Tema:

Arreglos

Instructor:

Andrés Moreno Collazos

Estudiante:

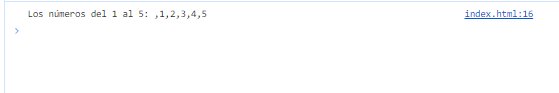
Jhampier Santos Ortiz

Neiva-Huila

2024

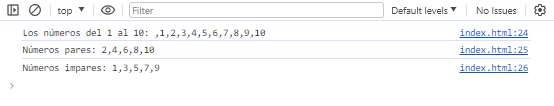
Ejercicio de array

| **Nombre del arreglo:** | |
| --- | --- |
| **Descripción: Arreglo que almacena los primeros 5 números.** | | |
| Longitud  Números |  | |
| let numeros = [];          for (let longitud = 1; longitud <= 5; longitud++) {              numeros[longitud]=longitud; *//Agregar el número al array 'numeros'* }          console.log("Los números del 1 al 5: " + numeros); | | |

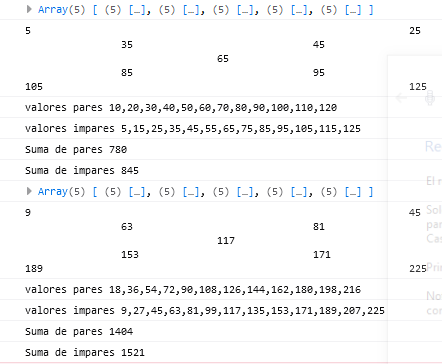




| **Nombre del arreglo:** | |
| --- | --- |
| **Descripción: Arreglo que almacena los números del 1/10**  **Y saber si son pares o impares** | | |
| Numeros  Pares  Impares  Longitud |  | |
| let numeros = [];          let pares = [];          let impares = [];          for (let longitud = 1; longitud <= 10; longitud++) {              numeros[longitud]=longitud; *// Agregar el número al array 'numeros'*              if (longitud % 2 === 0) {                  pares.push(longitud); *// Agregar el número par al array 'pares'*              } else {                  impares.push(longitud); *// Agregar el número impar al array 'impares'*              }          }          console.log("Los números del 1 al 10: " + numeros);          console.log("Números pares: " + pares);          console.log("Números impares: " + impares); | | |

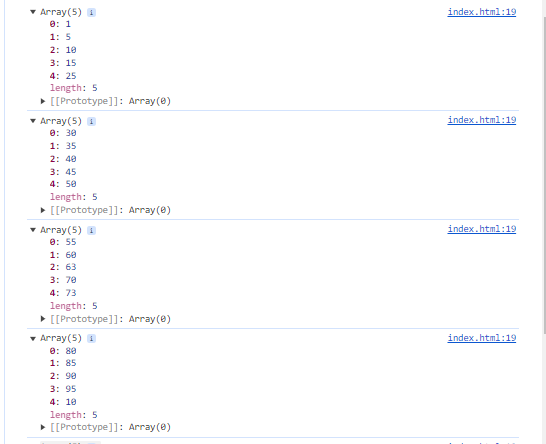


| **Nombre del arreglo:** | |
| --- | --- |
| **Descripción: Multiplicacion de la tabla 5 y 9, saber si son**  **Pares o impares** | | |
| Par  Contar1  Contar2  NúmeroUno  NumeroDos  Arreglo1  Arreglo2  Resultado  Impar |  | |
| *//Tabla del 5*  let par5 = [];  let impar5 = [];  let sumaimpar5;  let sumapar5;  let num3=1;  let matriX1 = [];    let matriX2 = "";  for (let num1 = 0; num1 < 5; num1++) {  matriX1[num1] = [];  for (let num2 = 0; num2 < 5; num2++) {  matriX1[num1][num2] = num3 \* 5;  num3++;  if (matriX1[num1][num2] % 2 == 0) {  par5.push(matriX1[num1][num2]);  } else {  impar5.push(matriX1[num1][num2]);  }  if (num1 === num2 || num1 + num2 === 4) {  matriX2 += matriX1[num1][num2] + "\t\t";  } else {  matriX2 += "\t\t";  }  }  matriX2 += "\n";  }    sumapar5 = par5.reduce((sum, current) => sum + current, 0);  sumaimpar5 = impar5.reduce((sum, current) => sum + current, 0);    console.log(matriX1);  console.log(matriX2);  console.log("valores pares " + par5);  console.log("valores impares " + impar5);  console.log("Suma de pares " + sumapar5);  console.log("Suma de impares " + sumaimpar5);    *//Tabla del 9*  let par9 = [];  let impar9 = [];  let sumaimpar9;  let sumapar9;  let num4=1;  let matriXx9 = [];    let matriX9 = "";  for (let num1 = 0; num1 < 5; num1++) {  matriXx9[num1] = [];  for (let num2 = 0; num2 < 5; num2++) {  matriXx9[num1][num2] = num4 \* 9;  num4++;  if (matriXx9[num1][num2] % 2 == 0) {  par9.push(matriXx9[num1][num2]);  } else {  impar9.push(matriXx9[num1][num2]);  }  if (num1 === num2 || num1 + num2 === 4) {  matriX9 += matriXx9[num1][num2] + "\t\t";  } else {  matriX9 += "\t\t";  }  }  matriX9 += "\n";  }    sumapar9 = par9.reduce((sum, current) => sum + current, 0);  sumaimpar9 = impar9.reduce((sum, current) => sum + current, 0);    console.log(matriXx9);  console.log(matriX9);  console.log("valores pares " + par9);  console.log("valores impares " + impar9);  console.log("Suma de pares " + sumapar9);  console.log("Suma de impares " + sumaimpar9); | | |

