	UNIVERSIDAD DON BOSCO FACULTAD DE INGENIERÍA ESCUELA DE COMPUTACIÓN		CICLO 02-2022
	MATERIA	LENGUAJES INTERPRETADOS EN EL CLIENTE	GUIA DE LABORATORIO N° 8
	PRÁCTICA	OBJETOS EN JAVASCRIPT	

I. OBJETIVOS

Que al finalizar la practica el estudiante aprenda a:

- Tenga claras todas las características de la Programación Orientada a Objetos con JavaScript.
- Utilice los diversos tipos de objetos incorporados con que cuenta el lenguaje JavaScript.
- Familiarización con los objetos del navegador más utilizados en los scripts.

II. INTRODUCCIÓN TEORICA

Programación Orientada a Objetos en JavaScript

La Programación Orientada a Objetos está basada en el concepto de dos entidades distintas, que son: las clases y las instancias. Una clase define las propiedades y métodos, a veces denominados simplemente miembros, que caracterizan a un cierto grupo de objetos. Una clase es una abstracción de ciertos objetos que existen en la realidad e intenta ser un modelo o plano de los objetos que representa, definiendo únicamente las características y el comportamiento que debe tener dicho objeto.


La instancia es un objeto concreto creado a partir de las propiedades y métodos definidos en la clase, a veces se prefiere decir que la instancia es un ejemplar concreto del tipo de objeto definido en la clase a partir de la cual se ha creado el objeto. Por ejemplo, Victoria Escobar es una instancia o un ejemplar concreto de la clase empleado, representando a una persona en particular que es o tiene las características y comportamiento de un empleado.

Los lenguajes basados en clases, como Java, C++ o C#, exigen que se deba definir primero una clase y a continuación realizar instancias de esta clase para crear un objeto en particular. JavaScript, es más bien un lenguaje basado en prototipos que no requiere hacer esta distinción, simplemente es un lenguaje que tiene objetos predefinidos, a los que se conoce como prototipos y que son, en la práctica, plantillas a partir de las cuales se obtienen propiedades y métodos para los nuevos objetos que se crean. Sin embargo, cada nuevo objeto puede especificar sus propias propiedades y métodos, tanto al ser creado o en tiempo de ejecución.

Diferencias entre lenguajes basados en clase versus lenguajes basados en prototipos

La siguiente tabla resume las diferencias entre los lenguajes basados en clases, como Java o C#, y los basados en prototipos como JavaScript.

Basado en Clases (Java)	Basado en Prototipos (JavaScript)
Clase e instancia son entidades distintas.	Todos los objetos son instancias.
Define una clase con la definición de clase; instancia una clase con los métodos constructores.	Define y crea un conjunto de objetos con las funciones constructoras.
Crea un único objeto con el operador new.	Igualmente.
Construye una jerarquía de objetos utilizando las definiciones de clases para definir subclases de una clase existente.	Construye una jerarquía de objetos mediante la asignación de un objeto como prototipo asociado con la función constructora.
Hereda las propiedades mediante el seguimiento de la cadena de clases.	Hereda las propiedades mediante el seguimiento de la cadena de prototipos.

	UNIVERSIDAD DON BOSCO FACULTAD DE INGENIERÍA ESCUELA DE COMPUTACIÓN		CICLO 02-2022
	MATERIA	LENGUAJES INTERPRETADOS EN EL CLIENTE	GUIA DE LABORATORIO N° 8
	PRÁCTICA	OBJETOS EN JAVASCRIPT	

Basado en Clases (Java)	Basado en Prototipos (JavaScript)
La definición de Clases especifica todas las propiedades de todas las instancias de una clase. No puede añadir propiedades dinámicamente en tiempo de ejecución.	La función constructora o prototipo puede especificar un conjunto inicial de propiedades. Puede añadir o remover dinámicamente a objetos individuales o a un conjunto entero de objetos.

Definición de objeto


Los objetos en JavaScript son tipos de datos compuestos, tal y como las matrices o arreglos, que dentro del lenguaje JavaScript son, más bien, objetos. Se dice que son compuestos porque combinan estado (propiedades que almacenan datos) y comportamiento (procedimientos o métodos para operar los datos).

Un objeto viene siendo una representación concreta, particular y real de algo. Esta representación determina su identidad, su estado y su comportamiento.

Clasificación de objetos en JavaScript

Los objetos en JavaScript se pueden subdividir en cuatro grupos:

- **Objetos definidos por el usuario.** Son objetos completamente personalizables que puede crear el desarrollador para dar estructura y coherencia a una tarea de programación en particular. Con estos objetos es posible definir propiedades y métodos mediante una sintaxis específica de JavaScript. Las propiedades identifican las características propias del objeto, mientras que los métodos permiten realizar tareas u operaciones con dichas propiedades.
- **Objetos incorporados del lenguaje.** Son objetos proporcionados por el propio lenguaje JavaScript, entre los que se incluyen: (1) objetos asociados a tipos de datos primitivos, como los objetos String, Number y Boolean, (2) objetos que permiten la creación de objetos definidos por el usuario, como los objetos Object y Form, (3) objetos utilizados para simplificar tareas frecuentes, como los objetos Date, Math y RegExp.
- **Objetos de navegador.** Son objetos que no están especificados como parte del lenguaje JavaScript, pero que son soportados por la mayor parte de navegadores que cumplen con los estándares. Los dos objetos más utilizados que pertenecen a esta categoría son los objetos Window y Navigator. El primero permite el control de la ventana del navegador y, el segundo, proporciona información acerca de la configuración del navegador del usuario.
- **Objetos de documento.** Son los objetos que forman parte del Modelo de Objetos de Documento (Document Object Model, abreviado como DOM) definido por la W3C. Es a través de estos objetos que JavaScript puede manejar Hojas de Estilo en Cascada y la ejecución de HTML dinámico. El acceso a los objetos de documento es proporcionado por el navegador mediante el uso del objeto document del objeto Window.

	UNIVERSIDAD DON BOSCO FACULTAD DE INGENIERÍA ESCUELA DE COMPUTACIÓN		CICLO 02-2022
	MATERIA	LENGUAJES INTERPRETADOS EN EL CLIENTE	GUIA DE LABORATORIO N° 8
	PRÁCTICA	OBJETOS EN JAVASCRIPT	

Creación de objetos en JavaScript

Existen dos formas de crear objetos en JavaScript:

1. Utilizando la palabra clave **new** y el nombre del **constructor del objeto** a crear. Se pueden crear objetos especializados haciendo uso del operador **new** y a continuación invocar al **método constructor** del objeto que inicializa las propiedades del objeto.

Ejemplo:

```

1  var carro = new Object();
2  carro.marca = "Toyota";
3  carro.modelo = "Corolla";
4  carro.npasajeros = 5;
5  carro.annio = 2007;
6  carro.placa = "503-618";
7  carro.nchasis = "JDSXV39Z200750226";
8  carro.encender = function (numero) {
9      carro.nchasis == numero ? (start = "on") : (start = "off");
10     return start;
11 };
12 carro.apagar = function () {
13     start = off;
14 };

```

2. Utilizando sintaxis de literales de objeto. Esta es la forma más fácil de crear objetos en JavaScript, que consiste de una lista de pares nombre_propiedad: valor separados por comas y encerrados entre llaves.

Ejemplo:

```

1  var coord = { x: 0, y: 0 };
2  var circle = { x: coord.x, y: coord.y, r: 2.5 };

```

Esta forma de construir objetos en JavaScript es generalizable a todos los tipos de datos del lenguaje. Veamos esto demostrado con dos tipos de objetos como **String** y **Array**.

