**Arreglos HTML y JS**

****

**Aprendiz:**

**Andres Felipe Suaza Bustos**

**Instructor:**

**Andres Moreno Collazos**

**Servicio Nacional de Aprendizaje – SENA**

**Centro de la Empresa, la Industria y los Servicios – CIES**

**Regional Huila**

**2024**

**Ejercicio 1**

|  |  |
| --- | --- |
| **Descripción**  Genera en pantalla dos listas de los números del uno al diez | |
| <!DOCTYPE html>  <html lang="es">  <head>      <title>Document</title>      <link rel="stylesheet" href="Bootstrap/css/bootstrap.css">  </head>  <body>      <div class="container">          <h1>Lista Uno</h1>          <ul class="list-group" id="listNum1"></ul>          <h1>Lista Dos</h1>          <ul class="list-group" id="listNum2"></ul>      </div>      <script src="Bootstrap/js/bootstrap.js"></script>      <script src="Bootstrap/js/bootstrap.bundle.js"></script>      <script src="js/script.js"></script>  </body>  </html> | |
| **Variables**   1. array1 2. resultado1 3. index | **Tipo de Variable**   1. array 2. string 3. int |
| let array1 = [1, 2, 3, 4, 5, 6, 7, 8, 10]  let resultado1 = ''  ***for*** (let index = 0; index < array1.length; index++) {      resultado1 += `<li class="list-group-item">${array1[index]}</li>`  }  document.**getElementById**('listNum1').innerHTML = resultado1 | |
| **Variables**   1. array2 2. index 3. resultado2 | **Tipo de Variable**   1. array 2. int 3. string |
| let array2 = []  ***for*** (let index = 0; index < 10; index++) {      array2.**push**(index+1)  }  let resultado2 = ''  ***for*** (let index = 0; index < array2.length; index++) {      resultado2 += `<li class="list-group-item">${array2[index]}</li>`  }  document.**getElementById**('listNum2').innerHTML = resultado2 | |
|  | |

**Ejercicio 2**

|  |  |
| --- | --- |
| **Descripción**  Presenta en pantalla el proceso y el resultado del factorial de los números del uno al cinco | |
| <!DOCTYPE html>  <html lang="es">  <head>      <meta charset="UTF-8">      <meta name="viewport" content="width=device-width, initial-scale=1.0">      <title>Document</title>      <link rel="stylesheet" href="Bootstrap/css/bootstrap.css">  </head>  <body>      <div class="container">          <ul class="list-group list-group-horizontal">              <li class="list-group-item col-4 list-group-item-danger"><h4>Factorial</h4></li>              <li class="list-group-item col-4 list-group-item-danger"><h4>Proceso</h4></li>              <li class="list-group-item col-4 list-group-item-danger"><h4>Resultado</h4></li>          </ul>          <div id="list-groups">          </div>      </div>      <script src="Bootstrap/js/bootstrap.js"></script>      <script src="Bootstrap/js/bootstrap.bundle.js"></script>      <script src="js/script.js"></script>  </body>  </html> | |
| **Variable**   1. factorial 2. data 3. index1 4. index2 | **Tipo de Variable**   1. array 2. array 3. int 4. int |
| let factorial = []  let data = []  ***for*** (let index1 = 1; index1 <= 5; index1++) {      data.**push**(index1)      data.**push**(‘’)      data.**push**(index1)  ***for*** (let index2 = index1; index2 >= 1; index2--) {          data[1] += `${index2}`  ***if*** (index2 !== 1) {              data[1] += ‘ X ‘  ***if*** (index2 === data[2]) {  ***continue***              }              data[2] \*= index2          }      }      factorial.**push**(data)      data = []  } | |
| **Variables**   1. screen 2. index 3. factorial | **Tipo de Variable**   1. string 2. int 3. array |
| let screen = ''  ***for*** (let index = 0; index < factorial.length; index++) {      screen += `          <ul class="list-group list-group-horizontal">              <li class="list-group-item col-4">${factorial[index][0]}!</li>              <li class="list-group-item col-4">${factorial[index][1]}</li>              <li class="list-group-item col-4">${factorial[index][2]}</li>          </ul>          `  }  document.**getElementById**('list-groups').innerHTML = screen | |
|  | |