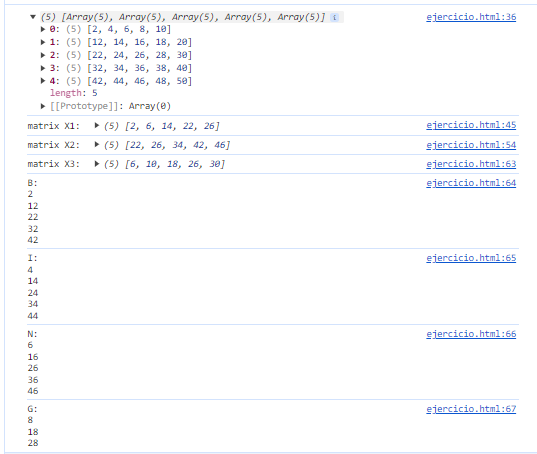


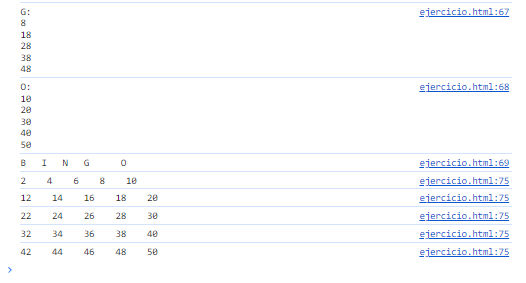


| **Nombre del arreglo:** | |
| --- | --- |
| **Descripción: Matriz** | | |
| Iterar  Matriz  Cantidad |  | |
| let iterar          let matriz =[          [1,5,10,15,25],          [30,35,40,45,50],          [55,60,63,70,73],          [80,85,90,95,10],          [105,110,115,120,125]];          let cantidad= matriz.length          for(iterar=0;iterar<cantidad;iterar++){              console.log(matriz[iterar])          } | | |



| **Nombre del arreglo:** | |
| --- | --- |
| **Descripción: Impresion del BINGO** | | |
| Matrix  MatrizxX1  MatrizxX2  MatrizxX3  Número  Fila  Columna |  | |
| let matrix = [];          let matrixX1 = [];          let matrixX2 = [];          let matrixX3 = [];          let numero = 1;          let B=[]          let I=[]          let N=[]          let G=[]          let O=[]  *//BINGO*          for (let fila = 0; fila < 5; fila++) {              matrix[fila] = [];              for (let columna = 0; columna < 5; columna++) {                  matrix[fila][columna] = numero \* 2;                  numero++;              }          }  *//letras*          for (let fila = 0; fila < 5; fila++) {              B.push("\n"+matrix[fila][0]);              I.push("\n"+matrix[fila][1]);              N.push("\n"+matrix[fila][2]);              G.push("\n"+matrix[fila][3]);              O.push("\n"+matrix[fila][4]);          }          console.log(matrix);  *//MATRIX X1*          for (let fila = 0; fila < 3; fila++) {              for (let columna = 0; columna < 3; columna++) {                  if (fila == columna || fila + columna == 2) {                      matrixX1.push(matrix[fila][columna]);                  }              }          }          console.log("matrix X1: " ,matrixX1);  *//MATRIX X2*           for (let fila = 2; fila < 5; fila++) {              for (let columna = 0; columna < 3; columna++) {    if (fila + columna == (columna + 1) \* 2 || fila + columna == 4) {                      matrixX2.push(matrix[fila][columna]);                  }              }          }          console.log("matrix X2: " ,matrixX2);  *//MATRIX X3*           for (let fila = 0; fila < 3; fila++) {              for (let columna = 2; columna < 5; columna++) {              if (fila + columna == (fila + 1) \* 2 || fila + columna == 4) {                      matrixX3.push(matrix[fila][columna]);                  }              }          }          console.log("matrix X3: " ,matrixX3);          console.log("B: " + B.join(""));          console.log("I: " + I.join(""));          console.log("N: " + N.join(""));          console.log("G: " + G.join(""));          console.log("O: " + O.join(""));          console.log("B   I   N   G      O");          for (let fila = 0; fila < 5; fila++) {              let bingototal= "";              for (let columna = 0; columna < 5; columna++) {                  bingototal+= matrix[fila][columna] + "    ";              }              console.log(bingototal);          } | | |





