**TEMA: ARREGLOS**

**INSTRUCTOR: ANDRES MORENO COLLASOS**

**GRUPO TECNOLOGICO: ADSO**

**FICHA: 2899747**

**NOMBRE: Cristofer David Lozano Contreras**

**c.c:1075322976**

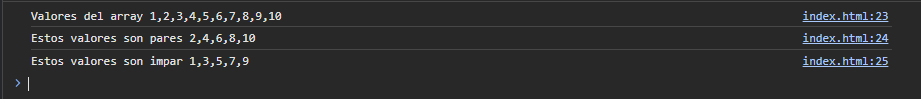
**SENA DE LA INDUSTIRIA Y DE LA EMPRESA DE LOS SERVICIOS**

**NEIVA/HUILA**

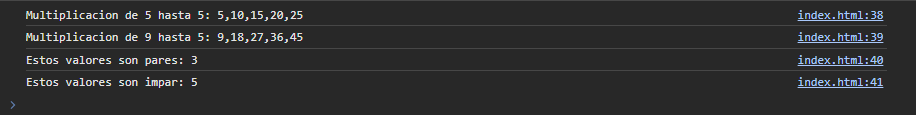
|  |  |  |
| --- | --- | --- |
| **Nombre de arreglo: arreglo** | | **Versión: 1.0** |
| **Descripción:**  Ejercicio de imprimir en consola desde posición 0 a posición 5. | | |
| **arreglo**  **cantidad**  **interar** | Tipo de variable:  Arreglo (Array int)  cantidad (int)  interar (int) | |
| Código:         <script>          let arreglo =[1,2,3,4,5];          let cantidad =arreglo.length;          let interar;          for(interar=0;interar<cantidad;interar++){              console.log(arreglo[interar]);          }      </script> | | |