Utilizando la palabra clave new y el respectivo método constructor:

```

1  var nombre = new String();
2  var paises = new Array("El Salvador", "Guatemala", "Honduras");


```

Utilizando sintaxis declarativa conocida como sintaxis de literales de tipos de datos:

```

1  var nombre = "";
2  var paises = ["El Salvador", "Guatemala", "Honduras"];

```

	UNIVERSIDAD DON BOSCO FACULTAD DE INGENIERÍA ESCUELA DE COMPUTACIÓN		CICLO 02-2022
	MATERIA	LENGUAJES INTERPRETADOS EN EL CLIENTE	GUIA DE LABORATORIO N° 8
	PRÁCTICA	OBJETOS EN JAVASCRIPT	

Objetos del lenguaje

Estos son el conjunto de objetos que vienen incorporados por el lenguaje. Entre los principales tenemos los objetos Array, Function, Date, String, Math, Boolean, Number.

1. Objeto Array

Este objeto se abordó en guías pasadas, por lo que trataremos únicamente su declaración, propiedades y métodos. La forma de declarar un objeto Array es haciendo uso del constructor Array(). Si se pasan argumentos al constructor, se interpreta que estos serán los elementos del arreglo. La excepción se da cuando se le pasa al constructor un solo valor numérico, en cuyo caso se entiende que se le proporciona al arreglo un tamaño, que podremos acceder mediante la propiedad length.

Los siguientes son ejemplos de declaración de objetos Array:

```

1 //Crea un arreglo vacío (sin elementos definidos) y crea una referencia
2 //a él en la variable arreglo1
3 var arreglo1 = new Array();
4 //Crea un arreglo con tres elementos
5 var arreglo2 = new Array("rojo", "verde", "azul");
6 //Crea un arreglo cuya propiedad length tiene el valor de 5
7 var arreglo3 = new Array(5);

```

Propiedades del objeto Array

- length

Como su nombre indica esta propiedad nos devuelve la longitud del array, es decir, el número de elementos que puede almacenar. Su uso es muy simple:

```

1 var lista = new Array(50);
2 tamagno = lista.length; /*tamaño que almacenaría el valor 50 */

```

- prototype

Esta es una propiedad muy potente en el sentido que nos permite agregar al objeto Array las propiedades y métodos que queramos.

```


1 Array.prototype.descriptor = null;
2 dias = new Array("lunes", "Martes", "Miercoles", "Jueves", "Viernes");
3 dias.descriptor = "Dias laborables de la semana";

```

En este ejemplo hemos creado una nueva propiedad para el objeto Array, la propiedad descriptor que podría utilizarse para darle un título a la matriz.

Métodos del objeto Array

Para ver estos métodos consulte la guía sobre arreglos realizada recientemente. Entre dichos métodos se encuentran: concat(), join(), reverse(), sort(), pop(), push(), etc.

	UNIVERSIDAD DON BOSCO FACULTAD DE INGENIERÍA ESCUELA DE COMPUTACIÓN		CICLO 02-2022
	MATERIA	LENGUAJES INTERPRETADOS EN EL CLIENTE	GUIA DE LABORATORIO N° 8
	PRÁCTICA	OBJETOS EN JAVASCRIPT	

2. Objeto Function

Las funciones también son consideradas como objetos por JavaScript. Aunque normalmente no estamos acostumbrados a definir las de esta forma, JavaScript proporciona una forma de definir funciones en donde explícitamente se declaran como objetos. Para ello se utiliza el operador new y el constructor Function, tal y como se muestra a continuación:

```
sumar = new Function("a", "b", "return a + b");
```

La llamada a la function se realice de la misma forma que conocemos. Puede ver el siguiente ejemplo de llamada:

```
document.write(sumar(90, 100));
```

Métodos de Function

Los métodos de Function son todos los heredados del objeto Object, que veremos más adelante.

Propiedades del objeto Function

Básicamente se tienen tres propiedades que se pueden usar con el objeto Function y son:

- **arguments**

Se trata de un array que contiene los argumentos pasados a la función. Esta propiedad permite el uso de funciones con un número variable de argumentos.

- **caller**

Contiene una referencia a la función que llamó a la actual.

- **constructor**

Heredada de la clase Object.


3. Objeto String

El objeto **String** se usa para manipular cadenas de caracteres. En JavaScript todo texto encerrado entre comillas, dobles o simples, se interpreta como una cadena, así '45' no es el número cuarenta y cinco sino la cadena formada por los caracteres 4 y 5. Este objeto contiene una gran cantidad de métodos para el manejo y el análisis de cadenas, la extracción de subcadenas e incluso para la conversión de cadenas a texto de marcado HTML. Como en los anteriores objetos se utiliza el operador new y el constructor String(), como se muestra a continuación:

```
var cadena = new String();
```

También se puede pasar argumentos al constructor, que será su valor inicial, como se muestra en este otro ejemplo:

```
var cadena = new String("No me importa.");
```

	UNIVERSIDAD DON BOSCO FACULTAD DE INGENIERÍA ESCUELA DE COMPUTACIÓN		CICLO 02-2022
	MATERIA	LENGUAJES INTERPRETADOS EN EL CLIENTE	GUIA DE LABORATORIO N° 8
	PRÁCTICA	OBJETOS EN JAVASCRIPT	

Propiedades del objeto String

- ***length***

Valor numérico que nos indica la longitud en caracteres de la cadena dada.

- ***prototype***

Nos permite asignar nuevas propiedades al objeto String.

Métodos del objeto String

- ***anchor(name)***

Crea un enlace (o ancla) a partir de un objeto String con el atributo name igual a la cadena que se le pasa como argumento al método.

Vea el siguiente ejemplo:

```
var refer = "referencia1";
var ancla1 = refer.anchor("anclaje1");
```

Después de ejecutadas las instrucciones anteriores el valor de ancla será:

```
<a name="anclaje1">referencia1</a>
```

La sintaxis de este método permite usar una constante String en lugar del nombre de un objeto String. El ejemplo anterior podría haber escrito como:


```
var ancla = "referencia1".anchor("anclaje1");
```

- ***charAt(position)***

Este método aplicado a una cadena devuelve el carácter que se encuentra en la posición dada por el atributo *position* teniendo en cuenta que el índice del primer carácter a la izquierda de la cadena es 0 y el último es una unidad menor que longitud de la cadena. Si el valor del atributo no es válido (igual o mayor que la longitud de la cadena o negativo) el método devuelve el valor *undefined*. Por ejemplo, el siguiente código devuelve el carácter en la tercera posición de la cadena nombre:

```
var nombre = "abcdefghij";
var car3 = nombre.charAt(2);
```

Devolverá "c", que es el tercer carácter por la izquierda (índice igual a 2).

	UNIVERSIDAD DON BOSCO FACULTAD DE INGENIERÍA ESCUELA DE COMPUTACIÓN		CICLO 02-2022
	MATERIA	LENGUAJES INTERPRETADOS EN EL CLIENTE	GUIA DE LABORATORIO N° 8
	PRÁCTICA	OBJETOS EN JAVASCRIPT	

4. Objeto Date

El objeto Date contiene un valor que representa fecha y hora de un instante dado. Para crear una instancia de este objeto usamos alguna de las siguientes sintaxis:

```
var fecha = new Date()
var fecha = new Date(número)
var fecha = new Date(cadena)
var fecha = new Date(año, mes, día[, hora[, minutos[, seg[, ms]]]])
```

Los argumentos encerrados entre corchetes son opcionales. En la primera forma la variable fecha contendrá la fecha del día actual. La segunda opción almacena en fecha la fecha dada por el argumento como el número de milisegundos transcurridos desde la media noche del 1 de Enero de 1970. El tercer tipo se usa cuando la fecha se pasa en forma de cadena. Por último la fecha puede crearse pasándole como argumento los números de año, mes, día, hora y opcionalmente, hora, minuto, segundo y milisegundo. Los años posteriores a 1970 puede escribirse con dos dígitos, pero es aconsejable usar siempre cuatro dígitos por aquello de los efectos 2000.

```
var hoy = new Date(); /*fecha del día en hoy */
var evento = new Date("November 10 1990");
var otro = new Date("10 Nov 1990");
var otro = new Date("10/02/2000"); //Oct, 2, 2000
var instante = new Date(1990, 11, 10, 20, 00);
```

Estas son las posibles formas de declarar objetos de tipo fecha. Las dos últimas almacenan el mismo día, pero en la última además se guarda la hora.


