentro de los plazos de fecha de entrega establecidos.	2	X	2	
7. Ortografía	El documento no muestra errores ortográficos.	2	X	1	
8. Madurez	El Informe muestra de manera general una evolución de la madurez del código fuente, explicaciones puntuales pero precisas y un acabado impecable. (El profesor puede preguntar para refrendar calificación).	4	X	3	
Total		20		17	

8. Referencias

- https://www.w3schools.com/java/default.asp
- https://www.geeksforgeeks.org/insertion-sort/