

[INICIO](#)[CARACTERÍSTICAS](#)[EL PROYECTO](#)[PLANES](#)[DESCARGAR](#)

IFC154_3 DESARROLLO DE APLICACIONES CON TECNOLOGÍAS WEB

GRUPO MAINJOBS

26/06/2024



[INICIO](#)[CARACTERÍSTICAS](#)[EL PROYECTO](#)[PLANES](#)[DESCARGAR](#)

UF 1842: DESARROLLO Y REUTILIZACIÓN DE COMPONENTES SOFTWARE Y MULTIMEDIA MEDIANTE LENGUAJES DE GUIÓN

GRUPO MAINJOBS

26/06/2024



Temario

(IFCD0210) DESARROLLO DE APLICACIONES CON TECNOLOGÍAS WEB (RD 1531/2011, de 31 de octubre modificado por el RD 628/2013, de 2 de agosto)	
MF0491_3: Programación web en el entorno cliente	UF1841: Elaboración de documentos web mediante lenguajes de marca
	UF1842: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión
	UF1843: Aplicaciones técnicas de usabilidad y accesibilidad en el entorno cliente
MF0492_3: Programación web en el entorno servidor	UF1844: Desarrollo de aplicaciones web en el entorno servidor.
	UF1845: Acceso a datos en aplicaciones web del entorno servidor.
	UF1846: Desarrollo de aplicaciones web distribuidas.
MF0493_3: Implantación de aplicaciones web en entorno internet, intranet y extranet	
MP0391: Módulo de prácticas profesionales no laborales	

Módulo formativo 1

PROGRAMACIÓN WEB EN EL ENTORNO CLIENTE

- **Unidad 2:** Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión (90 horas)

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

4. Lenguajes de guión general.

4.6 Listas arrays.



Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

4. Lenguajes de guión general.

4.6 Listas arrays.

EJERCICIO IMPORTANTE



Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

4. Lenguajes de guión general.

4.6 Listas arrays.

SOLUCIÓN



Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

4. Lenguajes de guión general.

4.6 Listas arrays.

EJERCICIO IMPORTANTE

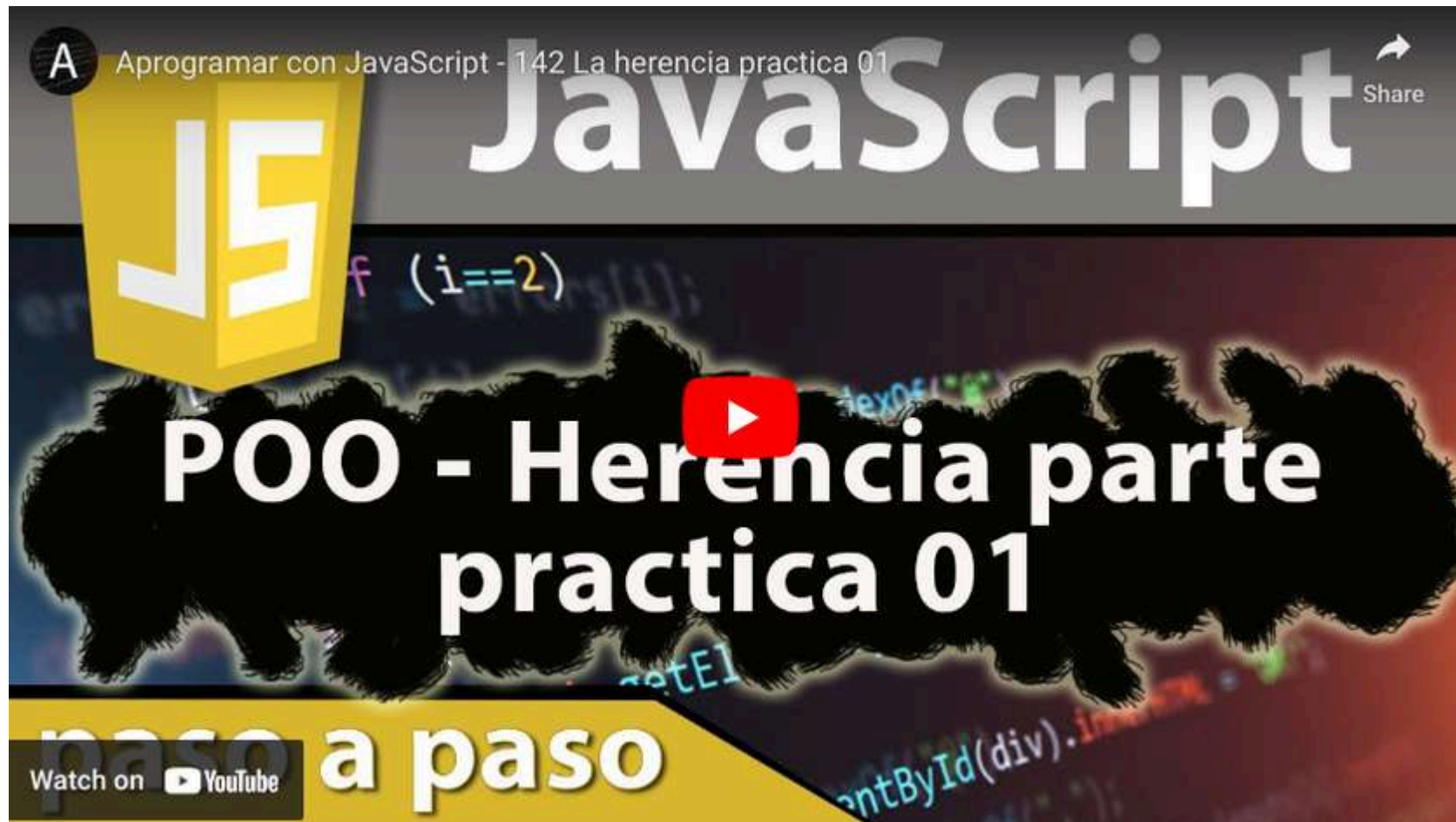


Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

4. Lenguajes de guión general.

4.6 Listas arrays.

La herencia practica 01

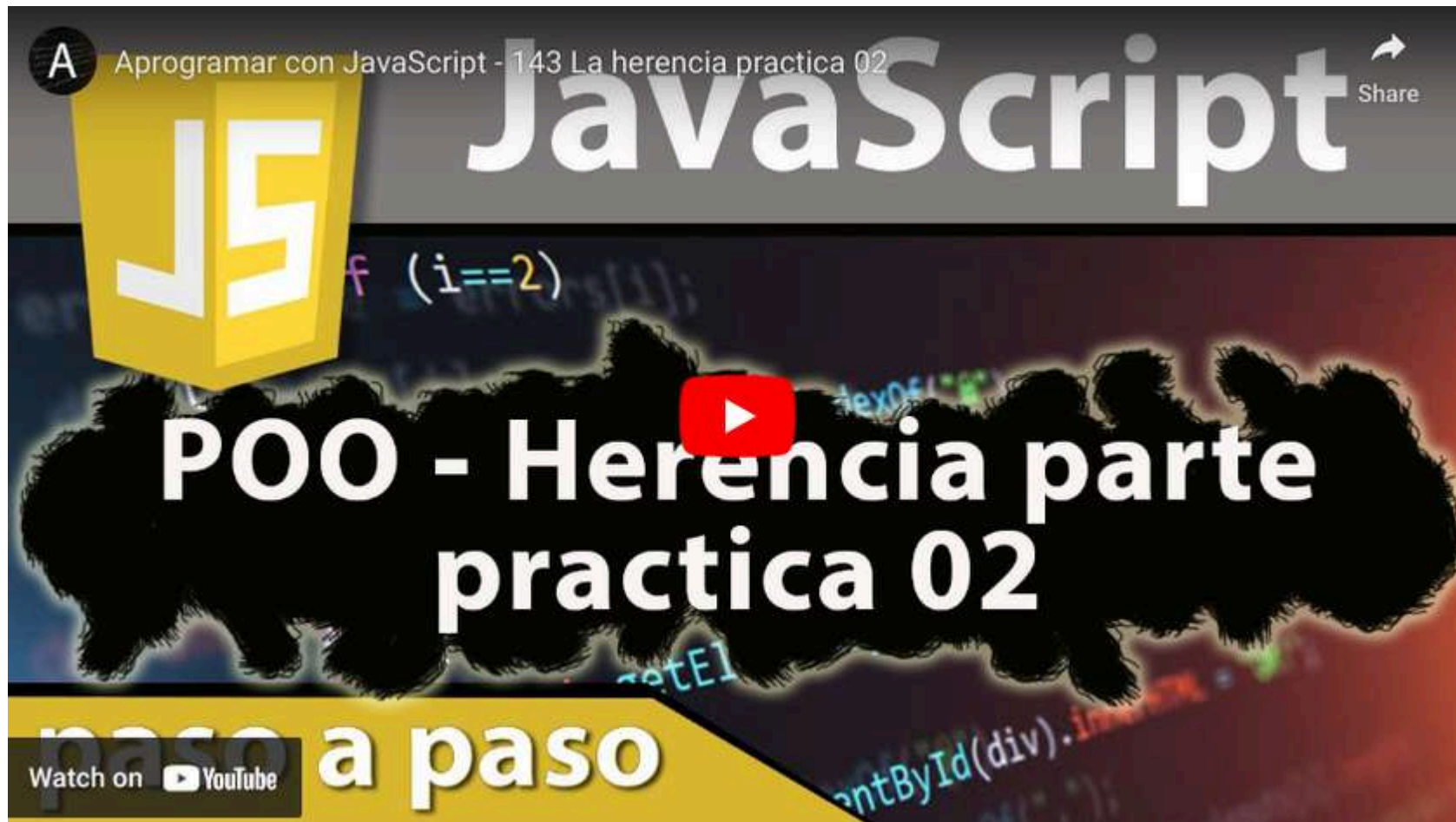


Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

4. Lenguajes de gui3n general.

4.6 Listas arrays.

La herencia practica 02



Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

4. Lenguajes de guión general.

Gestión de eventos.

Evento



Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

4. Lenguajes de guión general.

Gestión de eventos.

Tipos de eventos



Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

4. Lenguajes de gui3n general.

Gesti3n de eventos.

Eventos de carga inicial



Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

4. Lenguajes de guión general.

Gestión de eventos.

Manejo de eventos



Tutorial DOM Javascript | Eventos con addEventListener [2021]

Share

EVENTOS DEL DOM

CLICK, CHANGE Y MOUSE



```
addEventListener("click", () => {  
  console.log("click");  
});
```



Watch on  YouTube

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

4. Lenguajes de gui3n general.

Gesti3n de eventos.

Manejo de eventos

P3gina 203 a 211 no se estudia

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

4. Lenguajes de guión general.

Animaciones.



Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

4. Lenguajes de guión general.

Efectos especiales en elementos gráficos y texto.



Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

4. Lenguajes de guión general.

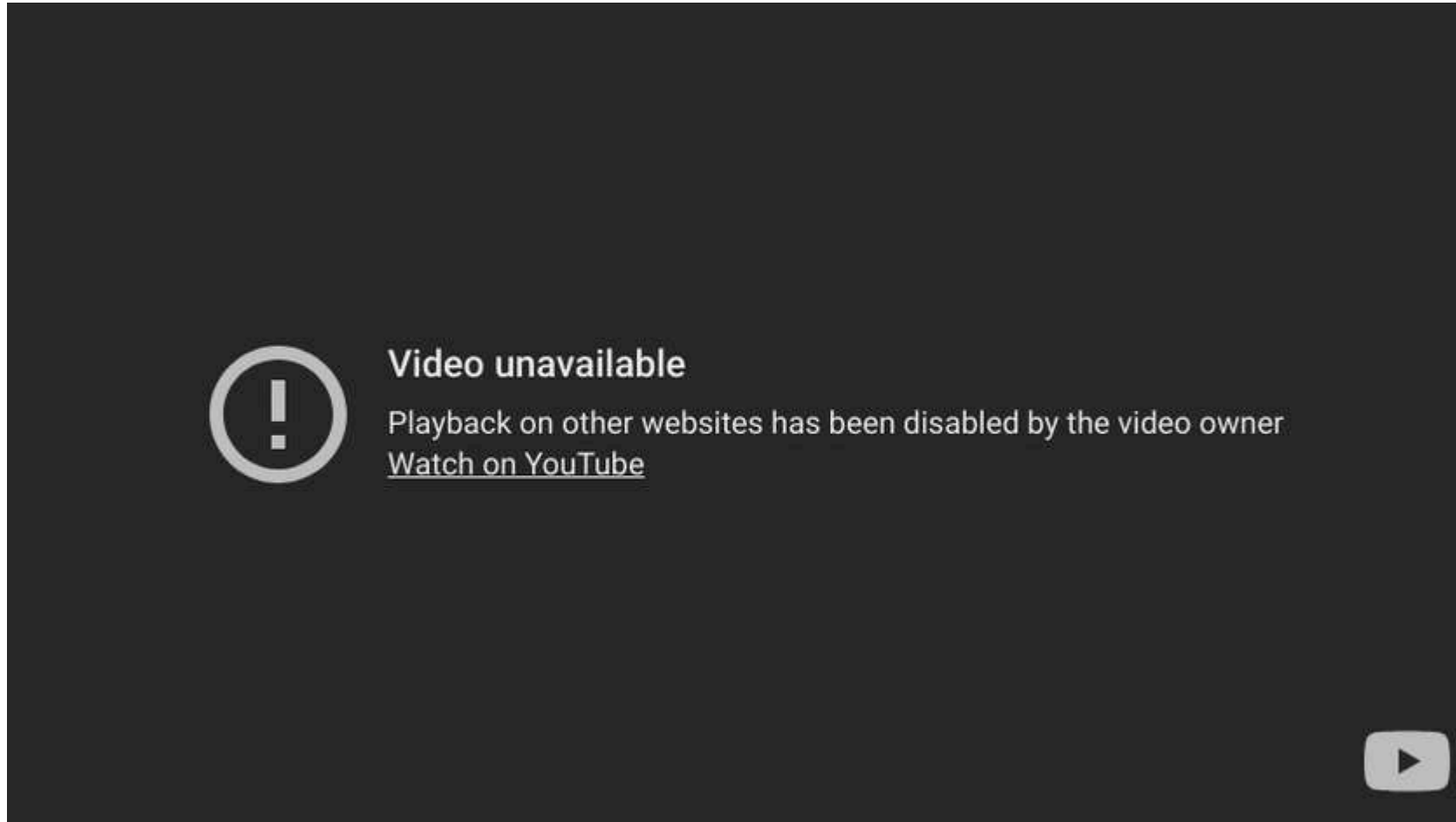
insertAdjacent para insertar elementos, html o nodos al DOM



Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

4. Lenguajes de guión general.

•



Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

4. Lenguajes de guión general.

- Descripción de las técnicas de acceso asíncrono (AJAX)

Las aplicaciones Ajax añaden una capa entre el cliente y el servidor para gestionar la comunicación entre ambos. Las solicitudes son asíncronas, de modo que el usuario puede seguir interactuando con la aplicación

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

4. Lenguajes de guión general.

- Descripción de las técnicas de acceso asíncrono (AJAX)

Las **aplicaciones más conocidas** que utilizan tecnología AJAX:

Gestores de correo electrónico: Gmail, Outlook....

Aplicaciones web: Google Drive....

Aplicaciones de mapa: Google maps.....

Otras aplicaciones: Twitter, Facebook....

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

4. Lenguajes de guión general.

- Descripción de las técnicas de acceso asíncrono (AJAX)

Las **aplicaciones más conocidas** que utilizan tecnología AJAX:

Gestores de correo electrónico: Gmail, Outlook....

Aplicaciones web: Google Drive....

Aplicaciones de mapa: Google maps.....

Otras aplicaciones: Twitter, Facebook....

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

4. Lenguajes de guión general.

- Descripción de las técnicas de acceso asíncrono (AJAX)



Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

4. Lenguajes de guión general.

- Utilidades adicionales para la realización de contenidos dinámicos.



Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

4. Lenguajes de guión general.

- Utilidades adicionales para la realización de contenidos dinámicos.

Joomla y WordPress, juntos y revueltos

Joomla! Share

Novedades Joomla! 3.0

- Nuevo instalador
- Twitter Bootstrap – Responsive “de fábrica”
- Nueva interfaz de usuario con drag & drop
- Grandes mejoras en usabilidad
- PostgreSQL
- Además de lo ya incluido en la 2.5:
 - Actualizaciones en 1 click
 - Multiidioma
 - SQL Server
 - Búsqueda avanzada

Watch on YouTube

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

4. Lenguajes de guión general.

- Entorno de desarrollo específicos: desarrollo sobre dispositivos móviles.



Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

5. Contenidos multimedia.

. Definición de multimedia. Tipos de recursos

Se aplica a cualquier objeto o sistema que utilice múltiples medios físicos o digitales para mostrar sus contenidos. Pueden ser muy variados , desde imágenes, animación, sonido, entre otros.

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

5. Contenidos multimedia.

- Inclusión para contenido de multimedia en páginas web.

Adaptadores para recursos multimedia



Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

5. Contenidos multimedia.

- *Inclusión para contenido de multimedia en páginas web.*

Adaptadores para recursos multimedia

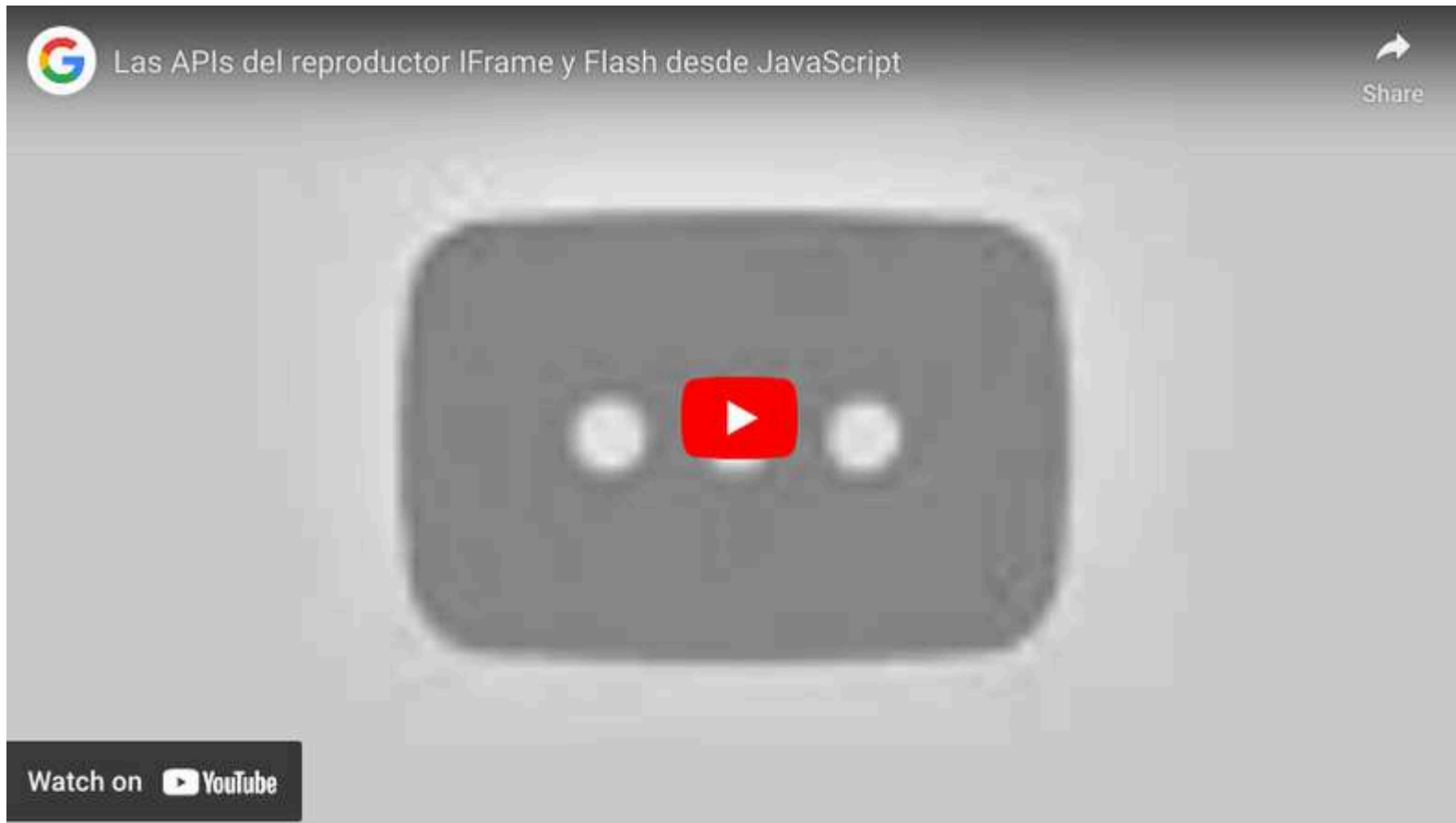


Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

5. Contenidos multimedia.

- *Inclusión para contenido de multimedia en páginas web.*

Las APIs del reproductor IFrame y Flash desde JavaScript

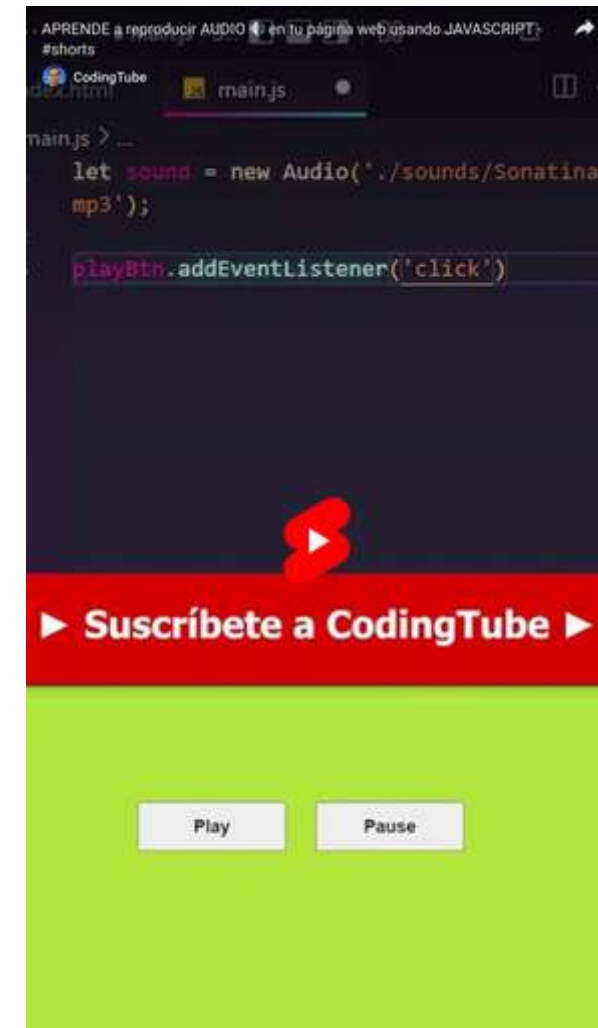


Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

5. Contenidos multimedia.

- Inclusión para contenido de multimedia en páginas web.