**Ejercicio 3**

|  |  |
| --- | --- |
| **Descripción**  Muestra en pantalla las tablas de multiplicar del uno hasta el cinco | |
| <!DOCTYPE html>  <html lang="en">  <head>      <meta charset="UTF-8">      <meta name="viewport" content="width=device-width, initial-scale=1.0">      <title>Document</title>      <link rel="stylesheet" href="Bootstrap/css/bootstrap.css">  </head>  <body>      <div class="container">          <h1 class="text-center">Tablas de multiplicar</h1>          <div class="row align-items-center" id="tablas">          </div>      </div>      <script src="Bootstrap/js/bootstrap.js"></script>      <script src="Bootstrap/js/bootstrap.bundle.js"></script>      <script src="js/script.js"></script>  </body>  </html> | |
| **Variables**   1. screen 2. index1 3. index2 | **Tipo de Variable**   1. string 2. int 3. int |
| document.**addEventListener**('DOMContentLoaded', function () {      let screen = ''  ***for*** (let index1 = 1; index1 <= 5; index1++) {          screen += `                  <div class="col">                      <div class="card">                          <h5 class="card-header text-center">                              Tabla del ${index1}                          </h5>                          <ul class="list-group list-group-flush">                      `  ***for*** (let index2 = 1; index2 <= 10; index2++) {              screen += `                  <li class="list-group-item">${index1} X ${index2} = ${index1 \* index2}</li>                  `          }          screen += `                          </ul>                      </div>                  </div>                  `      }      document.**getElementById**('tablas').innerHTML = screen  }) | |
|  | |

