Javascript Intermedio

OpenWebinars

¡BIENVENIDOS!

Soy Jonathan Rodríguez

Full stack developer

 es.linkedin.com/in/jojo5716

 @jojo5716

REQUISITOS

- **Nociones básicas de Javascript y HTML.**
- **Navegador web.**
- **Editor de código.**

CONTENIDO

- Objetos
- Arrays y Bucles
- DOM
- Eventos
- Expresiones regulares

QUE SON LOS OBJETOS

Colección de propiedades

↓

Grupo de propiedades ó atributos (Clave : Valor)

Objeto: Persona 

nombre: 'Jhon'

edad: 30

hablar: function() { ... }

```
{
    nombre: 'Jhon',
    edad: 30,
    hablar: function() { … },
}
```

CÓMO SE CREAN LOS OBJETOS

Objetos declarativos o literales

```
var persona1 = {
   nombre: 'Jhon',
   edad: 30,
  saludar: function() {
       console.log('Hola);
   }
}
```

Objetos construidos

```
function Persona(nombre, edad) {
   this.nombre = nombre;
   this.edad = edad;

   this.saludar = function () {
       console.log('Hola');
   }
}
var persona1 = new Persona('Jhon', 30);
```

Usando new Object

```
var persona1 = new Object({
   nombre: 'Jhon',
   edad: 30,
   saludar: function() { ... }
});
```

CONCLUSIÓN SOBRE LOS OBJETOS

En JavaScript, un objeto está compuesto por un conjunto de **propiedades**.

Una **propiedad** es una asociación entre un nombre y un valor.

Cuando el valor de una propiedad es una función, decimos que ésta es un **método** del objeto.

EJERCICIO

Imaginemos que vamos a desarrollar un juego, y debemos crear un objeto **Jugador** que tenga una propiedad **fuerza** que inicialmente tenga el valor 1, un método **incrementarFuerza** que nos permita incrementar en 1 la fuerza del jugador y un método **consultarFuerza** que nos muestre un mensaje con la fuerza del jugador.

Solución:
https://github.com/jojo5716/openwebinar/blob/master/intermedio/1.Objetos/ejercicio1.js

OBJETO DATE

Escribe un script que imprima un mensaje con la fecha de hoy con el siguiente formato.

Hola, hoy es XX del mes XXXX del año YYYY

Solución:
https://github.com/jojo5716/openwebinar/blob/master/intermedio/1.Objetos/ejercicioDate.js

OBJETO MATH

Escribe una función que reciba un número como parámetro y devuelva un número aleatorio entre cero y el número del parámetro.

Ejemplo:

- **CalcularAleatorio(10)** // Devuelve un número aleatorio entre cero y 10

Solución:
https://github.com/jojo5716/openwebinar/blob/master/intermedio/1.Objetos/ejercicioMath.js

QUE ES UN ARRAY

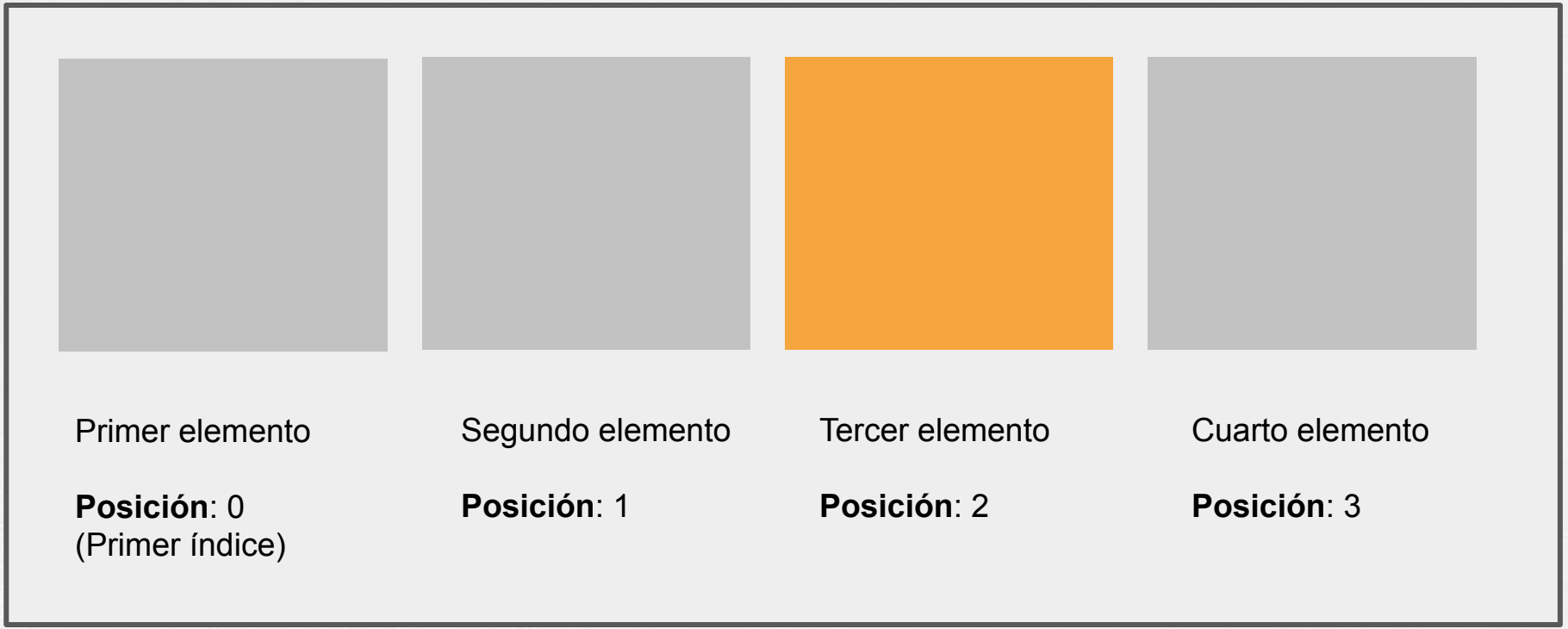
Conjunto de datos ordenados por posiciones