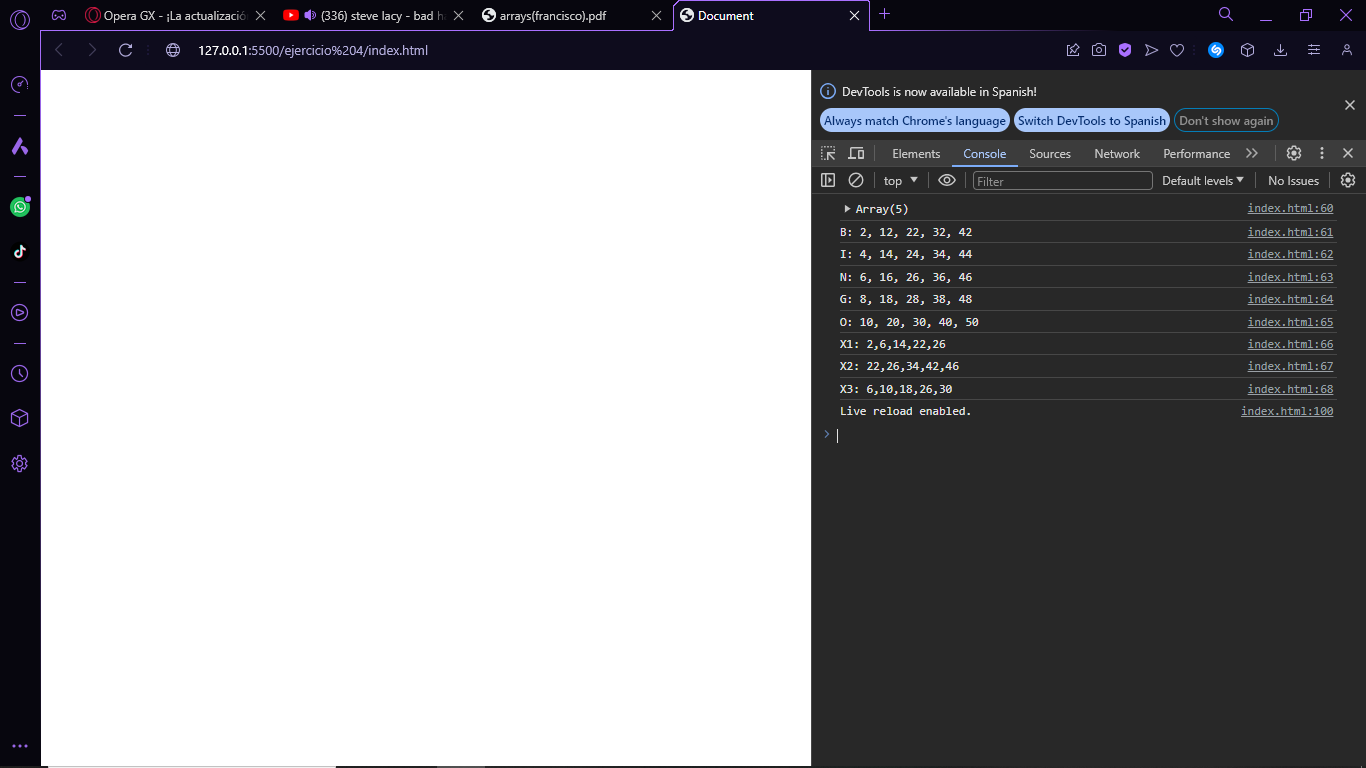
|  |  |  |
| --- | --- | --- |
| **Nombre de los arreglos: arreglo, pares, impares** | | **Versión: 1.0** |
| **Descripción:**  Contar los números del 1 al 10.   * En un arreglo agregar los números pares * En un arreglo agregar los números impares | | |
| **arreglo**  **par**  **impar** | Tipo de variable:  arreglo (array int)  cantidad (int)  interar (int)  par (array int)  impar (array int) | |
| Código:  let arreglo =[1,2,3,4,5,6,7,8,9,10];          let cantidad =arreglo.length;          let interar;          let par=[];          let impar=[];          for(interar=0;interar<cantidad;interar++){              if(arreglo[interar]%2==0){                  par.push(arreglo[interar]);              }else{                  impar.push(arreglo[interar]);              }          }          console.log("Valores del array "+ arreglo);          console.log("Estos valores son pares "+par);              console.log("Estos valores son impar "+impar); | | |