Donde se usen cadenas para indicar una fecha podemos añadir al final las siglas GMT (o UTC) para indicar que la hora se refiere a hora del meridiano Greenwich, si no se toma como hora local, o sea, según la zona horaria configurada en el ordenador donde se ejecute el script.

Métodos del objeto fecha

- ***getDate()***

Nos devuelve el día del mes del objeto fecha al que se aplica. Este método controla por supuesto el número de días de cada mes y contempla el caso de años bisiestos, incluida la excepción del 2000. En el siguiente ejemplo se presenta en pantalla Hoy es día 2, suponiendo que la fecha del sistema es 2-10-200. Primero creamos la variable fecha instanciada como un objeto Date() para a continuación escribir directamente el valor de getDate() aplicado a fecha.

```
var fecha = new Date();
document.write("Hoy es día: " + fecha.getDate());
```

	UNIVERSIDAD DON BOSCO FACULTAD DE INGENIERÍA ESCUELA DE COMPUTACIÓN		CICLO 02-2022
	MATERIA	LENGUAJES INTERPRETADOS EN EL CLIENTE	GUIA DE LABORATORIO N° 8
	PRÁCTICA	OBJETOS EN JAVASCRIPT	

- **setFullYear()**

Nos permite cambiar el año del objeto fecha por el valor pasado como argumento, un número de dos dígitos que se interpreta como decenas dentro del siglo, o sea, que para poner el año 1995 se debe pasar 95. El ejemplo pone precisamente este valor en el campo año de la variable fecha.

```
var fecha = new Date();
fecha.setYear(95);
document.write(fecha.toString());
```

Ojo si pasamos el valor 00 en el argumento el año obtenido es el 1900 (uno de los efectos 2000), por esto es recomendable usar la función setFullYear(agno).

- **setDate()**

Nos permite cambiar el día del mes del objeto fecha al que se aplica para poner el valor que se pasado en el argumento diasmes. Este método controla por supuesto el número de días de cada mes y contempla el caso de años bisiestos, incluida la excepción del 2000, de forma que si pasamos como argumento 31 y el mes es de 30 días la función corrige la fecha completa pasándola al día 1 del mes siguiente. Esto lo vemos en el ejemplo que sigue: creamos una variable como un objeto Date con el último día de Septiembre (mes de 30 días) e intentamos poner el día a 31, luego comprobamos la fecha almacenada:

```
var fecha = new Date("1 Sep 2008");
fecha.setDate(31);
document.write("Hoy es día: " + fecha.toString());
```


Como puede verse, si se prueba el ejemplo la fecha es corregida y pasa a 1 de Octubre.

5. Objetos del navegador

Los objetos del navegador se encuentran en el nivel más alto de la jerarquía de objetos que define JavaScript. Estos son **window**, **frame**, **location**, **history**, y **navigator**. El objeto más alto en la jerarquía de objetos que define JavaScript es **window**. El resto de objetos (excepto el objeto **navigator**) se ubican siempre dentro de una ventana. Los métodos y propiedades de los objetos **window** y **navigator** pueden utilizarse sin necesidad de declarar variables.

Ejemplo:

```
//Mostrar la hora segundo a segundo
function hourStatus() {
    var miFecha = new Date();
    window.status =
        formato(miFecha.getHours()) +
        ":" +
        formato(miFecha.getMinutes()) +
        ":" +
        formato(miFecha.getSeconds());
    //Llamando a la función segundo a segundo
    setTimeout("hourStatus()", 1000);
}
//Función para formatear las partes de la hora
function formato(valor) {
    if (valor < 10) valor = "0" + valor;
    return valor;
}
```


	UNIVERSIDAD DON BOSCO FACULTAD DE INGENIERÍA ESCUELA DE COMPUTACIÓN		CICLO 02-2022
	MATERIA	LENGUAJES INTERPRETADOS EN EL CLIENTE	GUIA DE LABORATORIO N° 8
	PRÁCTICA	OBJETOS EN JAVASCRIPT	

6. Objetos de documento

Los objetos del documento se encuentran en el segundo nivel de la jerarquía de objetos que define JavaScript. Estos son **document**, **link**, **anchor** e **image**. Estos objetos permiten a JavaScript, entre otras cosas, el manejo de Hojas de Estilo en Cascada y la ejecución de HTML dinámico (DHTML).

El acceso a los objetos del documento es proporcionado por el navegador mediante el uso de la **propiedad document** del **objeto window**.


Ejemplo:

```
//Función que obtiene el texto seleccionado
function getTextSelected() {
    var selText = "";
    //Internet Explorer
    if (document.selection) {
        selText = document.selection.createRange().text;
    }
    //Otros navegadores (Chrome, Firefox, Opera, etc.)
    if (window.getSelection) {
        selText = window.getSelection();
    }
    if (document.getSelection) {
        selText = document.getSelection();
    }
    //Asignar el texto seleccionado a un elemento P
    document.getElementById("paragraph").innerHTML =
        "<p>Texto seleccionado: " + selText + "</p>";
}
```

Estas últimas dos categorías de objetos se estudiarán con más detalle en próximas guías de práctica.

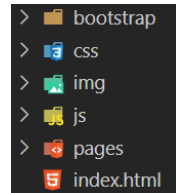
III. MATERIALES Y EQUIPO

No.	Requerimientos	Cantidad
1	Computadora con alguno de los siguientes editores de código fuente tales como: Sublime Text, Visual Studio Code, Notepad++ u otro y Navegadores Web Actualizados (Firefox, Chrome, Safari, Opera, Microsoft Edge entre otros)	1
2	Guía de laboratorio 8	1
3	Memoria USB o cualquier otro medio de almacenamiento.	1

	UNIVERSIDAD DON BOSCO FACULTAD DE INGENIERÍA ESCUELA DE COMPUTACIÓN		CICLO 02-2022
	MATERIA	LENGUAJES INTERPRETADOS EN EL CLIENTE	GUIA DE LABORATORIO N° 8
	PRÁCTICA	OBJETOS EN JAVASCRIPT	

III. DESARROLLO DE LA PRÁCTICA

Cree una carpeta con el nombre Guia8_[su número de carnet], y luego proceda a descomprimir los recursos proporcionados. Deberá de tener la siguiente estructura en su proyecto.



Ejemplo 1: Calculando área y perímetro de un rectángulo.


El ejemplo solicita dos valores numéricos (pueden ser con punto flotante) para calcular, presionando los botones respectivos, el perímetro y el área de un rectángulo con esas dimensiones. Se utilizan campos de formulario de tipo number, incorporados en HTML5 y se establecen valores mínimo, máximo y el desplazamiento para manejar los valores ingresados con el control que algunos navegadores modernos incorporan. También es factible ingresar los valores directamente

1. Localice el archivo ejemplo1.js y comience a desarrollar el siguiente código

```

1  //INICIALIZANDO CONTROLES
2  const btnarea = document.getElementById("idBtnAltura");
3  const btnperim = document.getElementById("idBtnPerimetro");
4  const base = document.getElementById("idTxtBase");
5  const altura = document.getElementById("idTxtAltura");
6
7  ⚡ Registrar evento click del ratón al presionar botones de envío
8  const iniciar = () => {
9      if (btnarea.addEventListener) {
10         btnarea.addEventListener("click", calculararea, false);
11     } else {
12         btnarea.attachEvent("onclick", calculararea);
13     }
14     if (btnperim.addEventListener) {
15         btnperim.addEventListener("click", calcularperimetro, false);
16     } else {
17         btnperim.attachEvent("onclick", calcularperimetro);
18     }
19 };
20
21 const calculararea = function () {
22     if (base.value == "" || altura.value == "") {
23         alert("Faltan campos por llenar");
24     } else {
25         var rect = new rectangulo(base.value, altura.value);
26         rect.mostrar(rect.carea(), " área");
27     }
28     return false;
29 };
30
31 const calcularperimetro = function () {
32     if (base.value == "" || altura.value == "") {
33         alert("Faltan campos por llenar");
34     } else {
35         var peri = new rectangulo(base.value, altura.value);
36         peri.mostrar(peri.cperimetro(), "perímetro");
37     }
38     return false;
39 };

