

UNIVERSIDAD DE LAS FUERZAS ARMADAS - ESPE
UNIDAD DE EDUCACIÓN A DISTANCIA

CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN



Estructuras de Control en JavaScript

Asignatura: PROGRAMACIÓN INTEGRATIVA DE COMPONENTES WEB

NRC: 23407

Nombre: GABRIEL ALEJANDRO MARTÍNEZ CALAHORRANO

Docente: VILMER DAVID CRIOLLO CHANCHICOCHA

Introducción: JavaScript, comúnmente abreviado como JS, es un lenguaje de programación que se utiliza principalmente para el desarrollo web. Los archivos con la extensión .js contienen código JavaScript que puede ejecutarse en navegadores web o en entornos como Node.js. Este lenguaje permite incorporar interactividad, manipular elementos del documento HTML, reaccionar a eventos de usuario, realizar cálculos, validar formularios, crear animaciones, entre otras funcionalidades.

Un archivo .js puede incluir funciones, variables y estructuras de control como condicionales y bucles, además de interactuar con el usuario mediante métodos como prompt para recibir datos y console.log para mostrar resultados en la consola. Para que un archivo .js funcione en una página web, se vincula desde un archivo HTML utilizando la etiqueta `<script src="archivo.js"></script>`.

En las actividades de este trabajo, se utilizan archivos .js externos para resolver problemas básicos mediante estructuras de control. Esto refuerza el aprendizaje de los fundamentos de la programación y permite escribir código que sea organizado, reutilizable y fácil de mantener. Emplear archivos .js externos, en lugar de incorporar el código directamente en HTML, mejora la separación entre el contenido (HTML), el estilo (CSS) y el comportamiento (JS), lo que se considera una buena práctica en el desarrollo web contemporáneo.

Objetivos Generales:

- Practicar if anidados y condiciones múltiples.
- Reforzar el uso de bucles for y operaciones matemáticas.
- Combinar if y for en una sola lógica con entrada del usuario.

Objetivos Personales:

- Dominar el uso de estructuras condicionales if y else if en JavaScript para una toma de decisiones efectivas en el código.
- Emplea bucles for para crear secuencias repetitivas de manera óptima y eficiente.
- Aprender a recibir entradas del usuario utilizando prompt y presentar resultados en la consola con console.log.
- Fomentar una lógica clara y estructurada para abordar problemas que requieren múltiples condiciones y tareas repetitivas.
- Mejorar las habilidades de pensamiento lógico y resolución de problemas a través de la práctica de la programación.

LINK DEL REPOSITORIO: <https://github.com/Gabo-prog/Estructuras-de-Control-en-JavaScript/tree/main/Estructuras%20de%20Control%20en%20JavaScript>

Desarrollo:

HTML

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Actividades JavaScript</title>
6   <link rel="stylesheet" href="styles.css">
7 </head>
8 <body>
9   <!-- Título principal de la página -->
10  <h1>Actividades de JavaScript</h1>
11  <!-- Contenedores para mostrar resultados de cada actividad -->
12  <div id="resultadoEdad"></div>
13  <div id="resultadoTabla"></div>
14  <div id="resultadoTablas"></div>
15
16  <!-- Inclusión de scripts JavaScript externos -->
17  <script src="edadCategoria.js"></script>
18  <script src="tablaMultiplicar.js"></script>
19  <script src="tablasSeleccionadas.js"></script>
20
21  <!-- Llamada inicial para ejecutar la primera actividad -->
22  <script>
23    actividadEdad();
24  </script>
25 </body>
26 </html>
```

CSS

```
1  /* Estilos generales del cuerpo del documento */
2  body {
3    font-family: Arial, sans-serif;
4    background-color: #f0f0f0;
5    margin: 0;
6    padding: 20px;
7    color: #333;
8  }
9
10 /* Estilo para el título principal */
11 h1 {
12   text-align: center;
13   color: #2c3e50;
14 }
15
16 /* Estilo para subtítulos */
17 h2 {
18   color: #34495e;
19 }
20
21 /* Estilos para las secciones de resultados */
22 #resultadoEdad,
23 #resultadoTabla,
24 #resultadoTablas {
25   background-color: #ffffff;
26   border: 1px solid #ddd;
27   padding: 15px;
28   margin: 20px auto;
29   border-radius: 10px;
30   box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);
31   max-width: 600px;
32 }
```