Reproductores de audio en JavaScript

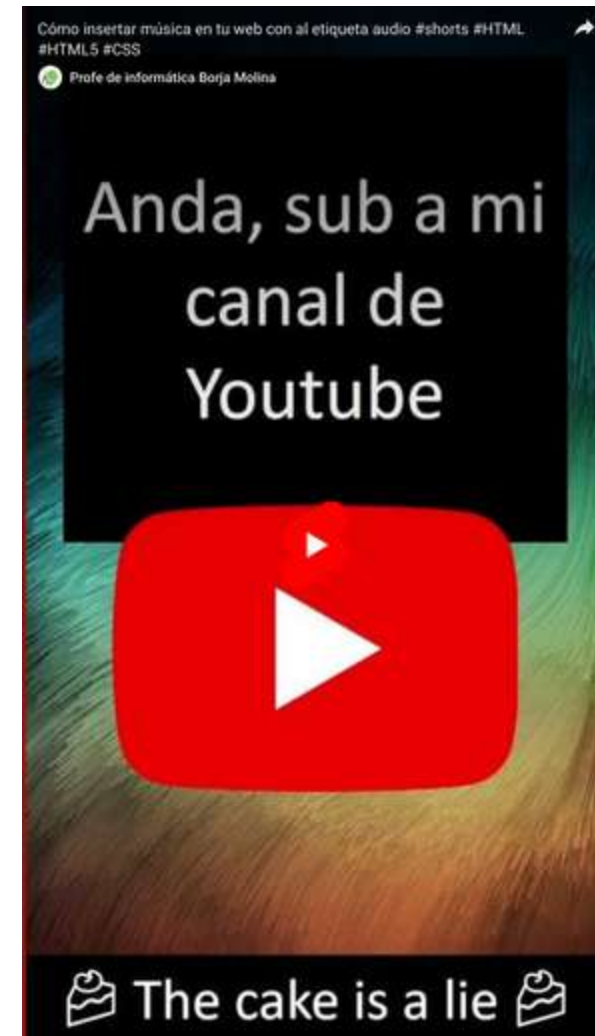


Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

5. Contenidos multimedia.

- *Inclusión para contenido de multimedia en páginas web.*

Como insertar música en JavaScript



Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

5. Contenidos multimedia.



JAVASCRIPT.INFO
The Modern JavaScript Tutorial

Animaciones JavaScript

Queremos que este proyecto de c3digo abierto est3 disponible para personas de todo el mundo.

 iliakan

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

Ejercicio 1.

Complete las siguientes oraciones:

Un script o PROGRAMA de JAVASCRIPT en el lado del cliente es un programa que puede acompañar a un documento HTML mediante un ARCHIVO externo, o incluso puede estar INCRUSTADO en el propio HTML. Este SCRIPT se ejecuta en la máquina del cliente cuando se carga el documento WEB

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

PREGUNTAS CORTAS

Ejercicio 2.

De las siguientes oraciones, ¿cuál es verdadera y cuál es falsa?

a. El lenguaje de programación principal para el código por parte del cliente es ActionScript.

Verdadero

Falso.

b. Javascript no es un lenguaje orientado a objetos.

Verdadero

Falso

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

PREGUNTAS CORTAS

Ejercicio 4.

¿C3mo se declara una variable en Javascript?

- Es preferible usar `let` y `const` en lugar de `var` debido a su alcance de bloque y a que ayudan a evitar errores relacionados con el alcance de las variables.
- Usa `const` siempre que sepas que el valor de la variable no cambiar3, y `let` cuando necesites que la variable pueda cambiar su valor.

Ejercicio 5.

a. El objeto de `math` permite realizar c3lculos matem3ticos.

Verdadero

Falso

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

PREGUNTAS CORTAS

Ejercicio 5.

b. El objeto document proporciona m3todos para manipular las ventanas del navegador.

Verdadero

Falso

c. El objeto window para manipular el documento actualmente visible en la ventana del navegador.

Verdadero

Falso

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

Ejercicio 8

Completa la siguiente oración:

Los arrays son __ESTRUCTURAS_____ de datos, que se encuentran formadas por __COLECCIONES_____ de datos relacionados. Permiten guardar varias __VALORES_____, y acceder a ellas de forma __ORDENADA_____. Para hacer referencia a un __ELEMENTO_____ concreto de un array, se especifica el __NOMBRE_____ del mismo y el __ÍNDICE_____ de posición del __ELEMENTO_____.

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

Ejercicio 9.

Los array multidimensionales....

- a. ... no existen.
- b. ... son navegadores web.
- c. ... pueden tener dos o más subíndices.
- d. ... son como una variable.

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

CAPÍTULO 5

Ejercicio 1 p3gina 331

De las siguientes frases, ¿cu3al es verdadera y cu3al es falsa?

a) Multimedia solamente utiliza un medio a la vez.

Verdadera

Falsa

b) Los gr3ficos no son un formato de informaci3n multimedia

Falso

Verdadera

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

Ejercicio 1 p3gina 331

c) En una p3gina web se pueden a1ñadir contenidos multimedia

Falso

Verdadera

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

Ejercicio 2 p3gina 331

¿Para que se usa principalmente la reproducci3n de streaming?

La reproducci3n de streaming se utiliza para transmitir contenido multimedia en directo por internet como pueden ser Netflix, Youtube, Amazon Prime entre otros.

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

Ejercicio 3_p3gina 331

¿Cu3l es verdadera y cu3l es falsa, en relaci3n a la reproducci3n progresiva?

a) Se realiza a trav3s del protocolo HTTP

Verdadero

Falso

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

Ejercicio 3_p3gina 331

¿Cu3l es verdadera y cu3l es falsa, en relaci3n a la reproducci3n progresiva?

b) Es posible ver cualquier parte del contenido multimedia antes de que se descargue por completo.

Verdadero

Falso

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

Ejercicio 3_p3gina 331

¿Cu3l es verdadera y cu3l es falsa, en relaci3n a la reproducci3n progresiva ?

c) Se puede consumir mucho ancho de banda .

Verdadero

Falso

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

Ejercicio 10 p3gina 333.

De las siguientes oraciones, ¿cu3l es verdadera y cu3l es falsa?

a) Existen tres grupos principales de formatos de audio .

Verdadero

Falso

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

Ejercicio 10 página 333.

De las siguientes oraciones, ¿cuál es verdadera y cuál es falsa?

b) El formato audio sin comprimir requiere de menos tiempo de procesamiento .

Verdadero

Falso

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

Ejercicio 10 p3gina 333.

De las siguientes oraciones, ¿cu3l es verdadera y cu3l es falsa?

c) En la compresi3n con p3rdida de audio, se comprimen los datos descart3ndose parte de estos.

Verdadero

Falso

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

Ejercicio 1.

Desarrolla un programa que determine el costo total de un viaje, considerando los gastos de alojamiento, comida y entretenimiento.

Solicita al usuario que introduzca los costos estimados para cada una de estas categorías. Calcula el costo total sumando todos estos gastos. Muestra el costo total al usuario. Ejemplo de entrada: Gastos de alojamiento: 200€, Gastos de comida: 150€, Gastos de entretenimiento: 100€. Salida esperada: El costo total del viaje es 450€.

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
1. let alojamiento = parseFloat(prompt("Ingrese el coste estimado de alojamiento: "));
2. let alimentacion = parseFloat(prompt("Ingrese el coste estimado de alimentación: "));
3. let entretenimiento = parseFloat(prompt("Ingrese el coste estimado de entretenimiento: "));
4.
5. let costeTotal = alojamiento + alimentacion + entretenimiento;
6. alert("El coste total del viaje es: " + costeTotal.toFixed(2) + "€");
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
// Función para solicitar y leer los costos del usuario
function obtenerCosto(categoria) {
    let costo = parseFloat(prompt(`Introduce los gastos de ${categoria}:`));
    while (isNaN(costo) || costo < 0) {
        alert("Por favor, introduce un número válido.");
        costo = parseFloat(prompt(`Introduce los gastos de ${categoria}:`));
    }
    return costo;
}

// Solicitar los costos al usuario
let costoAlojamiento = obtenerCosto("alojamiento");
let costoComida = obtenerCosto("comida");
let costoEntretenimiento = obtenerCosto("entretenimiento");

// Calcular el costo total
let costoTotal = costoAlojamiento + costoComida + costoEntretenimiento;

// Mostrar el costo total al usuario
alert(`El costo total del viaje es ${costoTotal}€`);
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

Calculadora de Costo Total del Viaje

Gastos de alojamiento (€):

Gastos de comida (€):

Gastos de entretenimiento (€):

Calcular Costo Total

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

PREGUNTAS CORTAS

RELACI3N DE EJERCICIOS

Ejercicio 2

Desarrolla un programa que convierta la edad de un perro a a3os equivalentes humanos. Solicita al usuario que ingrese la edad de su perro en a3os. Calcula la edad del perro en a3os humanos multiplicando su edad por 7. Muestra al usuario la edad del perro en a3os humanos. Ejemplo de entrada:

Edad del perro: 5 a3os. Salida esperada: La edad del perro en a3os humanos es 35 a3os.

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
1. let edadPerro = parseInt(prompt("Indica la edad de tu perro en años: "));  
2. let edadHumana = edadPerro * 7;  
3. console.log("La edad de tu perro en años humanos es: " + edadHumana + " años");
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
// Función para solicitar y leer la edad del perro del usuario
function obtenerEdadPerro() {
    let edadPerro = parseFloat(prompt("Introduce la edad de tu perro en años:"));
    while (isNaN(edadPerro) || edadPerro < 0) {
        alert("Por favor, introduce un número válido.");
        edadPerro = parseFloat(prompt("Introduce la edad de tu perro en años:"));
    }
    return edadPerro;
}

// Solicitar la edad del perro al usuario
let edadPerro = obtenerEdadPerro();

// Calcular la edad del perro en años humanos
let edadHumana = edadPerro * 7;

// Mostrar la edad del perro en años humanos al usuario
alert(`La edad del perro en años humanos es ${edadHumana} años.`);
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

Conversor de Edad de Perro a Años Humanos

Edad del perro (años):

Convertir Edad

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

PREGUNTAS CORTAS

RELACI3N DE EJERCICIOS

Ejercicio 3.

Desarrolla un programa que determine el IMC (3ndice de Masa Corporal) de una persona bas3ndose en su peso y altura.

