

## Desafío - JavaScript II

---

### Instrucciones

- Crea e implementa funcionalidades de JavaScript en un sitio web.
- Sube tu respuesta a la plataforma Empieza.

### Ejercicio 1:

---

- Identificar y solucionar el problema en el siguiente bloque de código para que se muestre la suma de los números ingresados:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title></title>
  </head>
  <body>
    <script type="text/javascript">
      a = prompt("ingresa un número");
      b = prompt("ingresa otro número");
      alert("la suma es " + a + b );
    </script>
  </body>
</html>
```

## Ejercicio 2

---

- Identificar el/los problema(s) en el siguiente bloque de código.
- El *alert* debe mostrar el mensaje "Eres mayor de edad" si la edad ingresada fue igual o mayor a 18.
- De lo contrario debe mostrar "Eres menor de edad".

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title></title>
  </head>
  <body>
    <script>
      prompt("Ingresa tu edad");
      if (edad = 18){
        alert(Eres mayor de edad);
      }
    </script>
  </body>
</html>
```

## Ejercicio 3

---

- El siguiente código muestra 'hola' varias veces, pero debería mostrar todos los días especificados en el array, además estos días deben mostrados como elementos de una lista ordenada.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title></title>
  </head>
  <body>
    <ol id="dias">
    </ol>
    <script type="text/javascript">
      var dias = ["Lunes", "Martes", "Miércoles", "Jueves", "Viernes"]
      var parent_tag = document.getElementById('dias');
      dias.map(function(x){
        var doc_element = document.createElement("p");
        node = document.createTextNode("HOLA");
        doc_element.appendChild(node);
        parent_tag.appendChild(doc_element);
      });
    </script>
  </body>
</html>
```

## Ejercicio 4

- Utilizando la sentencia **if** detener el conteo regresivo cuando el valor llegue a cero y cambiar el texto de bomba por "BOOM!" y el background-color de body a rojo.

Hint: Busca la documentación del método **clearInterval()**

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title></title>
  </head>
  <body>
    <h1> ¡¡¡ La bomba !!! </h1>
    <h2 id="timer">15</h2>
    <script>
      var timer_tag = document.getElementById('timer');
      var timer = timer_tag.innerHTML;
      timer = parseInt(timer);
      setInterval(function(){
        timer = timer - 1
        timer_tag.innerHTML = timer
      }, 1000);
    </script>
  </body>
</html>
```

## Ejercicio 5

- El siguiente código corresponde a una calculadora incompleta.
- El botón '=' funciona evaluando la expresión dentro del input de texto.
- Se requiere:
  - Al presionar un dígito u operador este debe ser agregado al input.
  - Agregar los dígitos faltantes.
  - Agregar operadores faltantes (multiplicación y división).
  - Agregar un botón para limpiar el input.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title></title>
  </head>
  <body>
    <input type="text" name="calculator" value="">
    <br>
    <button class="btn" type="button" name="button">1</button>
    <button class="btn" type="button" name="button">2</button>
    <button class="btn" type="button" name="button">3</button>
    <button class="btn" type="button" name="button">4</button>
    <button class="btn" type="button" name="button">5</button>
    <button class="btn" type="button" name="button">6</button>
    <br>
    <button class="btn" type="button" name="button">+</button>
    <button class="btn" type="button" name="button">-</button>
    <br>
    <button class="result" type="button" name="result" onclick="calcTotal()">=</button>
    <script type="text/javascript">
      var result = document.getElementsByName("result")[0];
      var calculator = document.getElementsByName("calculator")[0];
      var calcTotal = function(){
        calculator.value = eval(calculator.value);
      }
    </script>
  </body>
</html>
```

## Ejercicio 6

---

- Se tiene la siguiente vista y script que incluye un arreglo.

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <title></title>
</head>
<body>
  <button>Clasificar Arreglo</button>
  <ul id="par">
    <strong>Pares</strong>
  </ul>
  <ul id="impar">
    <strong>Impares</strong>
  </ul>
  <ul id="string">
    <strong>Strings</strong>
  </ul>
  <script>
    var arr = [1,'dos','tres',4,5,6,'siete',8,9,'diez'];
  </script>
</body>
</html>
```

### Se requiere:

- Crear una función llamada sortList() la cual debe ser ejecutada al presionar el botón "Clasificar Arreglo".
- La función debe iterar el arreglo:
  - Si el elemento del array es un número impar debe ser agregado a la lista '**Impares**'.
  - Si el elemento es un número par debe ser agregado a la lista '**Pares**'.
  - Si el elemento es un string debe ser agregado a la lista '**Strings**'.
- Los elementos deben ser agregados de manera dinámica manipulando el **DOM** mediante **JS**.

### Screenshot referencia:



Comer una empanada

Last updated 3 mins ago

Listo



Bailar una patita

Last updated 3 mins ago

Listo



Tomarse un terremoto

Last updated 3 mins ago

Listo



Jugar una pichanga

Last updated 3 mins ago

Listo



Ir una fonda

Last updated 3 mins ago

Listo



Encumbrar un volantín

Last updated 3 mins ago

Listo

## Avanzado (opcional)

---

- Se tiene la siguiente vista:

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <title>JS Basics</title>
  <style>
    .todo{
      float: left;
      cursor: pointer;
    }

    .done{
      float: left;
      margin-left: 100px;
    }
  </style>
</head>
<body>

<ol class="todo" id="todo">
  <h3>To Do:</h3>
</ol>

<ol class="done" id="done">
  <h3>Done:</h3>
</ol>
</body>
</html>
```

- Y el siguiente arreglo de objetos:

```
<script>
var asdf = [{
  "completed":false,
  "id":1,
  "title":"Delectus aut autem",
},
{
  "completed":false,
  "id":2,
  "title":"Quis ut nam facilis et officia qui",
},
{
  "completed":false,
  "id":3,
```



```

        "title":"Fugiat veniam minus",
    },
    {
        "completed":true,
        "id":4,
        "title":"Et porro tempora",
    },
    {
        "completed":false,
        "id":5,
        "title":"Laboriosam mollitia et enim quasi adipisci quia provident illum",
    },
    {
        "completed":false,
        "id":6,
        "title":"Qui ullam ratione quibusdam voluptatem quia omnis",
    },
    {
        "completed":false,
        "id":7,
        "title":"Illo expedita consequatur quia in",
        "userId":1
    },
    {
        "completed":true,
        "id":8,
        "title":"Quo adipisci enim quam ut ab",
    },
    {
        "completed":false,
        "id":9,
        "title":"Molestiae perspiciatis ipsa",
    },
    {
        "completed":true,
        "id":10,
        "title":"Illo est ratione doloremque quia maiores aut",
    }
  ];
</script>

```

### Se requiere:

- Agregar al script las instrucciones necesarias para que, al cargar la página, los objetos que poseen **completed:false** sean agregados de manera automática a la lista **"To Do"**.
- Los objetos que posean **completed:true** deben ser agregados a la lista **"Done"**
- Generar un método llamado **completeTask()** que reciba como argumento el **id** del objeto y permita marcar un objeto (tarea) como completado, es decir:
  - El objeto debe cambiar estado a **completed:true**.
  - La tarea debe ser removida de la lista **"To Do"**.
  - La tarea debe ser agregada, de manera dinámica, a la lista **"Done"**.