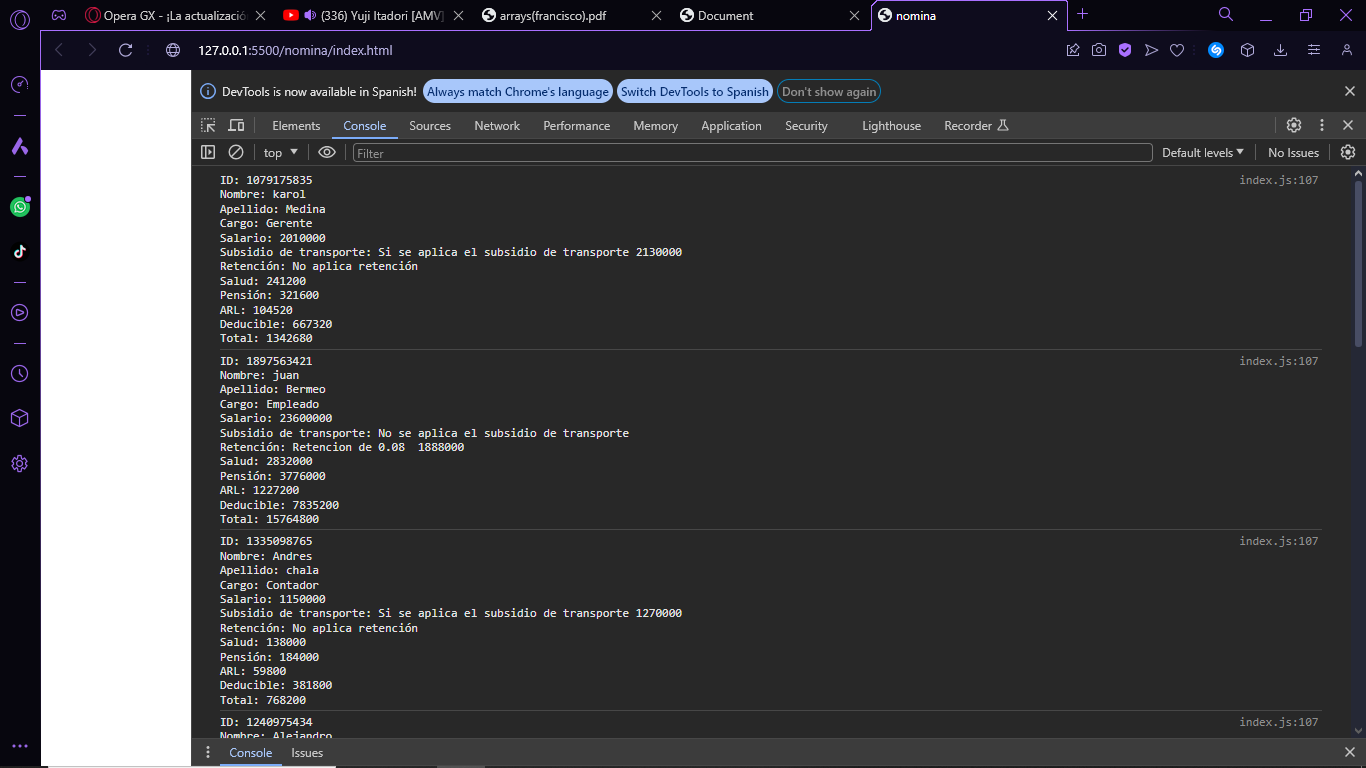
|  |  |  |
| --- | --- | --- |
| **Nombre de los arreglos: arreglo, arreglo2,par,impar** | | **Versión: 1.0** |
| **Descripción:**  Realizar una matriz de 5x5 de las tablas de multiplicar de 5 y el 9, sumar los números pares e impares. | | |
| **arreglo, arreglo2,**  **par,**  **impar** | Tipo de variable:  arreglo (array int)  arreglo2 (array int)  num1 (int)  num2 (int)  i (int)  e (int) | |
| Código:  <script>          let arreglo=[];          let arreglo2=[];          let par=[];          let impar=[];          let num1=1;          let num2=5;          let i = 0;          let e =0;          for (num1 ; num1 <= num2; num1++) {              resul=num1\*5;              resul2=num1\*9;              arreglo.push(resul)              arreglo2.push(resul2)              if (resul%2==0) {                  i++                  par =i;              } else{                  e++                  impar=e;              }              if (resul2%2==0) {                  i++                  par.i;              } else{                  e++                  impar.e;              }          }              console.log("Multiplicacion de 5 hasta 5: "+ arreglo);              console.log("Multiplicacion de 9 hasta 5: "+ arreglo2);              console.log("Estos valores son pares: "+par);              console.log("Estos valores son impar: "+impar);      </script> | | |