Pide al usuario que introduzca su peso en kilogramos y su altura en metros. Calcula el IMC dividiendo el peso por el cuadrado de la altura. Muestra al usuario el IMC calculado y su clasificaci3n correspondiente

seg3n la tabla de IMC. Ejemplo de entrada: Peso: 70kg, Altura: 1.75m. Salida esperada: IMC: 22.86,
Clasificaci3n: Normal.

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

El Índice de Masa Corporal (IMC) se calcula dividiendo el peso de una persona en kilogramos entre la altura en metros al cuadrado. El IMC proporciona una estimación de la grasa corporal de una persona y se utiliza comúnmente como un indicador de salud. Los rangos típicos de IMC son: Menor de 18.5: Bajo peso, 18.5 – 24.9: Peso normal, 25 – 29.9: Sobrepeso y 30 o más: Obesidad.

Solución:

```
1. let peso = parseFloat(prompt("Inserta tu peso en kilogramos: "));
2. let altura = parseFloat(prompt("Inserta tu altura en metros: "));
3.
4. let imc = peso / (altura * altura);
5. console.log("Su IMC es: " + imc.toFixed(2));
6.
7. if (imc < 18.5) {
8.     console.log("Clasificación: Bajo peso");
9. } else if (imc >= 18.5 && imc < 25) {
10.    console.log("Clasificación: Peso normal");
11. } else if (imc >= 25 && imc < 30) {
12.    console.log("Clasificación: Sobrepeso");
13. } else {
14.    console.log("Clasificación: Obesidad");
15. }
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
// Función para solicitar y leer el peso del usuario
function obtenerPeso() {
    let peso = parseFloat(prompt("Introduce tu peso en kilogramos:"));
    while (isNaN(peso) || peso <= 0) {
        alert("Por favor, introduce un número válido.");
        peso = parseFloat(prompt("Introduce tu peso en kilogramos:"));
    }
    return peso;
}

// Función para solicitar y leer la altura del usuario
function obtenerAltura() {
    let altura = parseFloat(prompt("Introduce tu altura en metros:"));
    while (isNaN(altura) || altura <= 0) {
        alert("Por favor, introduce un número válido.");
        altura = parseFloat(prompt("Introduce tu altura en metros:"));
    }
    return altura;
}

// Función para calcular el IMC
function calcularIMC(peso, altura) {
    return peso / (altura * altura);
}
```


Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
// Función para determinar la clasificación del IMC
function clasificarIMC(imc) {
  if (imc < 18.5) {
    return "Bajo peso";
  } else if (imc >= 18.5 && imc < 24.9) {
    return "Normal";
  } else if (imc >= 25 && imc < 29.9) {
    return "Sobrepeso";
  } else {
    return "Obesidad";
  }
}

// Solicitar el peso y la altura al usuario
let peso = obtenerPeso();
let altura = obtenerAltura();

// Calcular el IMC
let imc = calcularIMC(peso, altura);

// Determinar la clasificación del IMC
let clasificacion = clasificarIMC(imc);

// Mostrar el IMC y la clasificación al usuario
alert(`IMC: ${imc.toFixed(2)}, Clasificación: ${clasificacion}`);
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

Calculadora de IMC

Peso (kg):

Altura (m):

Calcular IMC

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

PREGUNTAS CORTAS

RELACI3N DE EJERCICIOS

Ejercicios con Operadores Aritm3ticos, Relacionales y L3gicos

Ejercicio 4.

Si $a = 33$, determinar si la siguiente operaci3n es VERDADERA o FALSA

1. $\text{NOT}((a > 10) \text{ AND } (a < 20))$
2. $\text{NOT}((a > 10) \text{ OR NOT } (a < 20))$

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
let a = 33;  
  
console.log("Expresión 1:", !((a > 10) && (a < 20)));  
console.log("Expresión 2:", !((a > 10) || !(a < 20)));
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
let a = 33;

// Evaluar la primera expresión
let expresion1 = !( (a > 10) && (a < 20) );
console.log(`La primera expresión es ${expresion1 ? 'VERDADERA' : 'FALSA'}`);

// Evaluar la segunda expresión
let expresion2 = !( (a > 10) || !(a < 20) );
console.log(`La segunda expresión es ${expresion2 ? 'VERDADERA' : 'FALSA'}`);
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

Ejercicio 4: Valor de verdad de expresiones lógicas

Resultado 1: true

Resultado 2: false

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

PREGUNTAS CORTAS

RELACI3N DE EJERCICIOS

Ejercicios con Operadores Aritm3ticos, Relacionales y L3gicos

Ejercicio 5.

Si $a = 20$ $b = a$ $c = 15$ $d = 10$

Determinar si la siguiente operaci3n es VERDADERA o FALSA

3. $((a = b) \text{ OR } (b > c)) \text{ OR } (c < d)$

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

Si $a = 20$ $b = a$ $c = 15$ $d = 10$

Determinar si la siguiente operación es VERDADERA o FALSA

3. $((a = b) \text{ OR } (b > c)) \text{ OR } (c < d)$

// Si $a = 20$, $b = a$, $c = 15$, $d = 10$

let $a = 20$;

let $b = a$;

let $c = 15$;

let $d = 10$;

// Operación 3

let operacion3 = $((a == b) || (b > c)) || (c < d)$;

console.log("Operación 3: ", operacion3); // VERDADERA

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
let a = 20;  
let b = a; // b = 20  
let c = 15;  
let d = 10;  
  
// Evaluar la expresión  
let expresion3 = ((a == b) || (b > c)) || (c < d);  
console.log(`La expresión es ${expresion3 ? 'VERDADERA' : 'FALSA'}`);
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

Ejercicio 5: Valor de verdad de expresiones con variables

Resultado: true

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

PREGUNTAS CORTAS

RELACI3N DE EJERCICIOS

Ejercicios con Operadores Aritm3ticos, Relacionales y L3gicos

Ejercicio 6

Siendo el valor de las variables: $a = 10$ $b = 12$ $c = 13$ $d = 10$

Encontrar el valor de verdad de cada una de las siguientes expresiones

4. $((a > b) \text{ OR } (a < c)) \text{ AND } ((a = c) \text{ OR } (a \geq b))$

5. $((a \geq b) \text{ OR } (a < d)) \text{ AND } ((a \geq d) \text{ AND } (c > d))$

6. $\text{NOT } (a = c) \text{ AND } (c > b)$

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

// Siendo el valor de las variables: a = 10, b = 12, c = 13, d = 10

let a = 10;

let b = 12;

let c = 13;

let d = 10;

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

// Operación 4

```
let operacion4 = ((a > b) || (a < c)) && ((a == c) || (a >= b));
```

```
console.log("Operación 4: ", operacion4); // FALSA
```

// Operación 5

```
let operacion5 = ((a >= b) || (a < d)) && ((a >= d) && (c > d));
```

```
console.log("Operación 5: ", operacion5); // FALSA
```

// Operación 6

```
let operacion6 = !(a == c) && (c > b);
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

PREGUNTAS CORTAS

RELACI3N DE EJERCICIOS

1. Evaluamos las condiciones internas:

- `(a > b)` es `false` porque `10 > 12` es falso.
- `(a < c)` es `true` porque `10 < 13` .
- `(a = c)` es `false` porque `10 = 13` es falso.
- `(a >= b)` es `false` porque `10 >= 12` es falso.

2. Evaluamos la primera disyunci3n (OR):

- `(a > b) OR (a < c)` es `false OR true` , lo cual es `true` .

3. Evaluamos la segunda disyunci3n (OR):

- `(a = c) OR (a >= b)` es `false OR false` , lo cual es `false` .

4. Evaluamos la conjunci3n (AND):

- `true AND false` es `false` .

Por lo tanto, la expresi3n 4 es FALSA.

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

Expresión 5: `((a >= b) OR (a < d)) AND ((a >= d) AND (c > d))`

1. Evaluamos las condiciones internas:

• `(a >= b)` es `false` porque `10 >= 12` es falso.

• `(a < d)` es `false` porque `10 < 10` es falso.

• `(a >= d)` es `true` porque `10 >= 10` .

• `(c > d)` es `true` porque `13 > 10` .

2. Evaluamos la primera disyunción (OR):

• `(a >= b) OR (a < d)` es `false OR false` , lo cual es `false` .

3. Evaluamos la segunda conjunción (AND):

• `(a >= d) AND (c > d)` es `true AND true` , lo cual es `true` .

4. Evaluamos la conjunción (AND):

• `false AND true` es `false` .

Por lo tanto, la expresión 5 es FALSA.

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

Expresión 6: `NOT (a = c) AND (c > b)`

1. Evaluamos las condiciones internas:

- `(a = c)` es `false` porque `10 = 13` es falso.
- `NOT (a = c)` es `true` porque `NOT (false)` es `true`.
- `(c > b)` es `true` porque `13 > 12`.

2. Evaluamos la conjunción (AND):

- `true AND true` es `true`.

Por lo tanto, la expresión 6 es VERDADERA.

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
let a = 10;
let b = 12;
let c = 13;
let d = 10;

// Evaluar la expresión 4
let expresion4 = ((a > b) || (a < c)) && ((a == c) || (a >= b));
console.log(`La expresión 4 es ${expresion4 ? 'VERDADERA' : 'FALSA'}`);

// Evaluar la expresión 5
let expresion5 = ((a >= b) || (a < d)) && ((a >= d) && (c > d));
console.log(`La expresión 5 es ${expresion5 ? 'VERDADERA' : 'FALSA'}`);

// Evaluar la expresión 6
let expresion6 = !(a == c) && (c > b);
console.log(`La expresión 6 es ${expresion6 ? 'VERDADERA' : 'FALSA'}`);
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

¿VERDAD O MENTIRA?

La expresión 4 es FALSA

La expresión 5 es FALSA

La expresión 6 es VERDADERA

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

¿VERDAD O MENTIRA?

Ejercicio 6: Valor de verdad de expresiones con variables

Resultado 1: false

Resultado 2: true

Resultado 3: false

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

PREGUNTAS CORTAS

RELACI3N DE EJERCICIOS

Ejercicios con Operadores Aritm3ticos, Relacionales y L3gicos

Ejercicio 7

Encontrar el valor de verdad de la siguiente expresi3n

$M = 8, N = 9, R = 5, S = 5, T = 4, V = 7$

7. $\text{NOT}((M > N \text{ AND } R > S) \text{ OR } (\text{NOT}(T < V \text{ AND } S > M)))$

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

// Encontrar el valor de verdad de la siguiente expresión

let M = 8;

let N = 9;

let R = 5;

let S = 5;

let T = 4;

let V = 7;

// Operación 7

let operacion7 = !((M > N && R > S) || (!(T < V && S > M)));

console.log("Operación 7: ", operacion7); // VERDADERA

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
let M = 8;
let N = 9;
let R = 5;
let S = 5;
let T = 4;
let V = 7;

// Evaluar la expresión 7
let expresion7 = !((M > N && R > S) || !(T < V && S > M));
console.log(`La expresión 7 es ${expresion7 ? 'VERDADERA' : 'FALSA'}`);
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

PREGUNTAS CORTAS

RELACI3N DE EJERCICIOS

La expresi3n 7 es FALSA

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

Ejercicio 7: Valor de verdad de expresiones con constantes

Resultado: false

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

PREGUNTAS CORTAS

RELACI3N DE EJERCICIOS

Ejercicios con Operadores Aritm3ticos, Relacionales y L3gicos

Ejercicio 8

Aplicando la jerarquía de los operadores, encontrar el valor de verdad de cada una de las siguientes expresiones

8. $(3 * 2^2 - 4 / 2 * 1) > (3 * 2^2 * 1) \text{ AND } (5 > 11 \text{ MOD } 4)$

9. $(3 \geq 3 \text{ OR } 5 <> 5) \text{ AND NOT } (15 / 5 + 2 <> 5)$

10. $\text{NOT}(\text{NOT}((3(-3))^*2 > (3 - (-3)^*2) \text{ OR } 1^3 * 2 > 6))$

11. $(3 \geq 4 \text{ AND } 5 > 3 \text{ AND } 3 > 3) \text{ OR NOT } (4 \leq 4 \text{ OR } 5 > 4 \text{ OR } 6 \geq 7)$

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

// Aplicando la jerarquía de los operadores, encontrar el valor de verdad de cada una de las siguientes expresiones

// Operación 8