```

33
34  /* Estilo para texto resaltado */
35  strong {
36      color: #2980b9;
37  }
38
39  /* Estilo para todos los divs */
40  div {
41      line-height: 1.6;
42  }

```

Actividad 1: IF anidados - Clasificación de edades

Enunciado:

Crea un programa en un archivo .js externo que le pida al usuario (usando prompt) su edad y muestre en la consola una categoría según la siguiente tabla:

- Menor de 0: "Edad no válida"
- 0 a 12 años: "Niño"
- 13 a 17 años: "Adolescente"
- 18 a 59 años: "Adulto"
- 60 años o más: "Adulto mayor"

Archivo sugerido: edadCategoria.js

```

1  // Función para solicitar edad y clasificar en una categoría
2  function actividadEdad() {
3      let edad = parseInt(prompt("Actividad 1 - Ingresar tu edad:"));
4      let resultado = "";
5
6      // Validación y clasificación según edad ingresada
7      if (isNaN(edad)) {
8          resultado = "Por favor, ingresa un número válido.";
9      } else if (edad < 0) {
10         resultado = "Edad no válida";
11     } else if (edad <= 12) {
12         resultado = "Niño";
13     } else if (edad <= 17) {
14         resultado = "Adolescente";
15     } else if (edad <= 59) {
16         resultado = "Adulto";
17     } else {
18         resultado = "Adulto mayor";
19     }
20
21     const mensaje = "Edad ingresada: " + edad + " | Categoría: " + resultado;
22     console.log("%cActividad 1: Clasificación de Edad", "color: blue; font-weight: bold;");
23     console.log(mensaje);
24
25     // Mostrar resultado en el HTML
26     document.getElementById("resultadoEdad").innerHTML =
27         `<h2>Actividad 1: Clasificación de Edad</h2>
28         <p><strong>Edad ingresada:</strong> ${edad}</p>
29         <p><strong>Categoría:</strong> ${resultado}</p>`;
30
31     // Ejecuta la siguiente actividad
32     actividadTabla();
33 }

```

Actividad 2: FOR - Tabla de multiplicar

Enunciado:

Crea un programa en un archivo .js externo que muestre la tabla de multiplicar del número que el usuario escriba, desde el 1 hasta el 12. Usa un bucle for.

Ejemplo de salida si el usuario escribe 5:

5 x 1 = 5

5 x 2 = 10

... ..

5 x 12 = 60

Archivo sugerido: tablaMultiplicar.js