```


	UNIVERSIDAD DON BOSCO FACULTAD DE INGENIERÍA ESCUELA DE COMPUTACIÓN		CICLO 02-2022
	MATERIA	LENGUAJES INTERPRETADOS EN EL CLIENTE	GUIA DE LABORATORIO N° 8
	PRÁCTICA	OBJETOS EN JAVASCRIPT	

```

41 //Creando una clase rectángulo
42 const rectangulo = function (base, altura) {
43     //Propiedades de la clase
44     this.base = parseFloat(base);
45     this.altura = parseFloat(altura);
46     //Métodos de la clase
47     //definidos usando el constructor Function()
48     this.carea = new Function("return this.base * this.altura");
49     this.cperimetro = new Function("return 2*this.base + 2*this.altura");
50     this.mostrar = new Function(
51         "valor",
52         "tipoc",
53         "alert('El ' + tipoc + ' es: ' + valor)"
54     );
55 };
56
57 //Iniciando controles
58 iniciar();

```

- Verifique el funcionamiento de su página index.html, tendría que obtener un resultado como el siguiente.



Ejemplo 2: Calculando edad exacta.

El siguiente ejemplo muestra una forma de poder calcular la edad exacta de una persona, ingresando la fecha de nacimiento mediante campos de formulario donde debe ingresarse el día, el mes y el año de nacimiento. El cálculo de la edad exacta se hace con respecto a la fecha actual, que tendría que estar bien configurada en el sistema operativo del servidor o computadora donde se ejecuta el script.

- Localice el archivo ejemplo2.js y comience a desarrollar el siguiente código

```

1 //Accediendo a los controles
2 const cmbDia = document.getElementById("idCmbDia");
3 const cmbMes = document.getElementById("idCmbMes");
4 const txtAnio = document.getElementById("idTxtAnio");
5 const btnCalcular = document.getElementById("idBtnCalcular");
6
7 const arrayMes = [
8     "Enero",
9     "Febrero",
10    "Marzo",
11    "Abril",
12    "Mayo",
13    "Junio",
14    "Julio",
15    "Agosto",
16    "Septiembre",
17    "Octubre",
18    "Noviembre",
19    "Diciembre",
20 ];

```



UNIVERSIDAD DON BOSCO
FACULTAD DE INGENIERÍA
ESCUELA DE COMPUTACIÓN

CICLO
02-2022

MATERIA	LENGUAJES INTERPRETADOS EN EL CLIENTE
PRÁCTICA	OBJETOS EN JAVASCRIPT

GUIA DE
LABORATORIO N° 8

```
22 //Función para determinar si un año es bisiesto o no
23 const esBisiesto = function (anno) {
24     if (anno % 4 == 0 && (anno % 400 == 0 || anno % 100 != 0)) {
25         bisiesto = true;
26     } else {
27         bisiesto = false;
28     }
29     return bisiesto;
30 };
31
32 //Función para limpiar el menú de selección Días
33 const quitarDias = function (menuDias) {
34     for (i = 0; i < menuDias.options.length; i++) {
35         menuDias.options[i] = null;
36     }
37 };
38
39 const llenarDias = function (mes) {
40     let i;
41     const menuDias = document.frmEdad.idCmbDia;
42     quitarDias(menuDias);
43     switch (mes) {
44         case "Enero":
45         case "Marzo":
46         case "Mayo":
47         case "Julio":
48         case "Agosto":
49         case "Octubre":
50         case "Diciembre":
51             for (i = 0; i < 31; i++) {
52                 menuDias[i] = new Option(parseInt(i + 1), parseInt(i + 1));
53             }
54             break;
55         case "Abril":
56         case "Junio":
57         case "Septiembre":
58         case "Noviembre":
59             for (i = 0; i < 30; i++) {
60                 menuDias[i] = new Option(parseInt(i + 1), parseInt(i + 1));
61             }
62             break;
63         case "Febrero":
64             if (esBisiesto(txtAnio.value)) {
65                 for (i = 0; i < 29; i++) {
66                     menuDias[i] = new Option(parseInt(i + 1), parseInt(i + 1));
67                 }
68             } else {
69                 for (i = 0; i < 28; i++) {
70                     menuDias[i] = new Option(parseInt(i + 1), parseInt(i + 1));
71                 }
72             }
73             break;
74     }
75 };
```



UNIVERSIDAD DON BOSCO
FACULTAD DE INGENIERÍA
ESCUELA DE COMPUTACIÓN

CICLO
02-2022

MATERIA	LENGUAJES INTERPRETADOS EN EL CLIENTE
PRÁCTICA	OBJETOS EN JAVASCRIPT

GUIA DE
LABORATORIO N° 8

```
77 // Cargando informacion al combo de mes
78 const loadMes = () => {
79     let option, textnode;
80     for (const m of arrayMes) {
81         document.appendChild;
82         option = document.createElement("option");
83         option.value = m;
84         textnode = document.createTextNode(m);
85         option.appendChild(textnode);
86         cmbMes.appendChild(option);
87     }
88 };
89
90 //Función para obtener la edad exacta de la persona
91 const calcularEdad = function (dia, mes, annio) {
92     var tusdias, tusmeses, tusannios;
93     var fecActual = new Date();
94     var year = fecActual.getFullYear();
95     var month = parseInt(fecActual.getMonth()) + 1;
96     var day = fecActual.getDate();
97     tusdias = day - dia;
98
99     switch (mes) {
100         case "Enero":
101             mes = 1;
102             if (tusdias < 0) {
103                 tusdias = day - dia + 31;
104                 month--;
105             }
106             break;
107         case "Febrero":
108             mes = 2;
109             if (tusdias < 0 && esBisiesto(annio)) {
110                 tusdias = day - dia + 29;
111                 month--;
112             } else if (tusdias < 0 && !esBisiesto(annio)) {
113                 tusdias = day - dia + 28;
114                 month--;
115             }
116             break;
117         case "Marzo":
118             mes = 3;
119             if (tusdias < 0) {
120                 tusdias = day - dia + 31;
121                 month--;
122             }
123             break;
124         case "Abril":
125             mes = 4;
126             if (tusdias < 0) {
127                 tusdias = day - dia + 30;
128                 month--;
129             }
130             break;
```




UNIVERSIDAD DON BOSCO
FACULTAD DE INGENIERÍA
ESCUELA DE COMPUTACIÓN

CICLO
02-2022

MATERIA	LENGUAJES INTERPRETADOS EN EL CLIENTE
PRÁCTICA	OBJETOS EN JAVASCRIPT

GUIA DE
LABORATORIO N° 8

```
131     case "Mayo":
132         mes = 5;
133         if (tusdias < 0) {
134             tusdias = day - dia + 31;
135             month--;
136         }
137         break;
138     case "Junio":
139         mes = 6;
140         if (tusdias < 0) {
141             tusdias = day - dia + 30;
142             month--;
143         }
144         break;
145     case "Julio":
146         mes = 7;
147         if (tusdias < 0) {
148             tusdias = day - dia + 31;
149             month--;
150         }
151         break;
152     case "Agosto":
153         mes = 8;
154         if (tusdias < 0) {
155             tusdias = day - dia + 31;
156             month--;
157         }
158         break;
159     case "Septiembre":
160         mes = 9;
161         if (tusdias < 0) {
162             tusdias = day - dia + 30;
163             month--;
164         }
165         break;
166     case "Octubre":
167         mes = 10;
168         if (tusdias < 0) {
169             tusdias = day - dia + 31;
170             month--;
171         }
172         break;
173     case "Noviembre":
174         mes = 11;
175         if (tusdias < 0) {
176             tusdias = day - dia + 30;
177             month--;
178         }
179         break;
180     case "Diciembre":
181         mes = 12;
182         if (tusdias < 0) {
183             tusdias = day - dia + 31;
184             month--;
185         }
186         break;
187 }
```