```
let operacion8 = ((3 * 2 ** 2 - 4 / 2 * 1) > (3 * 2 ** 2 * 1)) && (5 > 11 % 4);  
console.log("Operación 8: ", operacion8); // VERDADERA
```

// Operación 9

```
let operacion9 = ((3 >= 3 || 5 != 5)) && !((15 / 5 + 2) != 5);  
console.log("Operación 9: ", operacion9); // FALSA
```

// Operación 10

```
let operacion10 = !(!(3 * (-3)) * 2 > (3 - (-3) * 2) || 1 ** 3 * 2 > 6));  
console.log("Operación 10: ", operacion10); // FALSA
```

// Operación 11

```
let operacion11 = ((3 >= 4 && 5 > 3 && 3 > 3)) || !(4 <= 4 || 5 > 4 || 6 >= 7);  
console.log("Operación 11: ", operacion11); // FALSA
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
// Expresión 8
let expresion8 = ((3 * Math.pow(2, 2) - 4 / 2 * 1) > (3 * Math.pow(2, 2) * 1)) && (5 > 11);
console.log(`La expresión 8 es ${expresion8 ? 'VERDADERA' : 'FALSA'}`);

// Expresión 9
let expresion9 = (3 >= 3 || 5 != 5) && !(15 / 5 + 2 != 5);
console.log(`La expresión 9 es ${expresion9 ? 'VERDADERA' : 'FALSA'}`);

// Expresión 10
let expresion10 = !( !(3 * (-3)) * 2 > (3 - (-3) * 2) || Math.pow(1, 3) * 2 > 6 );
console.log(`La expresión 10 es ${expresion10 ? 'VERDADERA' : 'FALSA'}`);

// Expresión 11
let expresion11 = (3 >= 4 && 5 > 3 && 3 > 3) || !(4 <= 4 || 5 > 4 || 6 >= 7);
console.log(`La expresión 11 es ${expresion11 ? 'VERDADERA' : 'FALSA'}`);
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

PREGUNTAS CORTAS

RELACI3N DE EJERCICIOS

¿VERDAD O MENTIRA?

La expresi3n 8 es FALSA

La expresi3n 9 es VERDADERA

La expresi3n 10 es FALSA

La expresi3n 11 es FALSA

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

¿VERDAD O MENTIRA?

Ejercicio 8: Valor de verdad de expresiones con operadores

Resultado 1: false

Resultado 2: true

Resultado 3: true

Resultado 4: false

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

PREGUNTAS CORTAS

RELACI3N DE EJERCICIOS

EJERCICIOS CON BUCLES

Ejercicio 9.

Desarrolla una calculadora sencilla que pueda realizar sumas, restas, multiplicaciones y divisiones, seg3n lo que indique una variable de operaci3n. Utiliza las estructuras condicionales if, else if y else.

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
// Solicitamos al usuario los dos números
let num1 = parseFloat(prompt("Introduce el primer número:"));
let num2 = parseFloat(prompt("Introduce el segundo número:"));

// Solicitamos al usuario la operación a realizar
let operacion = prompt("Introduce la operación (+, -, *, /):");

// Declaramos una variable para almacenar el resultado
let resultado;

if (operacion === "+") {
    resultado = num1 + num2;
} else if (operacion === "-") {
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
resultado = num1 - num2;  
} else if (operacion == "*") {  
    resultado = num1 * num2;  
} else if (operacion == "/") {  
    resultado = num1 / num2;  
} else {  
    alert("Operación no válida");  
}  
  
// Mostramos el resultado  
alert("El resultado es: " + resultado);
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
// Definición de las variables
let num1 = 10;
let num2 = 5;
let operacion = "multiplicacion"; // Puede ser "suma", "resta", "multiplicacion" o "divis

// Variable para almacenar el resultado
let resultado;

// Estructura condicional para determinar la operación
if (operacion === "suma") {
    resultado = num1 + num2;
} else if (operacion === "resta") {
    resultado = num1 - num2;
} else if (operacion === "multiplicacion") {
    resultado = num1 * num2;
} else if (operacion === "division") {
    if (num2 !== 0) {
        resultado = num1 / num2;
    } else {
        resultado = "Error: División por cero";
    }
} else {
    resultado = "Operación no válida";
}

// Imprimir el resultado
console.log(`El resultado de la ${operacion} es: ${resultado}`);
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
El resultado de la multiplicacion es: 50
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

PREGUNTAS CORTAS

RELACI3N DE EJERCICIOS

EJERCICIOS CON BUCLES

Ejercicio 10

Crea un juego sencillo en el que el usuario tenga que adivinar un n3mero generado de forma aleatoria, utilizando un bucle do...while para mantener el juego en funcionamiento hasta que el usuario acierte.

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
// Generamos un número aleatorio entre 1 y 100
let numeroAleatorio = Math.floor(Math.random() * 100) + 1;
let intento;
let acertado = false;

do {
  // Solicitamos al usuario que adivine el número
  intento = parseInt(prompt("Adivina el número (entre 1 y 100):"));

  if (intento === numeroAleatorio) {
    acertado = true;
    alert("¡Felicidades! Adivinaste el número.");
  } else if (intento < numeroAleatorio) {
    alert("El número es mayor. Intenta de nuevo.");
  } else {
    alert("El número es menor. Intenta de nuevo.");
  }
} while (!acertado);
```


Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
// Generar un número aleatorio entre 1 y 100
const randomNumber = Math.floor(Math.random() * 100) + 1;
let guess;
let attempts = 0;

do {
  // Pedir al usuario que ingrese un número
  guess = parseInt(prompt("Adivina el número (entre 1 y 100):"), 10);
  attempts++;

  // Verificar si el número ingresado es correcto
  if (guess < randomNumber) {
    alert("El número es mayor. Intenta de nuevo.");
  } else if (guess > randomNumber) {
    alert("El número es menor. Intenta de nuevo.");
  } else if (guess === randomNumber) {
    alert(`¡Felicidades! Adivinaste el número en ${attempts} intentos.`);
  } else {
    alert("Por favor, ingresa un número válido.");
  }
} while (guess !== randomNumber);
```


Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

PREGUNTAS CORTAS

RELACI3N DE EJERCICIOS

EJERCICIOS CON BUCLES

Ejercicio 11

Escribe un bucle for que muestre en la consola los n3meros del 1 al 10.

```
for (let i = 1; i <= 10; i++) {  
  console.log(i);  
}
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

EJERCICIOS CON BUCLES

Ejercicio 12

Utiliza un bucle for para calcular la suma de los números del 1 al 100 y luego muestra el resultado en la consola.

```
let sum = 0;

for (let i = 1; i <= 100; i++) {
    sum += i;
}

console.log("La suma de los números del 1 al 100 es:", sum);
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

EJERCICIOS CON BUCLES

Ejercicio 13.

Mostrar la serie de Fibonacci hasta el décimo término.

```
let fibonacciSeries = [0, 1];

for (let i = 2; i < 10; i++) {
    fibonacciSeries[i] = fibonacciSeries[i - 1] + fibonacciSeries[i - 2];
}

console.log("La serie de Fibonacci hasta el décimo término es:", fibonacciSeries);
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

Ejercicio X

Desarrolla un programa que pida al usuario dos números y luego muestre todos los números primos que se encuentran entre esos dos números, incluyendo los propios extremos.

Solicita al usuario que introduzca dos números enteros. Identifica todos los números primos en el rango definido por los dos números proporcionados, incluyendo ambos extremos. Ejemplo de entrada: Primer número: 10, Segundo número: 20. Salida esperada: Los números primos entre 10 y 20 son: 11, 13, 17, 19.

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
// Función para verificar si un número es primo
```

```
function esPrimo(num) {
```

```
  if (num <= 1) return false;
```

```
  for (let i = 2; i <= Math.sqrt(num); i++) {
```

```
    if (num % i === 0) return false;
```

```
  }
```

```
  return true;
```

```
}
```

```
// Solicita al usuario que introduzca dos números enteros
```

```
let primerNumero = parseInt(prompt("Introduce el primer número:"));
```

```
let segundoNumero = parseInt(prompt("Introduce el segundo número:"));
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
// Identifica todos los números primos en el rango definido por los dos números proporcionados, incluyendo ambos extremos
let numerosPrimos = [];
for (let i = primerNumero; i <= segundoNumero; i++) {
  if (esPrimo(i)) {
    numerosPrimos.push(i);
  }
}

// Muestra los números primos encontrados
alert("Los números primos entre " + primerNumero + " y " + segundoNumero + " son: " + numerosPrimos.join(", "));
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
// Función para verificar si un número es primo
function esPrimo(num) {
  if (num <= 1) return false;
  if (num <= 3) return true;

  if (num % 2 === 0 || num % 3 === 0) return false;

  for (let i = 5; i * i <= num; i += 6) {
    if (num % i === 0 || num % (i + 2) === 0) return false;
  }

  return true;
}

// Función para obtener los números primos en un rango
function obtenerPrimosEnRango(inicio, fin) {
  let primos = [];
  for (let i = inicio; i <= fin; i++) {
    if (esPrimo(i)) {
      primos.push(i);
    }
  }
  return primos;
}
```


Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
// Solicitar al usuario que introduzca dos números
let primerNumero = parseInt(prompt("Introduce el primer número:"));
let segundoNumero = parseInt(prompt("Introduce el segundo número:"));

// Asegurarse de que primerNumero sea menor o igual a segundoNumero
if (primerNumero > segundoNumero) {
    [primerNumero, segundoNumero] = [segundoNumero, primerNumero];
}

// Obtener los números primos en el rango
let primos = obtenerPrimosEnRango(primerNumero, segundoNumero);

// Mostrar los números primos en la consola
console.log(`Los números primos entre ${primerNumero} y ${segundoNumero} son: ${primos.join(' ')}`);
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