↓

Los datos pueden ser de cualquier tipo de dato

```
[
   'Primer elemento',
   'Segundo elemento',
   11,
   true,
   { key: "Soy el quinto elemento" },
   null,
]
```

ESTRUCTURA DE UN ARRAY



CÓMO SE CREA UN ARRAY

```
var person1 = ['Jhon', 30, true];
```

```
var person1 = new Array(3);

// o también

var person1 = new Array("rojo", "azul", "verde");
```

EJERCICIOS

1.- Muestra los números pares del siguiente array
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12];

Solución:
https://github.com/jojo5716/openwebinar/blob/master/intermedio/2.Array/ejercicio1.js

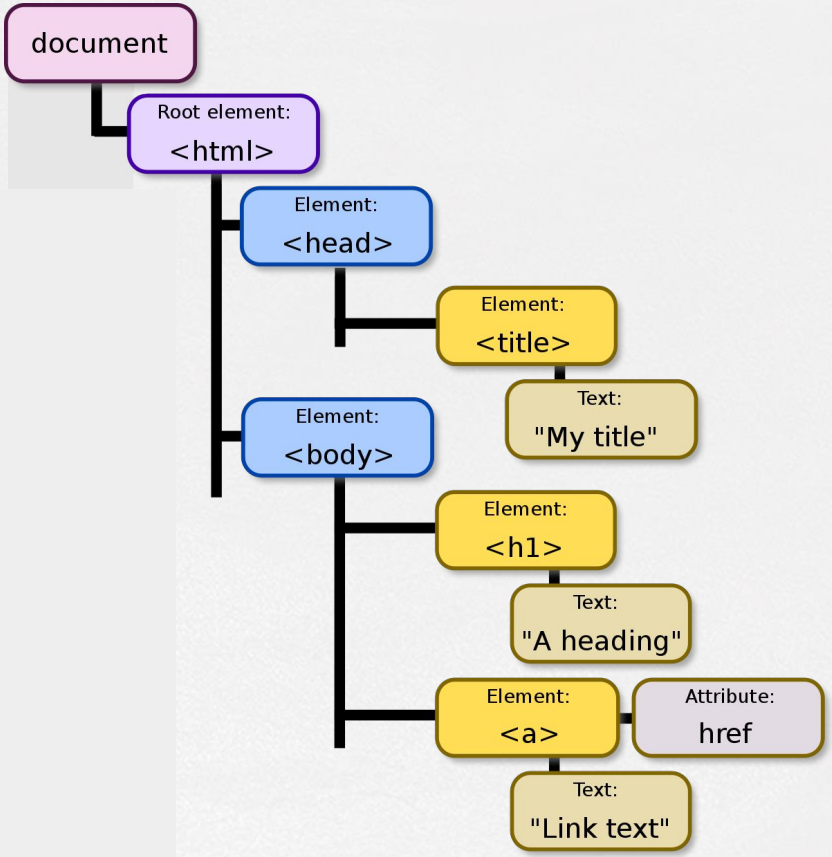
2.- Suma todos los números del array
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12];

Solución:
https://github.com/jojo5716/openwebinar/blob/master/intermedio/2.Array/ejercicio2.js

QUE ES EL DOM

Document
Object
Model

- Nodos
- Selectores
- Window
- Document



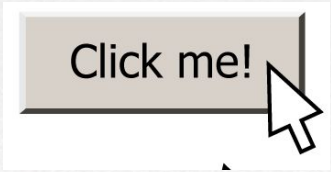
EJERCICIO SOBRE EL DOM

Crear un script que rellene un elemento UL (lista) de HTML con un array de strings.