```
1 // Función para mostrar la tabla de multiplicar de un número ingresado
2 function actividadTabla() {
3     let numero = parseInt(prompt("Actividad 2 - Ingresa un número para su tabla de multiplicar:"));
4     let contenido = "<h2>Actividad 2: Tabla de Multiplicar</h2>";
5
6     console.log("%cActividad 2: Tabla de Multiplicar", "color: green; font-weight: bold;");
7     if (isNaN(numero)) {
8         contenido += "<p>Por favor, ingresa un número válido.</p>";
9         console.log("Número inválido ingresado.");
10    } else {
11        contenido += `<p><strong>Tabla del ${numero}</strong></p>`;
12        for (let i = 1; i <= 12; i++) {
13            let linea = `${numero} x ${i} = ${numero * i}`;
14            contenido += `<p>${linea}</p>`;
15            console.log(linea);
16        }
17    }
18
19    // Mostrar resultado en el HTML
20    document.getElementById("resultadoTabla").innerHTML = contenido;
21
22    // Ejecuta la siguiente actividad
23    actividadTablas();
24 }
```

Actividad 3: IF + FOR - Tablas pares o impares

Enunciado:

Crea un programa en un archivo .js externo que le pida al usuario si quiere ver las tablas de multiplicar de números pares o impares entre 1 y 10. Usa if para decidir qué tipo mostrar, y for para imprimir las tablas del 1 al 10 según lo que el usuario elija.

Ejemplo:

- Si el usuario escribe "pares", mostrar tablas del 2, 4, 6, 8, 10.
- Si el usuario escribe "impares", mostrar tablas del 1, 3, 5, 7, 9.

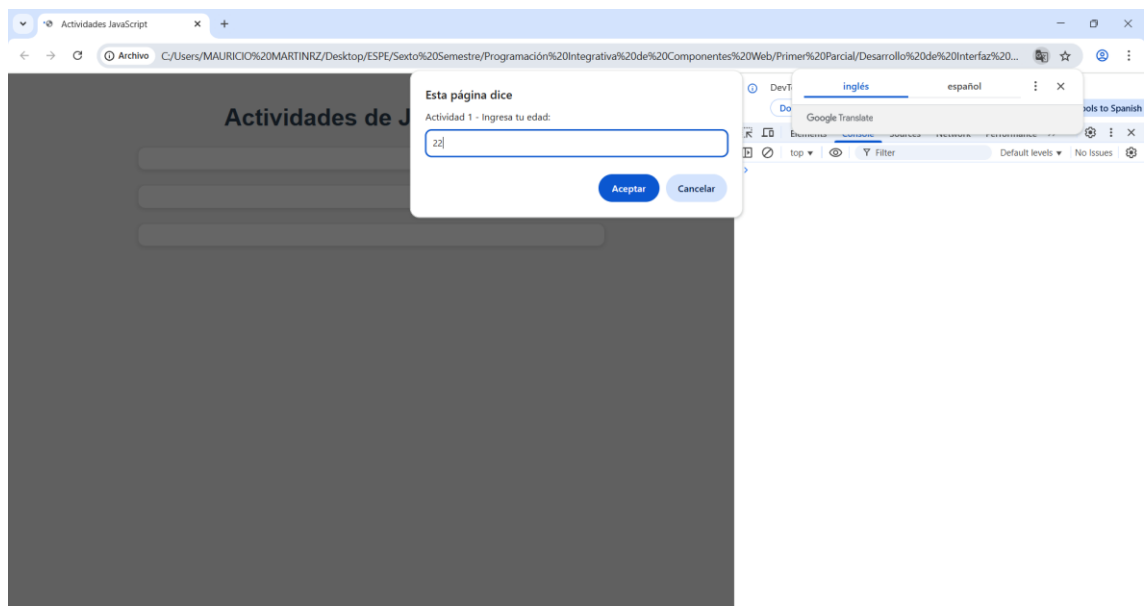
Archivo sugerido: tablasSeleccionadas.js

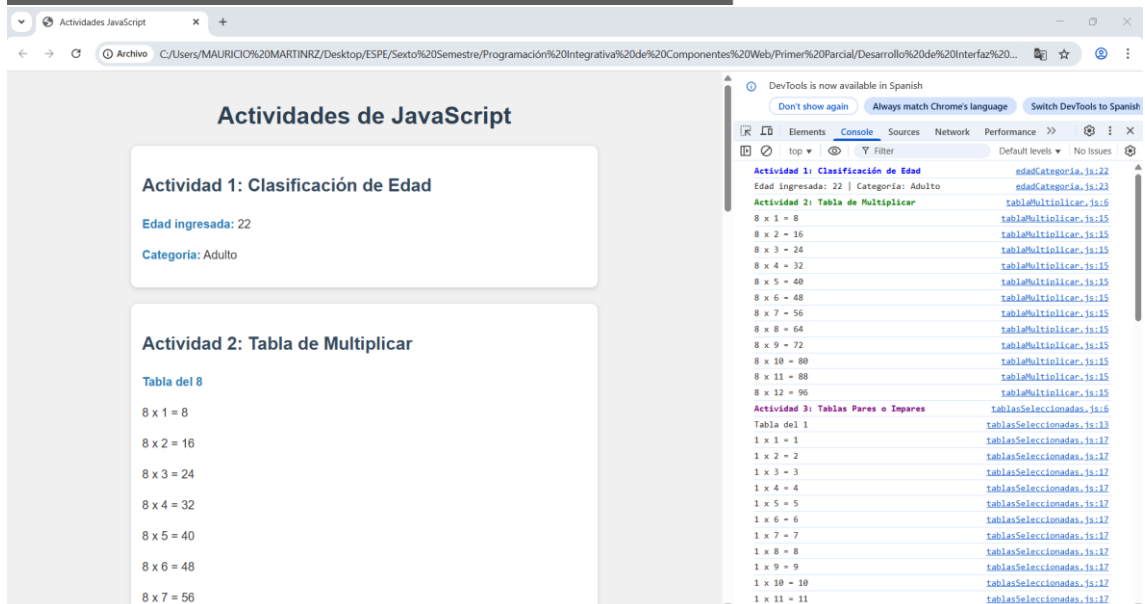
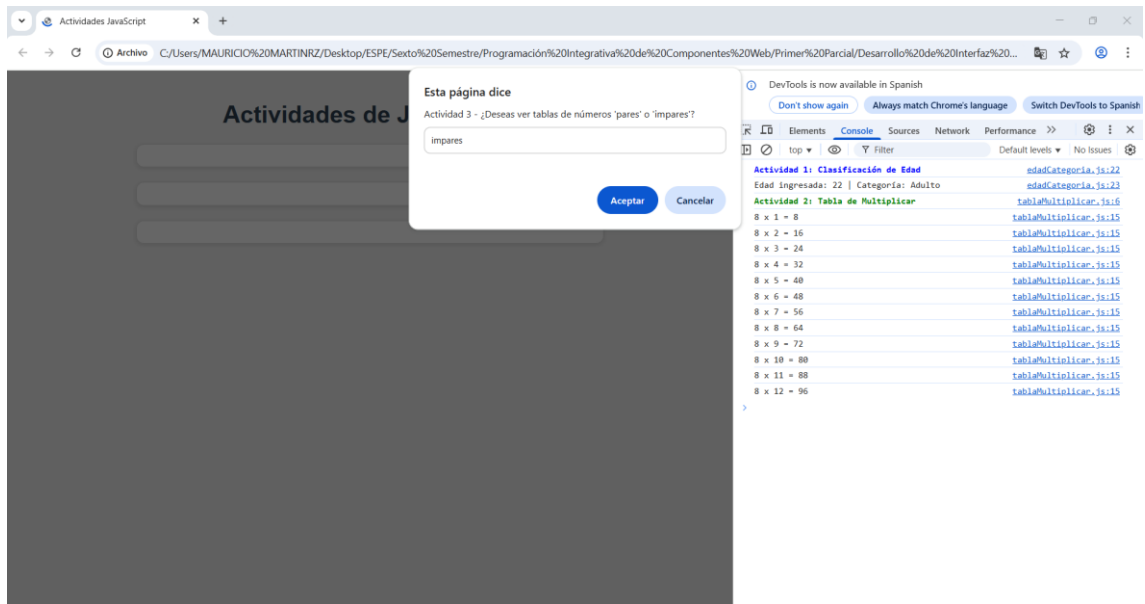
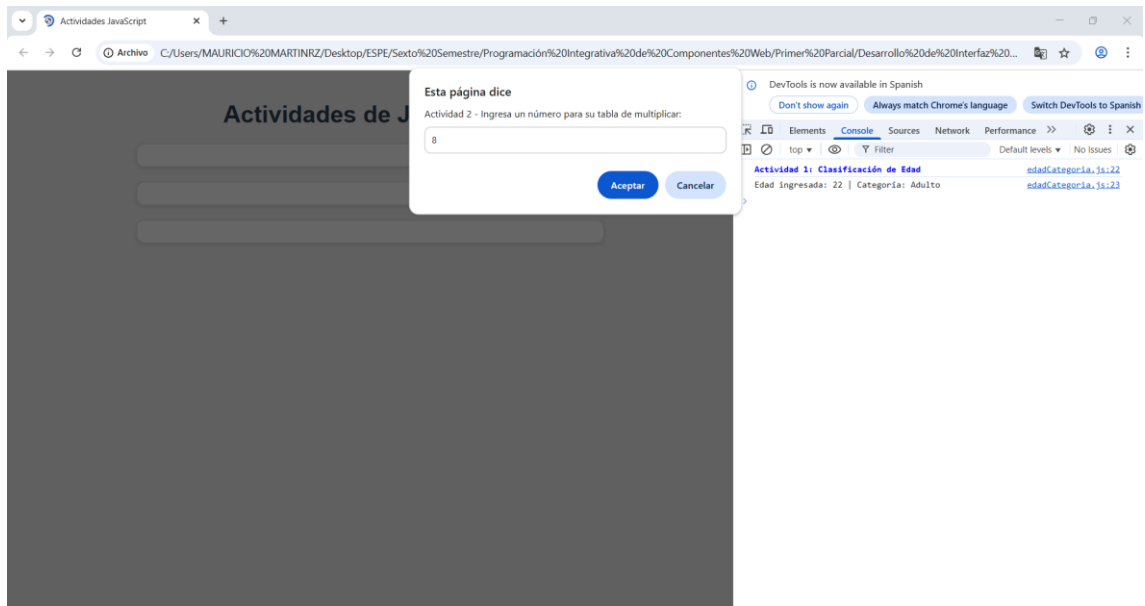
```