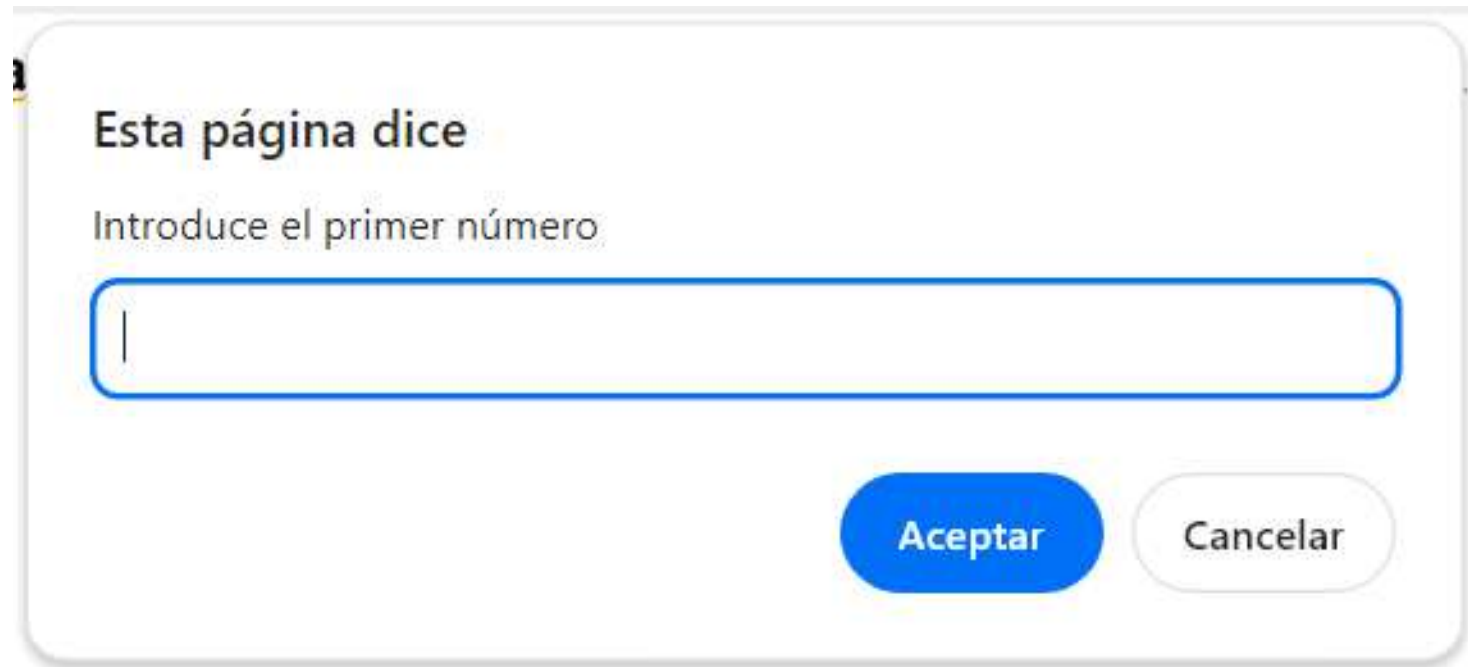
Ejercicio X: Números primos en un rango

Calcular Números Primos

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS



Esta página dice

Introduce el primer número

Aceptar Cancelar

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

PREGUNTAS CORTAS

RELACI3N DE EJERCICIOS

Ejercicio Y

Desarrolla un programa que produzca los primeros N t3rminos de la secuencia de Fibonacci, donde N es un n3mero proporcionado por el usuario.

Solicita al usuario que introduzca la cantidad de t3rminos de la secuencia de Fibonacci que desea generar. Emplea una funci3n para calcular la secuencia de Fibonacci. Muestra al usuario los primeros N t3rminos de la secuencia de Fibonacci.

Ejemplo de entrada: Introduce el n3mero de t3rminos de la secuencia de Fibonacci:

8. Salida esperada: Los primeros 8 t3rminos de la secuencia de Fibonacci son: 0, 1, 1, 2, 3, 5, 8, 13.

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
// Solicita al usuario que introduzca la cantidad de términos de la secuencia de Fibonacci
que desea generar

let numeroTerminos = parseInt(prompt("Introduce el número de términos de la secuencia de
Fibonacci"));

// Función para calcular la secuencia de Fibonacci
function fibonacci(n) {
  let secuencia = [0, 1];
  for (let i = 2; i < n; i++) {
    secuencia.push(secuencia[i - 1] + secuencia[i - 2]);
  }
  return secuencia;
}
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
// Genera la secuencia de Fibonacci y muestra los primeros N términos  
let secuenciaFibonacci = fibonacci(numeroTerminos);  
alert("Los primeros " + numeroTerminos + " términos de la secuencia de Fibonacci son: " +  
secuenciaFibonacci.join(", "););
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
// Función para generar la secuencia de Fibonacci
function generarFibonacci(n) {
  if (n <= 0) return [];
  if (n === 1) return [0];
  if (n === 2) return [0, 1];

  let fibonacciSeries = [0, 1];
  for (let i = 2; i < n; i++) {
    fibonacciSeries[i] = fibonacciSeries[i - 1] + fibonacciSeries[i - 2];
  }
  return fibonacciSeries;
}

// Solicitar al usuario que introduzca la cantidad de términos
let numeroDeTerminos = parseInt(prompt("Introduce el número de términos de la secuencia de"));

// Validar la entrada del usuario
if (isNaN(numeroDeTerminos) || numeroDeTerminos <= 0) {
  console.log("Por favor, introduce un número entero positivo.");
} else {
  // Generar y mostrar la secuencia de Fibonacci
  let fibonacci = generarFibonacci(numeroDeTerminos);
  console.log(`Los primeros ${numeroDeTerminos} términos de la secuencia de Fibonacci son`);
}
```


Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

Ejercicio Y: Secuencia de Fibonacci

Calcular Fibonacci

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

Esta página dice

Escribe un número N que determinará los N primeros números de la secuencia de Fibonacci por consola

Aceptar Cancelar

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

PREGUNTAS CORTAS

RELACI3N DE EJERCICIOS

Ejercicio 11

Escribe un bucle for que muestre en la consola los n3meros del 1 al 10.

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

PREGUNTAS CORTAS

RELACI3N DE EJERCICIOS

```
// Escribe un bucle for que muestre en la consola los n3meros del 1 al 10  
for (let i = 1; i <= 10; i++) {  
  console.log(i);  
}
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
// Bucle for que muestra los números del 1 al 10 en la consola  
for (let i = 1; i <= 10; i++) {  
  console.log(i);  
}
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

Ejercicio 11 - Bucle 1 a 10

0 1 2 3 4 5 6 7 8 9 10

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

PREGUNTAS CORTAS

RELACI3N DE EJERCICIOS

Ejercicio 12

Utiliza un bucle for para calcular la suma de los n3meros del 1 al 100 y luego muestra el resultado

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
// Utiliza un bucle for para calcular la suma de los números del 1 al 100 y luego muestra el resultado en la consola  
let suma = 0;  
for (let i = 1; i <= 100; i++) {  
    suma += i;  
}  
console.log("La suma de los números del 1 al 100 es: " + suma);
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

Ejercicio 12 - Bucle suma 1 a 100

1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31 33 35 37 39 41 43 45 47 49 51 53 55 57 59 61 63 65 67 69 71 73 75 77 79 81 83 85 87 89 91 93 95 97 99 101 103 105 107 109
111 113 115 117 119 121 123 125 127 129 131 133 135 137 139 141 143 145 147 149 151 153 155 157 159 161 163 165 167 169 171 173 175 177 179 181 183 185 187
189 191 193 195 197 199

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

Ejercicio 13.

Mostrar la serie de Fibonacci hasta el décimo término.

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
let a = 0;  
let b = 1;  
let siguiente;  
  
console.log(a); // Primer término  
console.log(b); // Segundo término  
  
for (let i = 3; i <= 10; i++) {  
    siguiente = a + b;  
    console.log(siguiente);  
    a = b;  
    b = siguiente;  
}
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

Ejercicio 14.

Diseña un programa que pida al usuario ingresar una lista de números y luego identifique el número más alto en esa lista.

- Pide al usuario que indique la cantidad de números que desea introducir.
- Solicita al usuario que ingrese cada número individualmente.
- Guarda estos números en un array.
- Define una función llamada `findLargestNumber` que reciba un array de números como parámetro y retorne el número más alto del array.
- Dentro de la función, recorre el array utilizando un bucle `for` para encontrar el número más alto.

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

PREGUNTAS CORTAS

RELACI3N DE EJERCICIOS

Ejercicio 14.

- Muestra al usuario el n3mero m3s alto encontrado.
- Aseg3rate de que el programa utilice arrays para almacenar los n3meros y que la b3squeda del n3mero m3s grande se realice mediante la funci3n.

Ejemplo:

- Ingresa la cantidad de n3meros: 5
- Ingresa el n3mero 1: 10
- Ingresa el n3mero 2: 25
- Ingresa el n3mero 3: 7
- Ingresa el n3mero 4: 18
- Ingresa el n3mero 5: 12 El n3mero m3s grande es: 25

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
// Solicita al usuario que ingrese cada número individualmente
let numeros = [];
for (let i = 0; i < cantidad; i++) {
  let numero = parseFloat(prompt('Introduce el número $ {i + 1} :'));
  numeros.push(numero);
}

// Define una función llamada findLargestNumber que reciba un array de números como parámetro
y retorne el número más alto del array
function findLargestNumber(arr) {
  let max = arr[0];
  for (let i = 1; i < arr.length; i++) {
    if (arr[i] > max) {
      max = arr[i];
    }
  }
  return max;
}
```


Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
}
```

```
// Llama a la función y muestra el número más alto encontrado
```

```
let numeroMayor = findLargestNumber(numeros);
```

```
alert("El número más grande es: " + numeroMayor);
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

Esta página dice

Escribe la cantidad de números

Aceptar Cancelar

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

PREGUNTAS CORTAS

RELACI3N DE EJERCICIOS

Ejercicio 15.

Mostrar los n3meros del 20 al 30 en la consola.

- Mostrar los n3meros pares entre 30 y 50 en la consola.
- Determinar la suma de los primeros 50 n3meros naturales.
- Mostrar la tabla de multiplicar del 8.
- Mostrar los elementos de un arreglo.
- Mostrar un patr3n de asteriscos en forma de tri3ngulo, con una altura de 9.
- Determinar la suma de los n3meros pares entre 1 y 50.
- Mostrar los n3meros del 30 al 20 en orden descendente.
- Determinar el promedio de un arreglo de n3meros.

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
// Mostrar los números del 20 al 30 en la consola
for (let i = 20; i <= 30; i++) {
  console.log(i);
}

// Mostrar los números pares entre 30 y 50 en la consola
for (let i = 30; i <= 50; i++) {
  if (i % 2 === 0) {
    console.log(i);
  }
}

// Determinar la suma de los primeros 50 números naturales
let suma = 0;
for (let i = 1; i <= 50; i++) {
  suma += i;
}
console.log("La suma de los primeros 50 números naturales es: " + suma);

// Mostrar la tabla de multiplicar del 8
```

```
for (let i = 1; i <= 10; i++) {
  console.log(`8 x ${i} = ${8 * i}`);
}

// Mostrar los elementos de un arreglo
let arreglo = [1, 2, 3, 4, 5];
for (let i = 0; i < arreglo.length; i++) {
  console.log(arreglo[i]);
}

// Mostrar un patrón de asteriscos en forma de triángulo, con una altura de 9
let altura = 9;
for (let i = 1; i <= altura; i++) {
  let fila = "";
  for (let j = 0; j < i; j++) {
    fila += "*";
  }
  console.log(fila);
}
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
// Determinar la suma de los números pares entre 1 y 50
let sumaPares = 0;
for (let i = 1; i <= 50; i++) {
  if (i % 2 === 0) {
    sumaPares += i;
  }
}
console.log("La suma de los números pares entre 1 y 50 es: " + sumaPares);

// Mostrar los números del 30 al 20 en orden descendente
for (let i = 30; i >= 20; i--) {
  console.log(i);
}

// Determinar el promedio de un arreglo de números
let numeros = [10, 20, 30, 40, 50];
let sumaNumeros = 0;
```

```
for (let i = 0; i < numeros.length; i++) {
  sumaNumeros += numeros[i];
}
let promedio = sumaNumeros / numeros.length;
console.log("El promedio del arreglo es: " + promedio);
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

Ejercicio 16

Mostrar los números del 20 al 30 en la consola.

- Mostrar los números pares entre 30 y 50 en la consola.
- Determinar la suma de los primeros 50 números naturales.
- Mostrar la tabla de multiplicar del 8.
- Mostrar los elementos de un arreglo.
- Mostrar un patrón de asteriscos en forma de triángulo, con una altura de 9.
- Determinar la suma de los números pares entre 1 y 50.
- Mostrar los números del 30 al 20 en orden descendente.
- Determinar el promedio de un arreglo de números.

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PREGUNTAS CORTAS

RELACIÓN DE EJERCICIOS