**Ejercicio 4**

|  |  |
| --- | --- |
| **Descripción**  Genera en pantalla 4 tablas de Bingo con números aleatorios y algunas con unos detalles demás | |
| <!DOCTYPE html>  <html lang="es">  <head>      <meta charset="UTF-8">      <meta name="viewport" content="width=device-width, initial-scale=1.0">      <title>Document</title>      <link rel="stylesheet" href="Bootstrap/css/bootstrap.css">      <link rel="stylesheet" href="css/style.css">  </head>  <body>      <div class="container text-center">          <h2>TABLA BINGO</h2>          <div class="row justify-content-center">              <div class="col-3">                  <table class="table table-bordered border-dark" id="table1">                      <thead class="table-dark">                          <tr>                              <th scope="col-2">B</th>                              <th scope="col-2">I</th>                              <th scope="col-2">N</th>                              <th scope="col-2">G</th>                              <th scope="col-2">O</th>                          </tr>                      </thead>                      <tbody name="nums">                      </tbody>                  </table>              </div>          </div>          <div class="row justify-content-evenly">              <div class="col-3">                  <table class="table table-bordered border-dark" id="table2">                      <thead class="table-dark">                          <tr>                              <th scope="col-2">B</th>                              <th scope="col-2">I</th>                              <th scope="col-2">N</th>                              <th scope="col-2">G</th>                              <th scope="col-2">O</th>                          </tr>                      </thead>                      <tbody name="nums">                      </tbody>                  </table>              </div>              <div class="col-3">                  <table class="table table-bordered border-dark" id="table3">                      <thead class="table-dark">                          <tr>                              <th scope="col-2">B</th>                              <th scope="col-2">I</th>                              <th scope="col-2">N</th>                              <th scope="col-2">G</th>                              <th scope="col-2">O</th>                          </tr>                      </thead>                      <tbody name="nums">                      </tbody>                  </table>              </div>              <div class="col-3">                  <table class="table table-bordered border-dark" id="table4">                      <thead class="table-dark">                          <tr>                              <th scope="col-2">B</th>                              <th scope="col-2">I</th>                              <th scope="col-2">N</th>                              <th scope="col-2">G</th>                              <th scope="col-2">O</th>                          </tr>                      </thead>                      <tbody name="nums">                      </tbody>                  </table>              </div>          </div>      </div>      <script src="Bootstrap/js/bootstrap.js"></script>      <script src="Bootstrap/js/bootstrap.bundle.js"></script>      <script src="js/createTable.js"></script>      <script src="js/drawX.js"></script>      <script src="js/colorLetters.js"></script>  </body>  </html> | |
| .line {      position: absolute;      z-index: 1;      height: 5px;      transform-origin: 0 50%;      border-radius: 50px;      border: 1px solid  }  #X1 {      background-color: rgba(13, 109, 253, 0.5);      border-color: rgba(13, 109, 253);  }  #X2 {      background-color: rgb(25, 135, 84, 0.5);      border-color: rgb(25, 135, 84);  }  #X3 {      background-color: rgb(220, 53, 69, 0.5);      border-color: rgb(220, 53, 69);  }  #X4 {      background-color: rgb(255, 193, 7, 0.5);      border-color: rgb(255, 193, 7);  } | |
| **createTable.js** | |
| **Variables**   1. numsBingo 2. cont 3. index1 4. index2 | **Tipo de Variable**   1. array 2. int 3. int 4. int |
| let numsBingo = [];  let cont = 1;  ***for*** (let index1 = 0; index1 < 5; index1++) {      numsBingo[index1] = [];  ***for*** (let index2 = 0; index2 < 15; index2++) {          numsBingo[index1][index2] = cont;          cont++;      }  } | |
| **Variables**   1. tableBingo 2. index1 3. nums 4. index2 5. num | **Tipo de Variable**   1. array 2. int 3. array 4. int 5. int |
| let tableBingo = [];  ***for*** (let index1 = 0; index1 < 5; index1++) {      let nums = [];  ***for*** (let index2 = 0; index2 < 5; index2++) {          let num = **numRandom**(index1);  ***if*** (num !== null) {  ***if*** (***!***tableBingo[index1]) {                  tableBingo[index1] = [];              }              tableBingo[index1].**push**(num);          } ***else*** {  ***break***; // No hay más números disponibles para esta columna          }      }  ***if*** (nums.length > 0) {          tableBingo[index1] = nums;      }  } | |
| **Variable**   1. index 2. avaibleNumbers 3. randomIndex | **Tipo de Variable**   1. int 2. array 3. int |