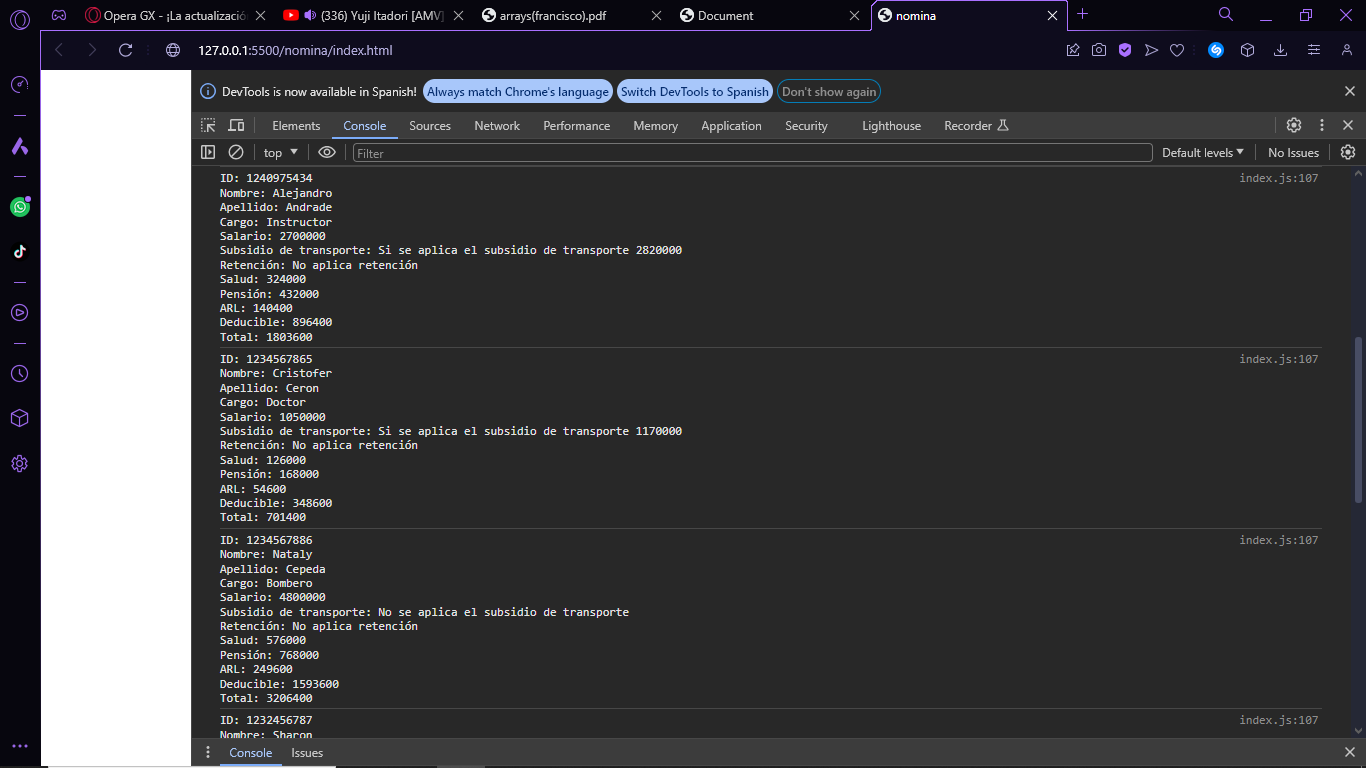


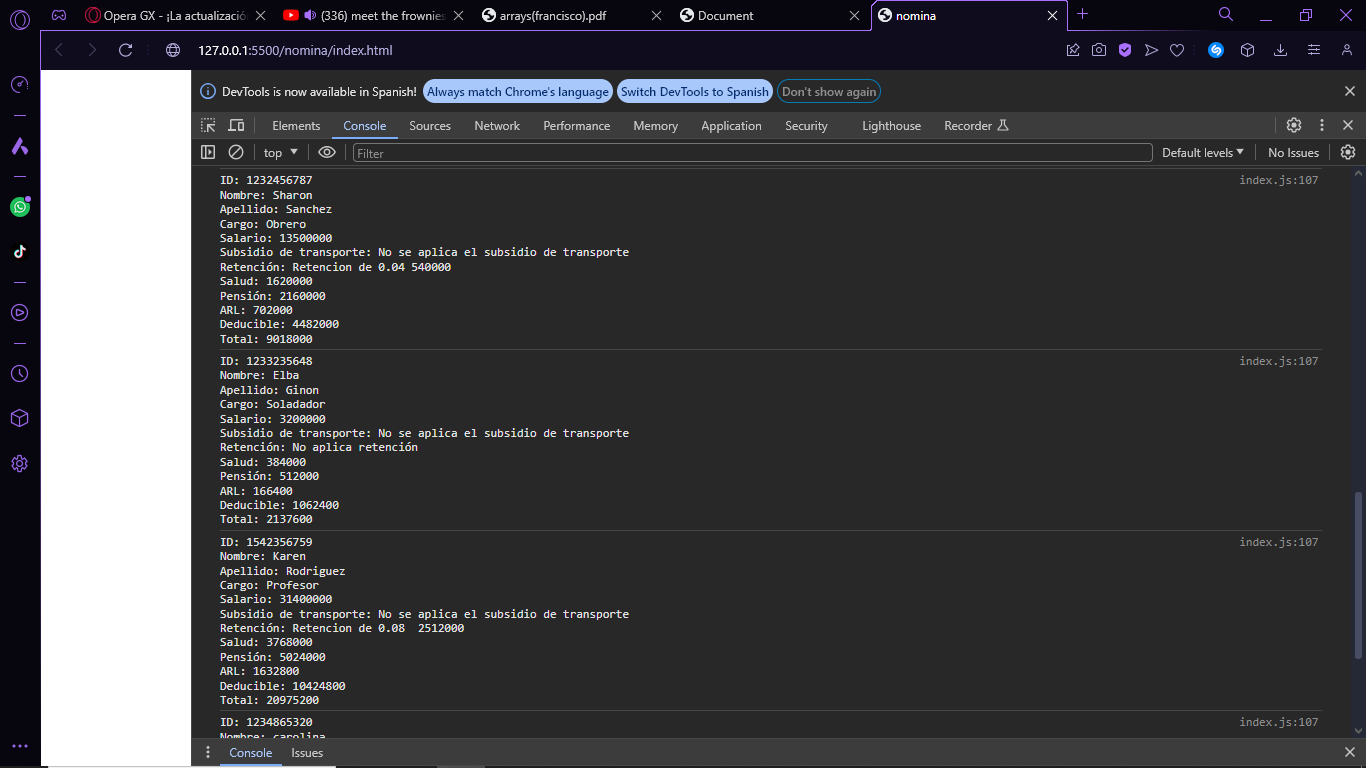
|  |  |  |
| --- | --- | --- |
| **Nombre de los arreglos: matriz** | | **Versión: 1.0** |
| **Descripción:**  Imprimir los números que están en x en la matriz | | |
| Matrix matrixX1 matrixX2 matrixX3  B  I  N  G  O  numero | Tipo de variable: Arreglo (array), Valores: Numérico (int) Tipo de variable: Arreglo (array), Valores: Numérico (int) Tipo de variable: Arreglo (array), Valores: Numérico (int) Tipo de variable: Arreglo (array), Valores: Numérico (int) Tipo de variable: Arreglo (array), Valores: Numérico (int) Tipo de variable: Arreglo (array), Valores: Numérico (int) Tipo de variable: Arreglo (array), Valores: Numérico (int) Tipo de variable: Arreglo (array), Valores: Numérico (int) Tipo de variable: Arreglo (array), Valores: Numérico (int) Tipo de variable: Numérico (int) | |
| Código: <script>          let matrix=[]          let matrixX1=[]          let matrixX2=[]          let matrixX3=[]          let B=[]          let I=[]          let N=[]          let G=[]          let O=[]          let numero=1          for(let fila=0;fila<5;fila++){              matrix[fila]=[]              for(let columna=0;columna<5;columna++){                  matrix[fila][columna]=numero \*2;                if (columna === 0) B.push(matrix[fila][columna]);                  else if (columna === 1) I.push(matrix[fila][columna]);                  else if (columna === 2) N.push(matrix[fila][columna]);                  else if (columna === 3) G.push(matrix[fila][columna]);                  else if (columna === 4) O.push(matrix[fila][columna]);                  numero++;          }      }      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]);              }            }      }        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]);              }          }      }      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);          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("X1: "+ matrixX1)          console.log("X2: "+matrixX2)          console.log("X3: "+matrixX3)      </script> | | |