| **Nombre del arreglo:** | |
| --- | --- |
| **Descripción: Impresión de la tienda** | | |
| Tienda  NumeroRegistro  Interacion  NuevoProducto  ListaBusqueda |  | |
| let tienda=[];       let numeroRegistro;       let interacion;       let nuevoProducto;       let listaBusqueda =[];       tienda=[       {producto:'Arroz', tipProducto:'Granos', tipoUnidad:'Gramos' ,cantidad:'1000', precio:'2450'},       {producto:'Papa', tipProducto:'Fruver', tipoUnidad:'Gramos' ,cantidad:'500', precio:'1000'},       {producto:'Mora', tipProducto:'Fruver', tipoUnidad:'Gramos' ,cantidad:'500', precio:'1500'},       {producto:'Trucha', tipProducto:'Carne', tipoUnidad:'Gramos' ,cantidad:'1000', precio:'9000'},       {producto:'Pollo completo', tipoProducto:'Carne', tipUnidad:'Gramos' ,cantidad:'1000', precio:'4500'},       ];       nuevoProducto = {producto:'res', tipProducto: 'Carne', tipUnidad: 'gramos', cantidad: '1000', precio: '7500'};       tienda.push(nuevoProducto)       numeroRegistros =  tienda.length;       for(interacion=0; interacion<numeroRegistros; interacion++){  *//console.log(tienda[iteracion]);*          if(tienda[interacion].tipProducto =='Carne'){              listaBusqueda.push(tienda[interacion]);          } else{            console.log(" ");         }       }  console.log(listaBusqueda);  console.log(tienda); | | |



| **Nombre del arreglo:** | |
| --- | --- |
| **Descripción: Imprimir el salario de la persona** | | |
| Tienda  Personas  Nomina  SalarioMin  numeroRegistros  Mostrar  MostrarRetencion  SalarioMin |  | |
| let tienda=[];  let personas = [];  let nomina = [];  let salarioMin=1600000  let numeroRegistros;  let mostrar;  let mostrarRetencion;  const funcionSalario = function(paValorDia, paDias) {      return paDias \* paValorDia;  };  const funcioSubtransporte = function(paSalario){      if(paSalario<salarioMin\*2){          paSalario= paSalario+120000          mostrar= "Si se aplica el subsidio " + paSalario      }else{          paSalario= paSalario+0          mostrar= "No se aplica el subsidio " + paSalario  }  return mostrar  }  const funcionSalud=function(paSalario){      return paSalario \* 0.008  }  const funcionPension=function(paSalario){      return paSalario \* 0.12  }  const funcionArl=function(paSalario){      return paSalario \* 0.05  }  const funcionDeducible = function(paSalario){      return funcionSalud(paSalario) + funcionPension(paSalario) + funcionArl(paSalario)  }  const funcionRetener = function(paSalario) {      let mostrarRetencion;      if (paSalario > salarioMin \* 12) {          paSalario = paSalario \* 0.08;          mostrarRetencion = "Retencion de 0.08 " + paSalario;      } else if (paSalario > salarioMin \* 8) {          paSalario = paSalario \* 0.04;          mostrarRetencion = "Retencion de 0.04 " + paSalario;      } else if (paSalario > salarioMin \* 6) {          paSalario = paSalario \* 0.02;          mostrarRetencion = "Retencion de 0.02 " + paSalario;      } else {          mostrarRetencion = "No aplica retención";      }      return mostrarRetencion;  };  const funcionTotal=function(paSalario){      return paSalario - funcionDeducible(paSalario)  }  .log(tienda); | | |