| const **numRandom** = function (*index*) {      let availableNumbers = **getAvailableNumbers**(index);  ***if*** (availableNumbers.length === 0) {  ***return*** null; // No hay más números disponibles      }      let randomIndex = Math.**floor**(Math.**random**() \* availableNumbers.length);  ***return*** availableNumbers[randomIndex];  } | |
| **Variables**   1. index 2. avaibleNumbers | **Tipo de Variable**   1. int 2. array |
| const **getAvailableNumbers** = function (*index*) {      let availableNumbers = [];      numsBingo[index].**forEach**(*num* => {  ***if*** (***!***tableBingo[index] ***||*** ***!***tableBingo[index].**includes**(num)) {              availableNumbers.**push**(num);          }      });  ***return*** availableNumbers;  } | |
| **Variables**   1. screen 2. index1 3. index2 4. tables 5. table | **Tipo de Variable**   1. string 2. int 3. int 4. array 5. object |
| screen = ‘’  ***for*** (let index1 = 0; index1 < 5; index1++) {      screen += ‘<tr>’  ***for*** (let index2 = 0; index2 < 5; index2++) {          screen += `<td>${tableBingo[index2][index1]}</td>`      }      screen += ‘</tr>’  }  let tables = document.**getElementsByName**(‘nums’)  tables.**forEach**(*table* => {      table.innerHTML = screen  }) | |
| **drawX.js** | |
| function **createX**() {      // Tabla 2  **drawX**('table2', [          [[0, 0], [4, 4], 'X1'],          [[4, 0], [0, 4], 'X1']      ]);      // Tabla 3  **drawX**('table3', [          [[0, 0], [2, 2], 'X2'],          [[2, 0], [0, 2], 'X2'],          [[2, 0], [4, 2], 'X3'],          [[4, 0], [2, 2], 'X3'],          [[0, 2], [2, 4], 'X4'],          [[2, 2], [0, 4], 'X4']      ]);  } | |
| **Variables**   1. tableId 2. coordinates 3. table 4. x1 5. y1 6. x2 7. y2 8. id 9. start 10. end | **Tipo de Variable**   1. parameter 2. parameter 3. object 4. parameter 5. parameter 6. parameter 7. parameter 8. parameter 9. object 10. object |
| function **drawX**(*tableId*, *coordinates*) {      // trae la tabla deseada del documento      let table = document.**getElementById**(tableId);      coordinates.**forEach**(([[*x1*, *y1*], [*x2*, *y2*], *id*]) => {          //trae los dos elementos necesarios para hacer la ínea          let start = table.children.nums.children[x1].cells[y1];          let end = table.children.nums.children[x2].cells[y2];  **drawLine**(start, end, id);      });  } | |
| **Variable**   1. start 2. end 3. id 4. point1 5. point2 6. x1 7. y1 8. x2 9. y2 10. length 11. m1 12. m2 13. angle 14. line | **Tipo de Variable**   1. parameter 2. parameter 3. parameter 4. object 5. object 6. float 7. float 8. float 9. float 10. float 11. float 12. int 13. float 14. object |
| function **drawLine**(*start*, *end*, *id*) {      // Extrae las coordenadas de cada elemento en el documento      let point1 = start.**getBoundingClientRect**()      let point2 = end.**getBoundingClientRect**()      // Calcula las coordenadas del punto central de cada elemento      let x1 = point1.left + point1.width / 2      let y1 = point1.top + point1.height / 2      let x2 = point2.left + point2.width / 2      let y2 = point2.top + point2.height / 2      // Calcula la distancia que existe entre dos puntos de una recta      let length = ((x2 - x1) \*\* 2 + (y2 - y1) \*\* 2) \*\* (1 / 2)      // Calcula el angulo que se origina entre dos puntos de una recta      let m1 = (y2 - y1) / (x2 - x1)      let m2 = 0      let angle = Math.**atan**((m1 - m2) / (1 + m1 \* m2)) \* 180 / Math.PI      // Crea la linea y la ubica en el documento en la posicion y el angulo calculado      let line = document.**createElement**('div')      line.id = `${id}`      line.className = 'line'      line.style.width = `${length}px`      line.style.transform = `rotate(${angle}deg)`      line.style.left = `${x1}px`      line.style.top = `${y1}px`      document.body.**appendChild**(line)  } | |
| **Variables**   1. lines 2. line | **Tipo de Variable**   1. array 2. parameter: object |
| window.**addEventListener**('resize', () => {  **setTimeout**(() => {          // Elimina todas las líneas existentes          let lines = document.**querySelectorAll**('.line');          lines.**forEach**(*line* => line.**remove**())          // crea nuevamente las lineas  **createX**()      }, 500);  }); | |
| **colorLetters.js** | |
| **Variables**   1. table4 2. index1 3. index2 4. cell | **Tipo de Variable**   1. object 2. int 3. int 4. object |
| function **colorLetter**() {      let table4 = document.**getElementById**('table4')  ***for*** (let index1 = 0; index1 < 5; index1++) {  ***for*** (let index2 = 0; index2 < 5; index2++) {              let cell = table4.children.nums.children[index1].cells[index2]  ***switch*** (true) {  ***case*** index2 === 0:                      cell.classList.**add**('table-primary')  ***break***  ***case*** index2 === 1:                      cell.classList.**add**('table-success')  ***break***  ***case*** index2 === 2:                      cell.classList.**add**('table-danger')  ***break***  ***case*** index2 === 3:                      cell.classList.**add**('table-warning')  ***break***  ***case*** index2 === 4:                      cell.classList.**add**('table-info')  ***break***              }              cell.classList.**add**('border-dark')          }      }  } | |
|  | |