```
// Array de frases
let frasesFrikisInformatica = [
  "Soy un código dentro de un código.",
  "Realmente, en la vida no hay bugs, solo características.",
  "Para entender la recursión, primero debes entender la recursión.",
  "No hay lugar como 127.0.0.1.",
  "Cualquier problema en informática puede resolverse añadiendo otra capa de abstracción."
];

// Array de colores
let colores = ["red", "blue", "green", "purple", "orange"];

// Función para mostrar una frase aleatoria y cambiar el color
function mostrarFrase() {
  // Seleccionamos una frase aleatoria
  let frase = frasesFrikisInformatica[Math.floor(Math.random() * frasesFrikisInformatica.length)];

  // Seleccionamos un color aleatorio
  let color = colores[Math.floor(Math.random() * colores.length)];

  // Mostramos la frase y cambiamos el color
  let contenedor = document.getElementById("contenedor-frase");
  contenedor.innerText = frase;
  contenedor.style.color = color;

  // Cambiamos el color del botón
  let boton = document.getElementById("texto-informatica");
  boton.style.backgroundColor = color;
}
```


Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

PREGUNTAS CORTAS

RELACI3N DE EJERCICIOS

```
// Asignamos la funci3n al evento de clic del bot3n  
document.getElementById("texto-informatica").addEventListener("click", mostrarFrase);
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

PREGUNTAS CORTAS

RELACI3N DE EJERCICIOS

Ejercicio 16 - JavaScript + CSS

Frases Frikis de Inform1tica

Mostrar Frase

¡Hola, mundo 1!

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

PREGUNTAS CORTAS

RELACI3N DE EJERCICIOS

Ejercicio 17

Se te entrega un c3digo HTML y CSS para crear una lista de compras como la que se muestra a continuaci3n

Escribe el c3digo JavaScript necesario para que el formulario funcione correctamente.

Instrucciones:

- Selecciona los elementos del DOM necesarios: el campo de entrada para los elementos de la lista, el bot3n para agregar elementos y la lista donde se mostrar3n los elementos.

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

RELACIÓN DE EJERCICIOS

Ejercicio 17

- Implementa una función llamada addItem que se ejecutará al hacer clic en el botón «Agregar». Esta función debe realizar las siguientes acciones:
 - Obtener el valor del campo de entrada y asegurarse de que no esté vacío.
 - Crear un nuevo elemento de lista () y establecer su contenido con el valor del campo de entrada.
 - Crear dos botones dentro del nuevo elemento de lista: uno para marcar como completado y otro para eliminar el elemento.
 - Agregar eventos a los botones de completado y eliminación.
 - Añadir el nuevo elemento a la lista de elementos.
 - Limpiar el campo de entrada después de agregar el elemento.
- Asocia la función addItem al evento de clic del botón «Agregar»

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

RELACIÓN DE EJERCICIOS

```
// Seleccionamos los elementos del DOM
let inputElemento = document.getElementById("elemento");
let botonAgrega = document.getElementById("agregar");
let lista = document.getElementById("lista");

// Función para agregar un item
function addItem() {
    let valor = inputElemento.value.trim();

    if (valor !== "") {
        let li = document.createElement("li");
        li.innerText = valor;

        // Botón para marcar como completado
        let botonCompletado = document.createElement("button");
        botonCompletado.innerText = "Completado";
        botonCompletado.addEventListener("click", function() {
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

RELACIÓN DE EJERCICIOS

```
li.style.textDecoration = "line-through";  
});  
  
// Botón para eliminar el item  
let botonEliminar = document.createElement("button");  
botonEliminar.innerText = "Eliminar";  
botonEliminar.addEventListener("click", function() {  
    lista.removeChild(li);  
});  
  
li.appendChild(botonCompletado);  
li.appendChild(botonEliminar);  
lista.appendChild(li);  
inputElemento.value = "";  
}  
  
/ Asocia la función addItem al evento de clic del botón "Agregar"  
botonAgregar.addEventListener("click", addItem);
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

RELACIÓN DE EJERCICIOS

Ejercicio 17 - JavaScript + CSS

Lista de compras



A screenshot of a web application for a shopping list. It features a light gray rounded rectangle containing a text input field with the placeholder 'Añadir nuevo elemento' and a green 'Agregar' button. Below this, a list item '• comida' is shown with a blue 'Completado' button and a red 'Eliminar' button.

Añadir nuevo elemento

- comida

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

RELACI3N DE EJERCICIOS

Ejercicio 18

- Una de las cosas que sorprenden son el sin fin de animaciones que se pueden hacer con JavaScript, por ello se ha recolectado en los siguientes ejercicios una aproximaci3n para trabajar la animaci3n con JavaScript. Se listan a continuaci3n los problemas a resolver.
- En un foro de animaci3n de javascript se ha colocado el siguiente desaf3o, que consiste en: Animar el movimiento de un objeto, se deber3a de crea un objeto en una p3gina HTML y utilizar JavaScript para moverlo hacia la izquierda, derecha, arriba o abajo. Crea lo propio para resolver este desaf3o.

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

RELACI3N DE EJERCICIOS

Ejercicio 18

- En un hackaton dedicado a la animaci3n con javascript se ha publicado un reto, en el que se debe de animar el cambio de tama1o de un objeto, para ello se ha sugerido utilizar JavaScript para animar el cambio de tama1o de un objeto en una p1gina HTML, puede ser aumentando o disminuyendo su tama1o. Crea lo propio para solventar este reto.
- Un creador de animaciones se ha topado con una dificultad, la cual consiste en animar el cambio de color de un objeto, ha le3do que puede utilizar JavaScript para animar el cambio de color de un objeto en una p1gina HTML, puede que el cambiando sea gradualmente su color de fondo o el color del texto. Crea los archivos pertinentes para ayudar al creador de animaciones.

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

RELACIÓN DE EJERCICIOS

Ejercicio 18

- Fabrixo, ha entrado al mundo de animaciones web y se le ha encomendado una tarea que consiste en, animar la opacidad de un objeto, le han recomendado utilizar JavaScript para animar la opacidad de un objeto en una página HTML, ya sea haciendo que se vuelva más transparente o más opaco. Crea los archivos adecuados para ayudar a Frabixo.
- Perry, ha sido contratado por una diseñadora web y le ha encargado crear una animación de desvanecimiento. Le sugirieron utilizar JavaScript para crear una animación en la que un objeto desaparece gradualmente de la pantalla. Crea los archivos que sean necesario para ayudar a Perry.

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

RELACI3N DE EJERCICIOS

Ejercicio 18

- Mike, ha empezado a hacer su Servicio Social en una empresa que se dedica a la creaci3n de sitios web, tiene la encomienda de crear una animaci3n de rebote, se ha recomendado que utilice JavaScript para crear la animaci3n en la que un objeto rebota de un lado a otro de la pantalla, como una pelota que rebota en una pared. Crea los ficheros necesarios para ayudar a Mike.
- Estos ejercicios permite acercar al lector a la animaci3n b3sica con javascript.

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

RELACI3N DE EJERCICIOS

Ejercicio 18

- Mike, ha empezado a hacer su Servicio Social en una empresa que se dedica a la creaci3n de sitios web, tiene la encomienda de crear una animaci3n de rebote, se ha recomendado que utilice JavaScript para crear la animaci3n en la que un objeto rebota de un lado a otro de la pantalla, como una pelota que rebota en una pared. Crea los ficheros necesarios para ayudar a Mike.
- Estos ejercicios permite acercar al lector a la animaci3n b3sica con javascript.

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

RELACIÓN DE EJERCICIOS

```
// Animar el movimiento de un objeto
function moverObjeto() {
  let objeto = document.getElementById("objeto");
  let posicion = 0;
  let id = setInterval(frame, 10);

  function frame() {
    if (posicion == 350) {
      clearInterval(id);
    } else {
      posicion++;
      objeto.style.top = posicion + "px";
      objeto.style.left = posicion + "px";
    }
  }
}
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

RELACIÓN DE EJERCICIOS

```
// Animar el cambio de tamaño de un objeto
function cambiarTamanoObjeto() {
  let objeto = document.getElementById("objeto");
  let tamaño = 100;
  let id = setInterval(frame, 10);

  function frame() {
    if (tamaño == 200) {
      clearInterval(id);
    } else {
      tamaño++;
      objeto.style.width = tamaño + "px";
      objeto.style.height = tamaño + "px";
    }
  }
}
```


Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

RELACIÓN DE EJERCICIOS

```
// Animar el cambio de color de un objeto
function cambiarColorObjeto() {
  let objeto = document.getElementById("objeto");
  let colores = ["red", "blue", "green", "purple", "orange"];
  let index = 0;
  let id = setInterval(frame, 1000);

  function frame() {
    objeto.style.backgroundColor = colores[index];
    index = (index + 1) % colores.length;
  }
}
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

RELACI3N DE EJERCICIOS

```
// Animar la opacidad de un objeto
function cambiarOpacidadObjeto() {
  let objeto = document.getElementById("objeto");
  let opacidad = 1;
  let id = setInterval(frame, 100);

  function frame() {
    if (opacidad <= 0) {
      clearInterval(id);
    } else {
      opacidad -= 0.1;
      objeto.style.opacity = opacidad;
    }
  }
}
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

RELACI3N DE EJERCICIOS

```
// Animaci3n de desvanecimiento
function desvanecerObjeto() {
  let objeto = document.getElementById("objeto");
  let opacidad = 1;
  let id = setInterval(frame, 100);

  function frame() {
    if (opacidad <= 0) {
      clearInterval(id);
      objeto.style.display = "none";
    } else {
      opacidad -= 0.1;
      objeto.style.opacity = opacidad;
    }
  }
}
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

RELACIÓN DE EJERCICIOS

```
// Animación de rebote
function rebotarObjeto() {
  let objeto = document.getElementById("objeto");
  let posicion = 0;
  let direccion = 1;
  let id = setInterval(frame, 10);

  function frame() {
    if (posicion >= 350 || posicion <= 0) {
      direccion = -direccion;
    }
    posicion += direccion;
    objeto.style.left = posicion + "px";
  }
}
```

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

RELACIÓN DE EJERCICIOS

Elemento que se mueve 150px



Elemento que cambia de tamaño a 1.5



Elemento que cambia de color cada segundo



Elemento que se vuelve transparente



Pelota que rebota y desaparece



Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PÁGINA WEB DESDE CERO

The image is a YouTube video thumbnail for a tutorial titled "Desarrollo de una página web desde 0" (Development of a web page from 0). The video is by the channel "Business Technology". The thumbnail features a dark blue background with white text. At the top, it says "Crear una página web desde 0 con HTML CSS3 y JAVASCRIPT". Below this, there's a screenshot of a web page with a form titled "Agenda una cita con nosotros" (Book an appointment with us). The main title "Desarrollo de una página web desde 0" is prominently displayed in large white letters. To the right of the title is a red play button icon. Below the title, the text "HTML CSS3 JS" is written in large, bold letters, with each language name above its respective logo (HTML5, CSS3, and JavaScript). At the bottom left, there's a "Watch on YouTube" button. The thumbnail also includes a "Share" icon in the top right corner and a "Sobre nosotros" (About us) section with a photo of a person working on a computer.