	UNIVERSIDAD DON BOSCO FACULTAD DE INGENIERÍA ESCUELA DE COMPUTACIÓN		CICLO 02-2022
	MATERIA	LENGUAJES INTERPRETADOS EN EL CLIENTE	GUIA DE LABORATORIO N° 8
	PRÁCTICA	OBJETOS EN JAVASCRIPT	

```

188     tusmeses = month - mes;
189     if (tusmeses < 0) {
190         tusmeses = month - mes + 12;
191         year--;
192     }
193     tusannios = year - annio;
194     alert(
195         "Tu edad exacta es:\n" +
196         tusannios +
197         " años,\n" +
198         tusmeses +
199         " meses, y\n" +
200         tusdias +
201         " días."
202     );
203 };
204
205 //Activando eventos de controles
206 btnCalcular.onclick = function () {
207     let dia = cmbDia.value;
208     let mes = cmbMes.value;
209     let anio = txtAnio.value;
210     calcularEdad(dia, mes, anio);
211 };
212
213 cmbMes.onchange = function () {
214     llenarDias(this.value);
215 };
216
217 txtAnio.onchange = function () {
218     llenarDias(cmbMes.value);
219 };
220
221 //Cargando informacion
222 llenarDias("Enero");
223 loadMes();

```

2. Verifique el funcionamiento de su página ejemplo2.html, tendría que obtener un resultado como el siguiente.


Objetos en JavaScript


Ejemplo 1
Ejemplo 2
Ejemplo 3
Ejemplo 4
Ejemplo 5
Ejemplo 6

Calculando la edad

Día

Mes

Año

	UNIVERSIDAD DON BOSCO FACULTAD DE INGENIERÍA ESCUELA DE COMPUTACIÓN		CICLO 02-2022
	MATERIA	LENGUAJES INTERPRETADOS EN EL CLIENTE	GUIA DE LABORATORIO N° 8
	PRÁCTICA	OBJETOS EN JAVASCRIPT	

Ejemplo 3: Zona horaria.


En el siguiente ejemplo se muestra cómo crear un script JS que permita indicar en base a una zona horaria mundial seleccionada por el usuario la hora y fecha de esa zona horaria. En este ejemplo se hace uso de los objetos incorporados de JavaScript Date para manejar la fecha y el objeto Math para realizar los cálculos de desfase horario entre las zonas.

1. Localice el archivo ejemplo3.js y comience a desarrollar el siguiente código

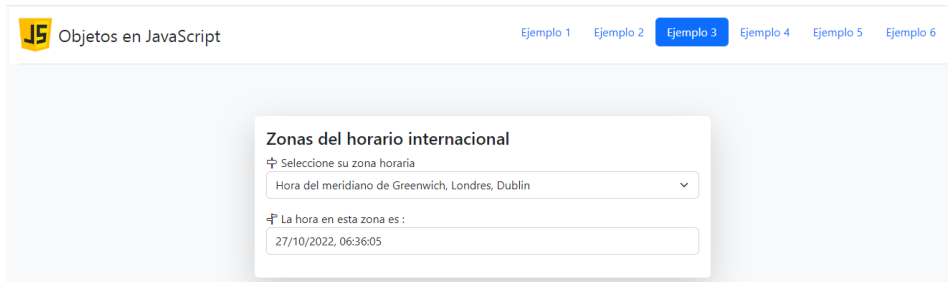
```

1  //La fecha del día en la zona del cliente web
2  //usando el objeto Date de JavaScript
3  let fechaHoy = new Date();
4
5  //Cálculo del desfase en el huso horario para hacer
6  //luego el ajuste con la zona horaria seleccionada
7  //en el elemento select del formulario
8  let desfase = -Math.round(fechaHoy.getTimezoneOffset() / 60) + 12;
9
10 //Registrar evento click del ratón al presionar botones de envío
11 const iniciar = function () {
12     let select = document.getElementById("zhselect");
13     if (select.addEventListener) {
14         select.addEventListener("change", getHoraLocal, false);
15     } else if (select.attachEvent) {
16         select.attachEvent("onchange", getHoraLocal);
17     }
18 };
19
20 //Función getHoraLocal()
21 //Calcula la hora en la zona horaria seleccionada
22 const getHoraLocal = function () {
23     let fechaHoy = new Date();
24     //Ajustar el desfase horario respecto a la zona seleccionada.
25     //Por ejemplo, la zona horaria para América Central está
26     //en el índice 6 de la colección de opciones del elemento select.
27     //Si un usuario seleccionara la zona horaria de Tokio que está
28     //en el índice 21 de la colección el cálculo se realizaría así:
29     //21 - desfase del cliente (zona horaria de América Central) = 15
30     let zh = document.frmZonaHoraria.zonas.selectedIndex - desfase;
31
32     //Sumar (o restar) las horas de desfase respecto a la hora
33     //local del cliente
34     fechaHoy.setHours(fechaHoy.getHours() + zh);
35
36     //Informar la hora local del huso horario elegido
37     document.frmZonaHoraria.hour.value = fechaHoy.toLocaleString();
38 };
39
40 //Iniciando datos
41 iniciar();
42 getHoraLocal();

```


	UNIVERSIDAD DON BOSCO FACULTAD DE INGENIERÍA ESCUELA DE COMPUTACIÓN		CICLO 02-2022
	MATERIA	LENGUAJES INTERPRETADOS EN EL CLIENTE	
	PRÁCTICA	OBJETOS EN JAVASCRIPT	
			GUIA DE LABORATORIO N° 8

- Verifique el funcionamiento de su página ejemplo3.html, tendría que obtener un resultado como el siguiente.



Ejemplo 4: Registrando información de libros.


Ejemplo que permite ingresar en un formulario datos de un libro que luego son almacenados en propiedades de un objeto libro definido con el objeto Object de JavaScript. Se verifica que los datos obtenidos sean correctos, se limpian los campos del formulario y finalmente se construye una tabla HTML con uno de los métodos del objeto donde se muestran los datos ya almacenados en las propiedades de ese objeto creado.

- Localice el archivo ejemplo4.js y comience a desarrollar el siguiente código

```

1 //Creando el objeto
2 let book = new Object();
3
4 //Registrar evento click del ratón al presionar botones de envío
5 const iniciar = function () {
6     const showinfo = document.getElementById("idBtnEnviar");
7     const txtAutor = document.getElementById("txtautor");
8     txtAutor.focus();
9     if (showinfo.addEventListener) {
10         showinfo.addEventListener(
11             "click",
12             function () {
13                 createObject(document.frmbook);
14             },
15             false
16         );
17     } else if (showinfo.attachEvent) {
18         showinfo.attachEvent("onclick", function () {
19             createObject(document.frmbook);
20         });
21     }
22 };
23
24 // Creando el nuevo objeto
25 const createObject = function (form) {
26     //Verificando que exista autor, nombre del libro y pais
27     if (
28         form.txtautor.value == "" ||
29         form.txttitulo.value == "" ||
30         form.txtpais.value == ""
31     ) {
32         alert("Faltan campos por completar");
33         return false;
34     }
35
36     book.autor = form.txtautor.value;
37     book.titulo = form.txttitulo.value;
38     book.editorial =
39         form.seleditorial.options[form.seleditorial.selectedIndex].text;
40     book.edicion = form.seledicion.options[form.seledicion.selectedIndex].text;
41     book.pais = form.txtpais.value;
42
43     showProperties(book, "InfoBook");
44 };