|  |  |  |
| --- | --- | --- |
| **Nombre de los arreglos:** personas, nomina | | **Versión: 1.0** |
| **Descripción:**  • personas: Arreglo que almacenara los datos de las personas  • nomina: Arreglo que almacenara mediante sus funciones, la nómina de las personas | | |
| personas nomina salarioMin numeroRegistros mostrar mostrarRetencion | Tipo de variable: Arreglo (array)  Tipo de variable: Arreglo (array)  Tipo de variable: Numérico (int)  Tipo de variable: Numérico (int)  Tipo de variable: Alfanumérico  Tipo de variable: Alfanumérico | |
| let personas = []; *//el array principal*  let nomina = []; *//el array para almacenar la nomina*  let salarioMin=1600000  let numeroRegistros;  let mostrar;  let mostrarRetencion;  *// Función para calcular salario*  const funcionSalario = function(*paValorDia*, *paDias*) {      return *paDias* \* *paValorDia*;  };  *//funcion que calcula el subsidio de transporte*  const funcioSubtrans = function(*paSalario*){      if(*paSalario*<salarioMin\*2){  *paSalario*= *paSalario*+120000          mostrar= "Si se aplica el subsidio de transporte " + *paSalario*      }else{  *paSalario*= *paSalario*+0          mostrar= "No se aplica el subsidio de transporte "  }  return mostrar  }  *// funcion que calcula la salud*  const funcionSalud=function(*paSalario*){      return *paSalario* \* 0.12  }  *// funcion que calcula la pension*  const funcionPension=function(*paSalario*){      return *paSalario* \* 0.16  }  *// funcion que calcula el arl*  const funcionArl=function(*paSalario*){      return *paSalario* \* 0.052  }  *// funcion que calcula el deducible*  const funcionDeducible = function(*paSalario*){      return funcionSalud(*paSalario*) + funcionPension(*paSalario*) + funcionArl(*paSalario*)  }  *//funcion que calcula la retencion del salario*  const funcionRetener = function(*paSalario*) {      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;  };  *// funcion que clacula el total*  const funcionTotal=function(*paSalario*){      return *paSalario* - funcionDeducible(*paSalario*)  }    *//--------\*\*\*REGISTRO DE PERSONAS\*\*\*--------- //*  personas = [      {id:1079175835 , nombre:'karol', apellido:'Medina',cargo:'Gerente',valorDia:67000 ,diasTrabajado:30 },      {id:1897563421 , nombre:'juan', apellido:'Bermeo',cargo:'Empleado',valorDia: 590000 ,diasTrabajado: 40 },      {id:1335098765 , nombre:'Andres', apellido:'chala',cargo:'Contador',valorDia:23000 ,diasTrabajado:50  },      {id:1240975434 , nombre:'Alejandro', apellido:'Andrade',cargo:'Instructor',valorDia: 45000 ,diasTrabajado:60  },      {id:1234567865 , nombre:'Cristofer', apellido:'Ceron',cargo:'Doctor',valorDia: 15000 ,diasTrabajado: 70 },      {id:1234567886, nombre:'Nataly', apellido:'Cepeda',cargo:'Bombero',valorDia:60000  ,diasTrabajado: 80 },      {id:1232456787 , nombre:'Sharon', apellido:'Sanchez',cargo:'Obrero',valorDia: 150000 ,diasTrabajado:90  },      {id:1233235648 , nombre:'Elba', apellido:'Ginon',cargo:'Soladador',valorDia: 320000 ,diasTrabajado: 10 },      {id:1542356759, nombre:'Karen', apellido:'Rodriguez',cargo:'Profesor',valorDia:1570000  ,diasTrabajado:20  },      {id:1234865320 , nombre:'carolina', apellido:'Moreno',cargo:'Empleado',valorDia:3670000  ,diasTrabajado: 45 }  ];  numeroRegistros = personas.length;  for (let iteracion = 0; iteracion < numeroRegistros; iteracion++) {      let trabajador = personas[iteracion]; *//permite rescorre toda el registro*      let salarioCalculado = funcionSalario(trabajador.valorDia, trabajador.diasTrabajado); *//calcula el salario de la persona*      let subTransCalculada=funcioSubtrans(salarioCalculado);      let retencionCalculada=funcionRetener(salarioCalculado);      let saludCalculada = funcionSalud(salarioCalculado);      let pensionCalculada=funcionPension(salarioCalculado);      let arlCalculada= funcionArl(salarioCalculado);      let deducibleCalculada=funcionDeducible(salarioCalculado);      let totalCalculada=funcionTotal(salarioCalculado);      nomina.push({          id: trabajador.id,          nombre: trabajador.nombre,          apellido: trabajador.apellido,          cargo: trabajador.cargo,          salario: salarioCalculado,          subTransporte: subTransCalculada,          retencion: retencionCalculada,          salud: saludCalculada ,          pension:  pensionCalculada,          arl:arlCalculada,          deducible: deducibleCalculada,          total:totalCalculada,      });      console.log(          "ID: " + trabajador.id + "\n" +          "Nombre: " + trabajador.nombre + "\n" +          "Apellido: " + trabajador.apellido + "\n" +          "Cargo: " + trabajador.cargo + "\n" +          "Salario: "  + salarioCalculado + "\n" +          "Subsidio de transporte: "  + subTransCalculada + "\n" +          "Retención: "  + retencionCalculada + "\n" +          "Salud: "  + saludCalculada+ "\n" +          "Pensión: "  + pensionCalculada + "\n" +          "ARL: "  + arlCalculada + "\n" +          "Deducible: "  + deducibleCalculada + "\n" +          "Total: "  + totalCalculada + "\n"      )  } | | |