Business Technology

Crear una página web desde 0 con HTML CSS3 y JAVASCRIPT

Agenda una cita con nosotros.

Estamos a la vanguardia de la tecnología

Desarrollo de una página web desde 0

HTML CSS3 JS

Sobre nosotros

Hacemos magia con código

Watch on YouTube

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

PÁGINA WEB DESDE CERO



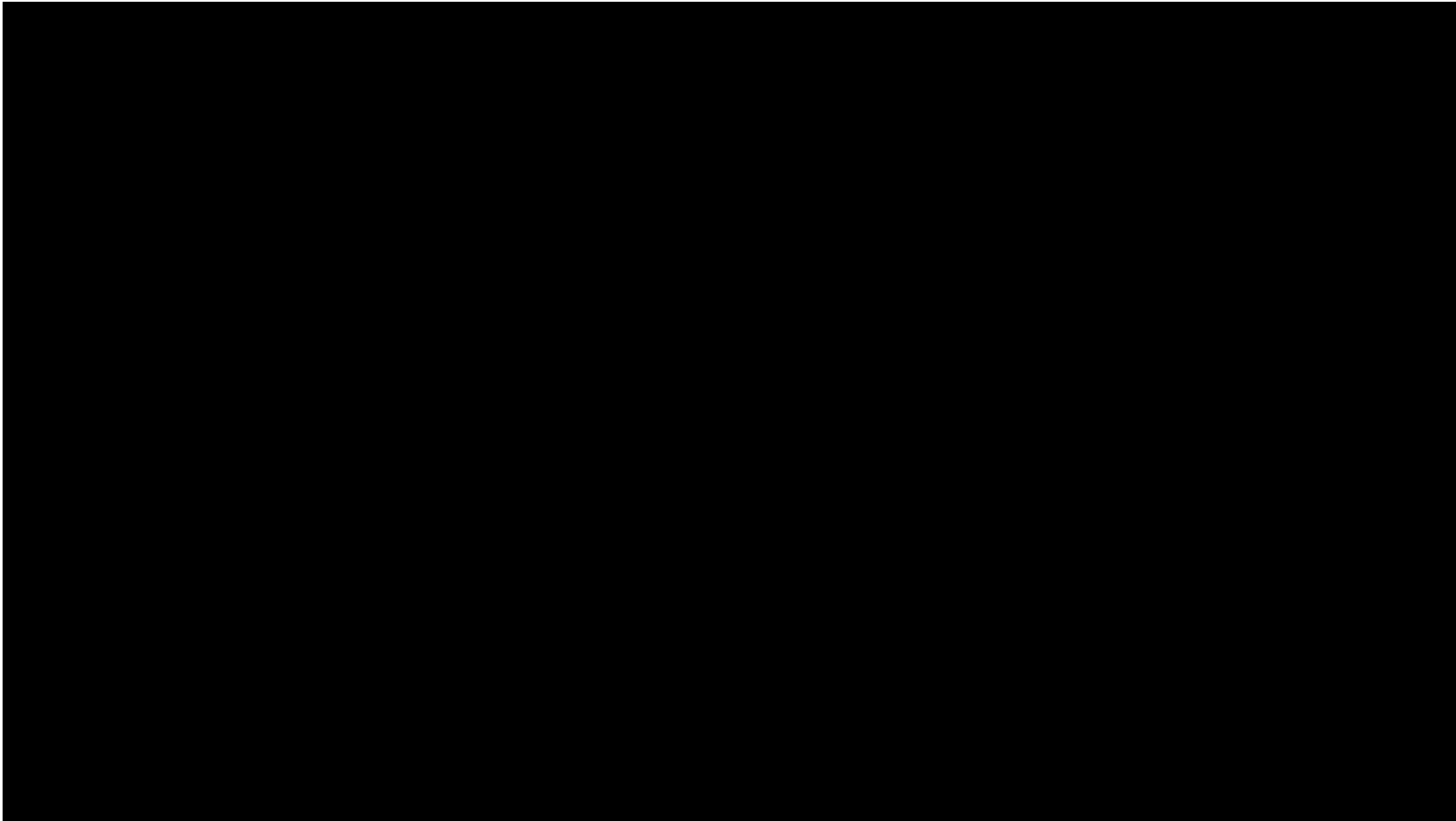
Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de gui3n

¿PARA QUÉ SIRVE JAVASCRIPT?



Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

Ejercicios resueltos básicos de Javascript



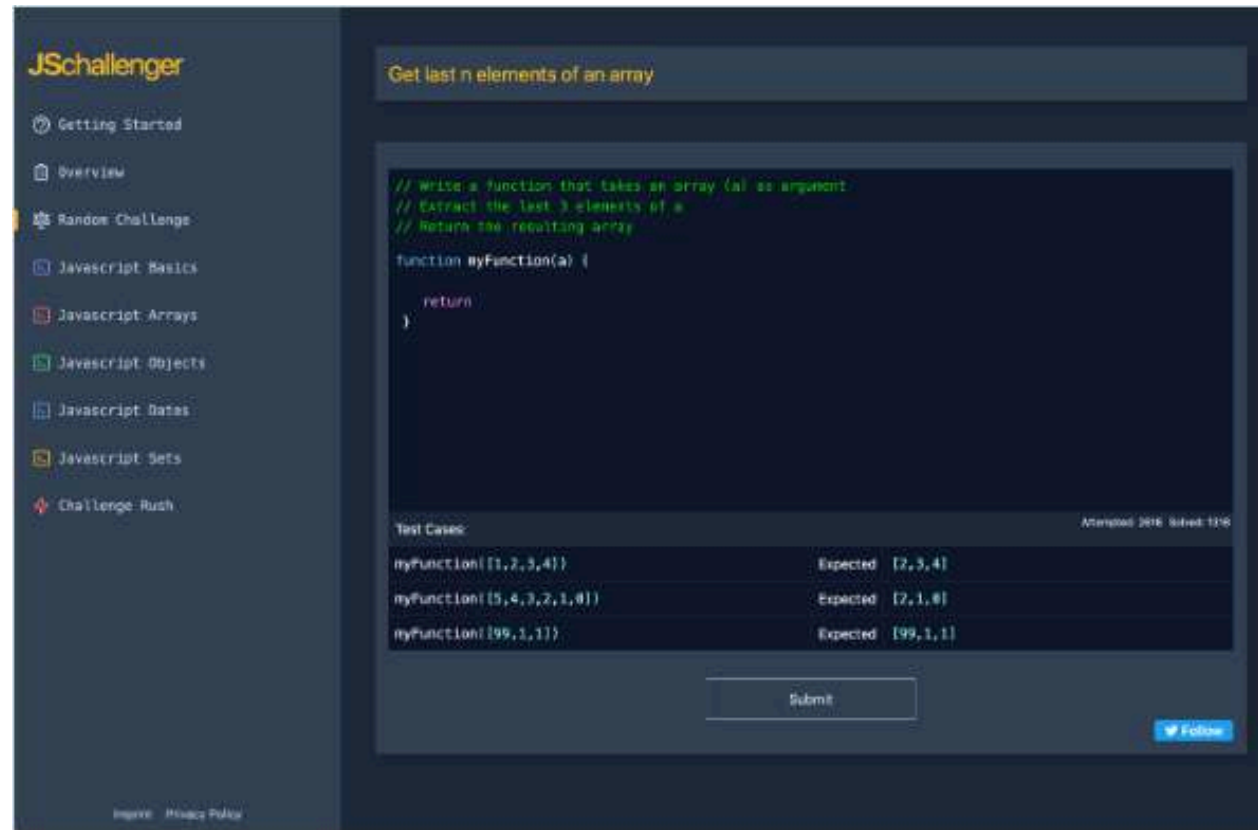
Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

Ejercicios resueltos básicos de Javascript



Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

JSCHALLENGER



JSchallenger

Free Javascript challenges. Learn Javascript online by solving coding exercises. Javascript for all levels. Solve Javascript tasks from beginner to advanced levels.

Unidad 2: Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión

Proyectos pequeños



Resolver las preguntas

Bajar lo del kahoot

Poner el fichaje del día de mañana

[https://www.eniun.com/ejercicios-resueltos-javascript-practicas-](https://www.eniun.com/ejercicios-resueltos-javascript-practicas-examenes/#111_Ejercicio_resuelto_JavaScript_Calculadora_de_gastos_de_viaje)

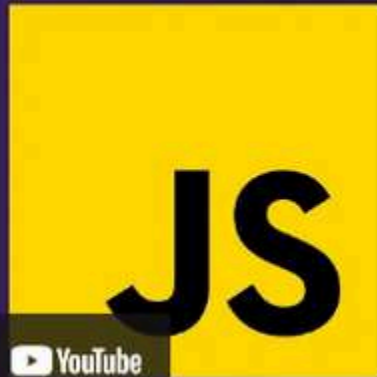
[examenenes/#111_Ejercicio_resuelto_JavaScript_Calculadora_de_gastos_de_viaje](https://www.eniun.com/ejercicios-resueltos-javascript-practicas-examenes/#111_Ejercicio_resuelto_JavaScript_Calculadora_de_gastos_de_viaje)




Entiende y Domina todas las Estructuras y Tipos de Datos en JavaScript



TIPOS DE DATOS



Watch on  YouTube

undefined

true

2

'texto'



Symbol()

null

{ name: 'Miguel' }

function doThings() {}

Ejercicios de practica de JS- María Eugenia Costa

Hola bienvenid@ a mi pagina de práctica de JavaScript, soy María Eugenia Costa, Desarrollador...

 github.io

<https://www.eniun.com/ejercicios-resueltos-javascript-practicas-examenes/>



Ejercicios de iniciación JavaScript 1. JavaScript desde cero

Ejercicios de Iniciación JavaScript. Ejercicios resueltos para aprender a programar en JavaScript. Variables, condicionales y bucles



Ejercicios de practica de JS- María Eugenia Costa

Hola bienvenid@ a mi pagina de práctica de JavaScript, soy María Eugenia Costa, Desarrollador...

github.io