```

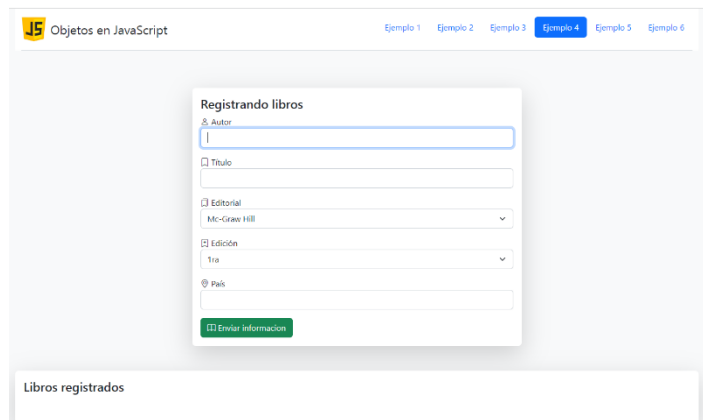
	UNIVERSIDAD DON BOSCO FACULTAD DE INGENIERÍA ESCUELA DE COMPUTACIÓN		CICLO 02-2022
	MATERIA	LENGUAJES INTERPRETADOS EN EL CLIENTE	GUIA DE LABORATORIO N° 8
	PRÁCTICA	OBJETOS EN JAVASCRIPT	


```

46 const showProperties = function (objeto, objName) {
47     let infBook = "";
48     let tblBook = "";
49
50     for (let i in objeto) {
51         infBook = infBook + objName + "." + i + " = " + objeto[i] + "\n";
52     }
53     if (!confirm(infBook + "\n¿Desea agregar la siguiente información?")) {
54         frmbook.txtautor.value = "";
55         frmbook.txttitulo.value = "";
56         frmbook.seleditorial.value = "a";
57         frmbook.seledicion.value = "a";
58         frmbook.txtpais.value = "";
59     }
60     tblBook = `
61     <table class="table table-striped table-hover table-bordered">
62         <thead>
63             <tr>
64                 <th scope="col" class="text-center">Título</th>
65                 <th scope="col" class="text-center">Autor</th>
66                 <th scope="col" class="text-center">Editorial</th>
67                 <th scope="col" class="text-center">Edición</th>
68                 <th scope="col" class="text-center">País</th>
69             </tr>
70         </thead>
71         <tbody>
72             <tr>
73                 <td>${book.titulo}</td>
74                 <td>${book.autor}</td>
75                 <td>${book.editorial}</td>
76                 <td>${book.edicion}</td>
77                 <td>${book.pais}</td>
78             </tr>
79         </tbody>
80     </table>
81     `;
82
83     document.getElementById("idDivResultado").innerHTML = tblBook;
84 };
85
86 //Asociando función que manejará el evento load al cargar la página
87 if (window.addEventListener) {
88     window.addEventListener("load", iniciar, false);
89 } else if (window.attachEvent) {
90     window.attachEvent("onload", iniciar);
91 }

```

- Verifique el funcionamiento de su página ejemplo4.html, tendría que obtener un resultado como el siguiente.



	UNIVERSIDAD DON BOSCO FACULTAD DE INGENIERÍA ESCUELA DE COMPUTACIÓN		CICLO 02-2022
	MATERIA	LENGUAJES INTERPRETADOS EN EL CLIENTE	GUIA DE LABORATORIO N° 8
	PRÁCTICA	OBJETOS EN JAVASCRIPT	

Ejemplo 5: Registrando información de carros.

Este ejemplo muestra cómo controlar con JavaScript las opciones de un campo select dependiente de otro campo select. De modo que al cambiar la opción seleccionada en el campo select principal, se limpia por completo el campo select dependiente y se vuelve a llenar inmediatamente con las opciones que se corresponden con la opción seleccionada.

1. Localice el archivo ejemplo5.js y comience a desarrollar el siguiente código

```

1  //Registrar evento click al presionar botones de envío
2  //y evento change al cambiar de opción en el elemento select
3  const iniciar = function () {
4      const select = document.getElementById("selfab");
5      const button = document.getElementById("idBtnEnviar");
6      //Al producirse en evento change en el elemento select
7      //invocar a la función addOptions para volver a llenar
8      //el select dependiente con los datos adecuados
9      if (select.addEventListener) {
10         select.addEventListener(
11             "change",
12             function () {
13                 addOptions(
14                     marcas[this.options[this.selectedIndex].text],
15                     document.frmcar.selmod
16                 );
17             },
18             false
19         );
20     } else {
21         select.attachEvent("onchange", function () {
22             addOptions(
23                 marcas[this.options[this.selectedIndex].text],
24                 document.frmcar.selmod
25             );
26         });
27     }

```



UNIVERSIDAD DON BOSCO
FACULTAD DE INGENIERÍA
ESCUELA DE COMPUTACIÓN

CICLO
02-2022

MATERIA

LENGUAJES INTERPRETADOS EN EL CLIENTE


PRÁCTICA

OBJETOS EN JAVASCRIPT

GUIA DE
LABORATORIO N° 8

```
28 //Al producirse un click sobre el botón de envío
29 //invocar los métodos del objeto carro que mostrarán
30 //los valores ingresados en el formulario
31 ✓ if (button.addEventListener) {
32 ✓   button.addEventListener(
33     "click",
34 ✓   function () {
35     let seleccion = showRadioSelected(document.frmcar.radcolor);
36
37 ✓     if (document.frmcar.txtanio.value != "" && seleccion != undefined) {
38 ✓       carro.pedido(
39         document.frmcar.selfab.value,
40         document.frmcar.selmod.value,
41         seleccion,
42         document.frmcar.txtanio.value
43       );
44       carro.mostrar();
45       alert("Informacion registrada");
46 ✓     } else {
47       alert("Faltan campos por completar");
48     }
49   },
50   false
51 );
52 ✓ } else {
53 ✓   button.attachEvent("onclick", function () {
54     let seleccion = showRadioSelected(document.frmcar.radcolor);
55 ✓     carro.pedido(
56       document.frmcar.selfab.value,
57       document.frmcar.selmod.value,
58       seleccion,
59       document.frmcar.txtanio.value
60     );
61     carro.mostrar();
62   });
63 }
64 };


66 //Iniciando matriz para manejar las marcas y sus respectivos modelos
67 const marcas = new Array(7);
68 marcas["Toyota"] = [
69   "Corolla",
70   "Echo",
71   "Yaris",
72   "Avensis",
73   "Camry",
74   "Land Cruiser",
75   "4 Runner",
76   "Hilux",
77 ];
```

	UNIVERSIDAD DON BOSCO FACULTAD DE INGENIERÍA ESCUELA DE COMPUTACIÓN		CICLO 02-2022
	MATERIA	LENGUAJES INTERPRETADOS EN EL CLIENTE	GUIA DE LABORATORIO N° 8
	PRÁCTICA	OBJETOS EN JAVASCRIPT	

```

78  marcas["Nissan"] = [
79      "Sentra",
80      "Platina",
81      "Almera",
82      "Altima",
83      "Tiida",
84      "Pathfinder",
85      "Patrol",
86      "X-Trail",
87      "Frontier",
88  ];
89  marcas["Hyundai"] = ["Elantra", "Accent", "Coupé", "Santa Fe", "i30"];
90  marcas["Volkswagen"] = [
91      "Golf",
92      "Jetta",
93      "Passat",
94      "Phaeton",
95      "Thunder Bunny",
96      "Touareg",
97      "Saveiro",
98  ];
99  marcas["Chevrolet"] = [
100     "Optra",
101     "Aveo",
102     "Cobalt",
103     "Malibu",
104     "Corvette",
105     "Chevy",
106     "Avalanche",
107     "Trailblazer",
108 ];
109  marcas["Honda"] = [
110     "Civic",
111     "Acura",
112     "Accord",
113     "Fit",
114     "Odyssey",
115     "CR-V",
116     "Pilot",
117     "RidgeLine",
118 ];
119  marcas["Mitsubishi"] = [
120     "Lancer",
121     "Galant",
122     "Eclipse",
123     "Montero",
124     "Nativa",
125     "Outlander",
126     "L200",
127 ];