|  |  |  |
| --- | --- | --- |
| **Nombre de los arreglos: tienda, listadoBusqueda** | | **Versión: 1.0** |
| **Descripción:**  Busqueda de objeto en array | | |
| **bingo,**  **matriz1**  **,matriz2,**  **matriz3** | **Tipo de variable:**  **Tienda(array alfanumerico)**  **numeroRegistro (int)**  **iteración(int)**  **listadoBusqueda (string)** | |
| Código:  let tienda =[];          let numeroRegistro;          let iteracion;          let nuevoProducto;          let listaBusqueda=[];          tienda=[          {producto:'Arroz',          tipoProducto:'granos',          tipoUnidad:'granos',          cantidad:'500',          precio:'2450',},          {producto:'papa',          tipoProducto:'fruver',          tipoUnidad:'granos',          cantidad:'500',          precio:'1000',},            {producto:'mora',          tipoProducto:'fruver',          tipoUnidad:'gramos',          cantidad:'500',          precio:'1500',},          {producto:'trucha',          tipoProducto:'carnes',          tipoUnidad:'gramos',          cantidad:'1000',          precio:'9000',},          {producto:'pollo entero',          tipoProducto:'carnes',          tipoUnidad:'gramos',          cantidad:'500',          precio:'4500',}          ]          nuevoProducto={producto:'Res',          tipoProducto:'carnes',          tipoUnidad:'gramos',          cantidad:'1000',          precio:'7500',};            tienda.push(nuevoProducto);          numeroRegistro=tienda.length;          for(iteracion=0;iteracion<numeroRegistro;iteracion++){              if(tienda[iteracion].tipoProducto=='fruver'){                  listaBusqueda.push(tienda[iteracion]);              }else{                  console.log('');              }          }          console.log(listaBusqueda); | | |