Solución:
https://github.com/jojo5716/openwebinar/blob/master/intermedio/3.DOM/ejercicio1.html

QUE ES UN EVENTO

Acción que sucede en el sistema y le informa para que pueda responder de alguna manera si lo desea



```
function onClickButton() {
   // Código que se ejecuta al
   // producirse el evento
}
```

DIFERENTES TIPOS DE ACCIONES

Eventos del teclado

keyDown: Tecla hundida
keyUp: Al dejar de presionar una tecla
keyPress (*Deprecado*): Tecla hundida y sólo números y caracteres.

Eventos del ratón

onclick: Click sobre un elemento
onmousedown: Se pulsa un botón del ratón sobre un elemento.
onmouseenter: El puntero del ratón entra en el área de un elemento.
onmousemove: El puntero del ratón se está moviendo sobre el área de un elemento.

Eventos del navegador

DOMContentLoaded: La página se ha cargado

Eventos propios

EJERCICIO SOBRE EVENTOS

Continuando el ejercicio anterior, debemos agregar un input y un botón a nuestro HTML para crear elementos en nuestra lista de TODO.

Capturaremos el evento del botón y crearemos un elemento en la lista según el contenido del input.

Solución:
https://github.com/jojo5716/openwebinar/blob/master/intermedio/3.DOM/ejercicio1.html

Introducción a las Expresiones regulares

Que es

- Un objeto más.
- Describen patrones en cadena de texto.
- Incluido en la mayoría de lenguajes de programación.
- Simplifica las tareas de procesamiento de cadenas de texto.

Uso

Las expresiones regulares se utilizan para ***buscar***, ***leer*** y ***reemplazar*** información de cadenas de texto.

Estructura

/expresión-regular/[**parámetros**]

Ejemplo:

/(aaa)/gi (Coincidencia: caaandy, cAAAndy, caAAndy)

Métodos utilizado con Expresiones regulares

search `regexObj.test(cadena)`

- Nos permite saber si dicho patrón está presenta en un string o no.
- Devuelve el índice de la primera coincidencia y si no hay devuelve -1.
- Similar al método indexOf de los Strings.

exec `regexObj.exec(cadena)`

- Ejecuta una búsqueda de las coincidencias de la expresión regular en un string y devuelve un array o null.

test `regexObj.test(cadena)`

- Ejecuta una búsqueda de una ocurrencia entre la expresión regular y un string y retorna un booleano.

match `cadena.match(regexp)`

- Obtiene todas las ocurrencias de una expresión regular en un string.

Reglas de las EXPRESIONES REGULARES

Punto “.”

- Se interpreta como cualquier carácter

Contra barra “\”

- Se utiliza para que el siguiente carácter de la expresión adquiera un significado ó deje de tenerlo.

  Por ejemplo, al utilizarlo antes de un punto, éste deja de tener su significado original y lo toma de forma literal.

  Caracteres especiales:

- \t: Representa un tabulador.
- \r: Representa el "retorno de carro".
- \n: Representa la "nueva línea".

- \d: representa un dígito del 0 al 9.
- \w: representa cualquier carácter alfanumérico (incluidos los guiones bajos _).
- \s: representa un espacio en blanco

Reglas de las EXPRESIONES REGULARES

Corchetes “[]”

- Se utiliza para crear grupos de caracteres
- Se utiliza el guión medio para indicar rangos

Barra “|”

- Se utiliza para indicar varias opciones.

Dólar “$”

- Representa el final del string
- Si se utiliza en modo multi linea, no representa un caracter en especial sino una posición.

Ejemplo, si se utiliza la expresión .$, se buscará donde el punto finalice una línea.

Ejemplos más complejos de EXPRESIONES REGULARES

Validar una url

```
var regex = /(ftp|http|https):\/\/(\w+:{0,1}\w*@)?(\S+)(:[0-9]+)?(\/|\/([\w#!:.?+=&%@!\-\/]))?/
regex.test("https://openwebinars.net/");  // true
```

Validar un email

```
const regex =
/^(([^<>()\[\]\\.,;:\s@"]+(\.[^<>()\[\]\\.,;:\s@"]+)*)|(".+"))@((\[[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\])|(([a-zA-Z\-0-9]+\.)+[a-zA-Z]{2,}))$/;
regex.test("jhon.doe@gmail.com");  // true
```