```

	UNIVERSIDAD DON BOSCO FACULTAD DE INGENIERÍA ESCUELA DE COMPUTACIÓN		CICLO 02-2022
	MATERIA	LENGUAJES INTERPRETADOS EN EL CLIENTE	GUIA DE LABORATORIO N° 8
	PRÁCTICA	OBJETOS EN JAVASCRIPT	

```

129 //Creando el objeto carro con el constructor Object()
130 let carro = new Object();
131 //Propiedades del objeto
132 carro.fabricante = "";
133 carro.modelo = "";
134 carro.color = "";
135 carro.anio = "";
136 //Métodos del objeto
137 carro.pedido = function (fab, mod, col, an) {
138     carro.fabricante = fab;
139     carro.modelo = mod;
140     carro.color = col;
141     carro.anio = an;
142 };
143
144 carro.mostrar = function () {
145     let tblCarro = "";
146     tblCarro = `
147     <table class="table table-striped table-hover table-bordered">
148         <thead>
149             <tr>
150                 <th scope="col" class="text-center">Fabricante</th>
151                 <th scope="col" class="text-center">Modelo</th>
152                 <th scope="col" class="text-center">Color</th>
153                 <th scope="col" class="text-center">Año</th>
154             </tr>
155         </thead>
156         <tbody>
157             <tr>
158                 <td>${carro.fabricante}</td>
159                 <td>${carro.modelo}</td>
160                 <td>${carro.color}</td>
161                 <td>${carro.anio}</td>
162             </tr>
163         </tbody>
164     </table>
165     `;
166
167     document.getElementById("idDivResultado").innerHTML = tblCarro;
168 };
169
170 const showRadioSelected = function (radiogroup) {
171     let seleccionado;
172     let numradios = radiogroup.length;
173     for (let i = 0; i < numradios; i++) {
174         if (radiogroup[i].checked) {
175             seleccionado = radiogroup[i].value;
176         }
177     }

```


	UNIVERSIDAD DON BOSCO FACULTAD DE INGENIERÍA ESCUELA DE COMPUTACIÓN		CICLO 02-2022
	MATERIA	LENGUAJES INTERPRETADOS EN EL CLIENTE	
	PRÁCTICA	OBJETOS EN JAVASCRIPT	
		GUIA DE LABORATORIO N° 8	

```

178     return seleccionado;
179 };
180
181 const removeOptions = function (optionMenu) {
182     for (i = 0; i < optionMenu.options.length; i++) {
183         optionMenu.options[i] = null;
184     }
185 };
186
187 const addOptions = function (optionList, optionMenu) {
188     let i = 0;
189     removeOptions(optionMenu); //Limpia las opciones
190     for (i = 0; i < optionList.length; i++) {
191         optionMenu[i] = new Option(optionList[i], optionList[i]);
192     }
193 };
194
195 //Asociando función que manejará el evento load al cargar la página
196 if (window.addEventListener) {
197     window.addEventListener("load", iniciar, false);
198 } else if (window.attachEvent) {
199     window.attachEvent("onload", iniciar);
200 }

```

2. Verifique el funcionamiento de su página ejemplo5.html, tendría que obtener un resultado como el siguiente.



Objetos en JavaScript

Ejemplo 1
Ejemplo 2
Ejemplo 3
Ejemplo 4
Ejemplo 5
Ejemplo 6

Datos del carro

☐ Blanco
☐ Negro
☐ Gris
☐ Azul
☐ Verde
☐ Rojo
☐ Amarillo
☐ Beige

Vehiculos registrados

	UNIVERSIDAD DON BOSCO FACULTAD DE INGENIERÍA ESCUELA DE COMPUTACIÓN		CICLO 02-2022
	MATERIA	LENGUAJES INTERPRETADOS EN EL CLIENTE	GUIA DE LABORATORIO N° 8
	PRÁCTICA	OBJETOS EN JAVASCRIPT	

Ejemplo 6: Registro de matrícula para estudiantes.

El siguiente ejemplo simula el proceso de matrícula de un alumno. Para ello se llenan los campos de un formulario con datos básicos. Con un script JS se obtienen los datos ingresados en los campos del formulario y se crean propiedades para un alumno.

1. Localice el archivo ejemplo6.js y comience a desarrollar el siguiente código

```

1  //Registrar evento click al presionar el botón de envío
2  const iniciar = function () {
3      const btnenviar = document.getElementById("idBtnEnviar");
4      document.frmmat.txtname.focus();
5      //Al producirse un click sobre el botón de envío
6      //invocar los métodos del objeto carro que mostrarán
7      //los valores ingresados en el formulario
8      if (btnenviar.addEventListener) {
9          btnenviar.addEventListener(
10             "click",
11             function () {
12                 var chkvalue, selvalue, nuevoalumno;
13                 var radiofield = document.frmmat.elements["chkgender"];
14                 for (var i = 0; i < radiofield.length; i++) {
15                     if (radiofield[i].checked) {
16                         chkvalue = radiofield[i].value;
17                     }
18                 }
19                 selvalue =
20                     document.frmmat.seldegree.options[frmmat.seldegree.selectedIndex]
21                     .value;
22
23                 //Verificando que los datos se encuentren llenos
24                 if (
25                     document.frmmat.txtname.value != "" &&
26                     document.frmmat.txtlastname.value != "" &&
27                     document.frmmat.txtage.value != "" &&
28                     selvalue != ""
29                 ) {
30                     nuevoalumno = new alumnoUDB(
31                         document.frmmat.txtname.value,
32                         document.frmmat.txtlastname.value,
33                         document.frmmat.txtage.value,
34                         chkvalue,
35                         selvalue
36                     );
37                     nuevoalumno.matricular();
38                     nuevoalumno.imprimir();
39                 } else {
40                     alert("Faltan campos por completar");
41                 }
42             },
43             false
44         );

```




UNIVERSIDAD DON BOSCO
FACULTAD DE INGENIERÍA
ESCUELA DE COMPUTACIÓN

CICLO
02-2022

MATERIA

LENGUAJES INTERPRETADOS EN EL CLIENTE

PRÁCTICA

OBJETOS EN JAVASCRIPT

GUIA DE
LABORATORIO N° 8

```
45     } else {
46         btnenviar.attachEvent("onclick", function () {
47             var chkvalue, nuevoalumno;
48             var radiofield = document.frmmat.elements["chkgender"];
49             for (var i = 0; i < radiofield.length; i++) {
50                 if (radiofield[i].checked) {
51                     chkvalue = radiofield[i].value;
52                 }
53             }
54             selvalue =
55                 document.frmmat.seldegree.options[frmmat.seldegree.selectedIndex].value;
56             nuevoalumno = new alumnoUDB(
57                 document.frmmat.txtnome.value,
58                 document.frmmat.txtlastname.value,
59                 document.frmmat.txtage.value,
60                 chkvalue,
61                 selvalue
62             );
63             nuevoalumno.matricular();
64             nuevoalumno.imprimir();
65         });
66     }
67 };
68
69 //Definiendo la clase alumnoUDB haciendo uso de sintaxis de función
70 function alumnoUDB(nombre, apellido, edad, genero, carrera) {
71     //Propiedades de la clase
72     this.nombre = nombre;
73     this.apellido = apellido;
74     this.edad = edad;
75     this.genero = genero;
76     this.carrera = carrera;
77     this.numCarnet = null;
78     //Métodos de la clase
79     this.matricular = function () {
80         var fecha = new Date();
81         var year = fecha.getFullYear();
82         var day = fecha.getDate();
83         var sec = fecha.getSeconds();
84         this.numCarnet =
85             this.nombre.substring(0, 1) +
86             this.apellido.substring(0, 1) +
87             this.formato(sec) +
88             this.formato(day) +
89             year;
90     };
91 }
```