1 // Función para mostrar tablas de multiplicar de números pares o impares
2 function actividadTablas() {
3     let tipo = prompt("Actividad 3 - ¿Deseas ver tablas de números 'pares' o 'impares'?").toLowerCase();
4     let contenido = "<h2>Actividad 3: Tablas Pares o Impares</h2>";
5
6     console.log("%cActividad 3: Tablas Pares o Impares", "color: purple; font-weight: bold;");
7     if (tipo === "pares" || tipo === "impares") {
8         let esPar = tipo === "pares";
9         contenido += `<p><strong>Mostrando tablas de números ${tipo}</strong></p>`;
10        for (let i = 1; i <= 10; i++) {
11            if ((esPar && i % 2 === 0) || (!esPar && i % 2 !== 0)) {
12                contenido += `<div><strong>Tabla del ${i}</strong><br>`;
13                console.log(`Tabla del ${i}`);
14                for (let j = 1; j <= 12; j++) {
15                    let linea = `${i} x ${j} = ${i * j}`;
16                    contenido += `${linea}<br>`;
17                    console.log(linea);
18                }
19                contenido += `</div><br>`;
20            }
21        }
22    } else {
23        contenido += "<p>Opción no válida. Escribe 'pares' o 'impares'.</p>";
24        console.log("Opción no válida ingresada.");
25    }
26
27    // Mostrar resultado en el HTML
28    document.getElementById("resultadoTablas").innerHTML = contenido;
29 }

```

Código Ejecutado





Nota: Las actividades se pueden visualizar tanto en pantalla como en consola.

Conclusiones:

Las actividades propuestas facilitan la aplicación de conceptos clave de JavaScript, estableciendo una base sólida para el aprendizaje continuo en programación. A través de los condicionales, se pueden realizar decisiones automatizadas, mientras que los bucles permiten iterar eficientemente sobre grupos de valores. Además, la interacción con el usuario mediante prompt y la visualización de resultados con console.log brindar una experiencia práctica que refleja situaciones reales de desarrollo. El dominio de estas herramientas y conceptos será esencial para futuros proyectos más complejos, por lo que dominar estas estructuras básicas es un paso crucial en el proceso de formación de un programador.

Bibliografía:

for - JavaScript. (n.d.). MDN Web Docs. Retrieved May 21, 2025, from <https://developer.mozilla.org/es/docs/Web/JavaScript/Reference/Statements/for>

Haverbeke, M. (2018). *Eloquent JavaScript, 3rd edition: A modern introduction to programming*. No Starch Press.

if...else. (n.d.). MDN Web Docs. Retrieved May 21, 2025, from <https://developer.mozilla.org/es/docs/Web/JavaScript/Reference/Statements/if...else>

JavaScript for loop. (n.d.). W3schools.com. Retrieved May 21, 2025, from https://www.w3schools.com/js/js_loop_for.asp

JavaScript if else else if. (n.d.). W3schools.com. Retrieved May 21, 2025, from https://www.w3schools.com/js/js_if_else.asp

Window.prompt(). (n.d.). MDN Web Docs. Retrieved May 21, 2025, from <https://developer.mozilla.org/es/docs/Web/API/Window/prompt>