UNIVERSIDAD DON BOSCO
FACULTAD DE INGENIERÍA
ESCUELA DE COMPUTACIÓN

CICLO
02-2022

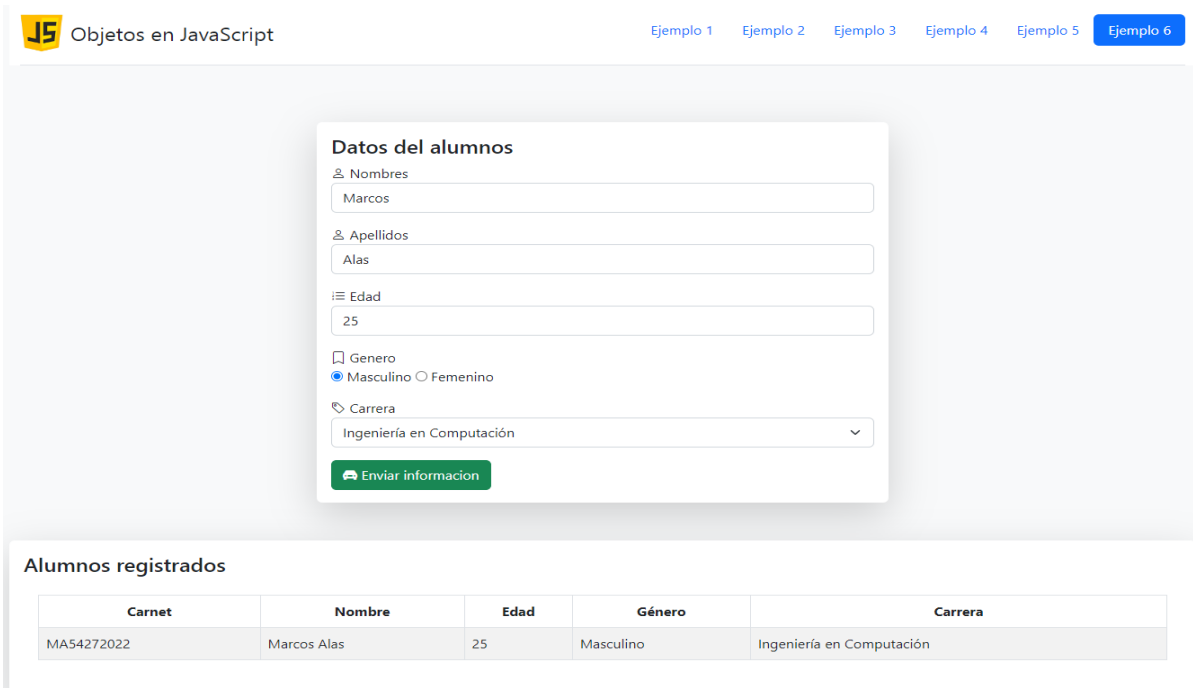
MATERIA	LENGUAJES INTERPRETADOS EN EL CLIENTE
PRÁCTICA	OBJETOS EN JAVASCRIPT

GUIA DE
LABORATORIO N° 8

```
91     this.imprimir = function () {
92         let tblAlumno = "";
93         tblAlumno = `
94             <table class="table table-striped table-hover table-bordered">
95                 <thead>
96                     <tr>
97                         <th scope="col" class="text-center">Carnet</th>
98                         <th scope="col" class="text-center">Nombre</th>
99                         <th scope="col" class="text-center">Edad</th>
100                        <th scope="col" class="text-center">Género</th>
101                        <th scope="col" class="text-center">Carrera</th>
102                    </tr>
103                </thead>
104                <tbody>
105                    <tr>
106                        <td>${this.numCarnet}</td>
107                        <td>${this.nombre} ${this.apellido}</td>
108                        <td>${this.edad}</td>
109                        <td>${this.genero}</td>
110                        <td>${this.carrera}</td>
111                    </tr>
112                </tbody>
113            </table>
114        `;
115
116        document.getElementById("idDivResultado").innerHTML = tblAlumno;
117    };
118    this.formato = function (valor) {
119        if (valor < 10) valor = "0" + valor;
120        return valor;
121    };
122 }
123
124 //Asociando función que manejará el evento load al cargar la página
125 if (window.addEventListener) {
126     window.addEventListener("load", iniciar, false);
127 } else if (window.attachEvent) {
128     window.attachEvent("onload", iniciar);
129 }
130
```

	UNIVERSIDAD DON BOSCO FACULTAD DE INGENIERÍA ESCUELA DE COMPUTACIÓN		CICLO 02-2022
	MATERIA	LENGUAJES INTERPRETADOS EN EL CLIENTE	
	PRÁCTICA	OBJETOS EN JAVASCRIPT	
			GUIA DE LABORATORIO N° 8

- Verifique el funcionamiento de su página ejemplo6.html, tendría que obtener un resultado como el siguiente.



Objetos en JavaScript

Ejemplo 1 Ejemplo 2 Ejemplo 3 Ejemplo 4 Ejemplo 5 **Ejemplo 6**

Datos del alumnos

Nombres
Marcos

Apellidos
Alas

Edad
25

Genero
☒ Masculino
 ☐ Femenino

Carrera
Ingeniería en Computación


Enviar información

Alumnos registrados

Carnet	Nombre	Edad	Género	Carrera
MA54272022	Marcos Alas	25	Masculino	Ingeniería en Computación

IV. EJERCICIOS COMPLEMENTARIOS

- Realice una aplicación orientada a objetos llamada potencia. Su aplicación debe crear una clase utilizando funciones, en donde las propiedades deben ser tres: la base y la potencia que deben obtener sus valores de dos argumentos enviados a la definición de la función que hará las veces de la clase. Una tercera propiedad que tendrá el resultado de la operación realizada por la base y potencia. Su clase debe poseer dos métodos, uno para retornar el cálculo de la potencia con las propiedades de la clase y otro para mostrar el resultado del cálculo, utilice la propiedad innerHTML para mostrar el resultado en su página web. Debe realizar una interfaz de formulario apropiada para este problema, no es permitido utilizar el método prompt para obtener los datos.
- Siguiendo la metodología de los ejemplos 5 y 6 (función anónima u objeto Object) del procedimiento, cree un objeto llamado Persona que permita registrar la información de los usuarios de un sitio web. El usuario debe ingresar la siguiente información: nombre, apellido, fecha de nacimiento, correo electrónico, nombre de usuario y contraseña. Deberá de crear dos botones, uno para almacenar la información en el objeto Persona y el siguiente botón permitirá mostrar la información que ha sido registrada en el objeto Persona. Adicionalmente cree una propiedad identificadora, la cual almacenará el id de cada Persona, para la generación de este id, deberá tomar las siguientes consideraciones: las iniciales de los dos apellidos (si sólo tiene un apellido duplicar la primera letra del primer apellido) concatenadas con el año de registro y un número aleatorio de 4 cifras que deberá crear mediante el uso del método random del objeto Math. Utilice un formulario para la captura de datos y muestre el resultado en una tabla.

	UNIVERSIDAD DON BOSCO FACULTAD DE INGENIERÍA ESCUELA DE COMPUTACIÓN		CICLO 02-2022
	MATERIA	LENGUAJES INTERPRETADOS EN EL CLIENTE	GUIA DE LABORATORIO N° 8
	PRÁCTICA	OBJETOS EN JAVASCRIPT	

V. BIBLOGRAFÍA

- Flanagan, David. JavaScript La Guía Definitiva. 1ª Edición. Editorial ANAYA Multimedia / O'Reilly. 2007. Madrid, España.
- Terry McNavage. JavaScript Edición 2012. 1ª Edición. Editorial ANAYA Multimedia / Apress. Octubre 2011. Madrid, España.
- Tom Negrino / Dori Smith. JavaScript & AJAX Para Diseño Web. 6ª Edición. Editorial Pearson – Prentice Hall. 2007. Madrid España.
- Powell, Thomas / Schneider, Fritz. JavaScript Manual de Referencia. 1ra Edición. Editorial McGraw-Hill. 2002. Madrid, España.
- McFedries, Paul. JavaScript Edición Especial. 1ra Edición. Editorial Prentice Hall. 2002. Madrid, España.