

# TALLER INTERMEDIO DE MANEJO ÁUTOMÁTICO DE DATOS CON GOOGLE APPS SCRIPT.

Marina Corchado Sánchez

Técnico contratada para el proyecto

Tech4EfficiencyEDIH





## **OBJETIVOS:**

- Digitalización y eficiencia energética.

## **SECTOR:**

- Agrícola, forestal e industria alimentaria.

## **SERVICIOS:**

- Ensayos demostrativos de tecnologías de bajo coste y bajo consumo energético
- Formación- cursos y talleres prácticos para aprender a manejar estas tecnologías



# Tech4EfficiencyEDIH



## **Investigadores:**

**Adrián Javier Montero Calvo**  
Sistemas forestales mediterráneos

**Carlos Campillo Torres**  
Hortofruticultura. Tecnología para la sostenibilidad

**Manuel Joaquin Serradilla Sánchez**  
tecnologías postcosecha

## **Técnico:**

**Marina Corchado Sánchez**  
Sistemas forestales mediterráneos





## TALLER DE CICYTEX PARA EL PROYECTO TECH4E

1. TALLER BÁSICO DE REDES DE SENSORES INÁLAMBRICOS
2. TALLER DE INICIACIÓN A APPS SCRIPT PARA GOOGLE SHEETS
3. TALLER INTERMEDIO DE MANEJO AUTOMÁTICO DE DATOS CON GOOGLE APPS SCRIPT.
4. CREACIÓN DE DASHBOARDS DINÁMICOS CON LOOCKER STUDIO



European  
Digital Innovation  
Hubs Network



European  
Commission

JUNTA DE EXTREMADURA  
Consejería de Educación, Ciencia y Formación Profesional



The TECH4E project is financed by European Union under the Agreement - 101083667 of the Project "TECH4E. Tech4efficiencyEDIH" regarding the Call: DIGITAL-2021-EDIH-01 supported by the European Commission through the Digital Europe Program

# TALLER DE INICIACIÓN A APPS SCRIPT PARA GOOGLE SHEETS

## Contenido básico

- Acceder al entorno de desarrollo desde Google sheets
  - Funciones
  - Logger.log y console.log
  - Tipos de variables
  - Variables locales y globales
  - Parámetros de función
- Variables complejas
    - Array
    - Objetos
    - Array de objetos
  - Añadir elementos a variables complejas: Método push
  - Bucles y condicionales

# ¿Qué es Google Apps Script?

Google Apps Script es una plataforma de desarrollo que permite crear scripts para automatizar tareas. **ESTÁ BASADO EN JAVASCRIPT**

En este taller vamos a aprender los conceptos básicos de apps script

Esto nos permitirá empezar a comprender el lenguaje y la estructura del código



European  
Digital Innovation  
Hubs Network



European  
Commission

JUNTA DE EXTREMADURA  
Consejería de Educación, Ciencia y Formación Profesional



The TECH4E project is financed by European Union under the Agreement - 101083667 of the Project "TECH4E. Tech4efficiencyEDIH" regarding the Call: DIGITAL-2021-EDIH-01 supported by the European Commission through the Digital Europe Program

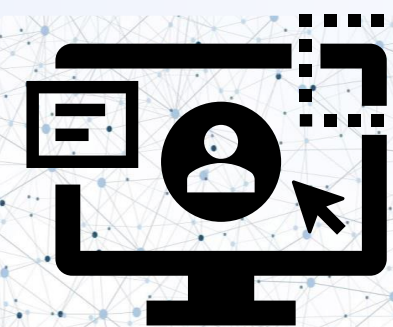


# ¿QUÉ NECESITAMOS?

CUENTA DE GOOGLE

HOJAS DE CÁLCULO DE GOOGLE

APPS SCRIPT



Google Sheets



	fecha-hora	parcela	nombre	CE...	Humedad...	Temperatura °C
1.	1 dic 2023, 5:55:47	P770	torno4	396	25,59	13,68
2.	1 dic 2023, 5:35:48	P770	torno4	396	25,68	13,68
3.	1 dic 2023, 6:15:47	P770	torno4	395	25,52	13,67
4.	1 dic 2023, 5:15:48	P770	torno4	395	25,74	13,69
5.	1 dic 2023, 4:55:48	P770	torno4	394	25,81	13,69
6.	1 dic 2023, 6:35:50	P770	torno4	394	25,48	13,66
7.	1 dic 2023, 6:55:46	P770	torno4	394	25,41	13,65
8.	1 dic 2023, 7:15:46	P770	torno4	393	25,36	13,63



European  
Digital Innovation  
Hubs Network



European  
Commission

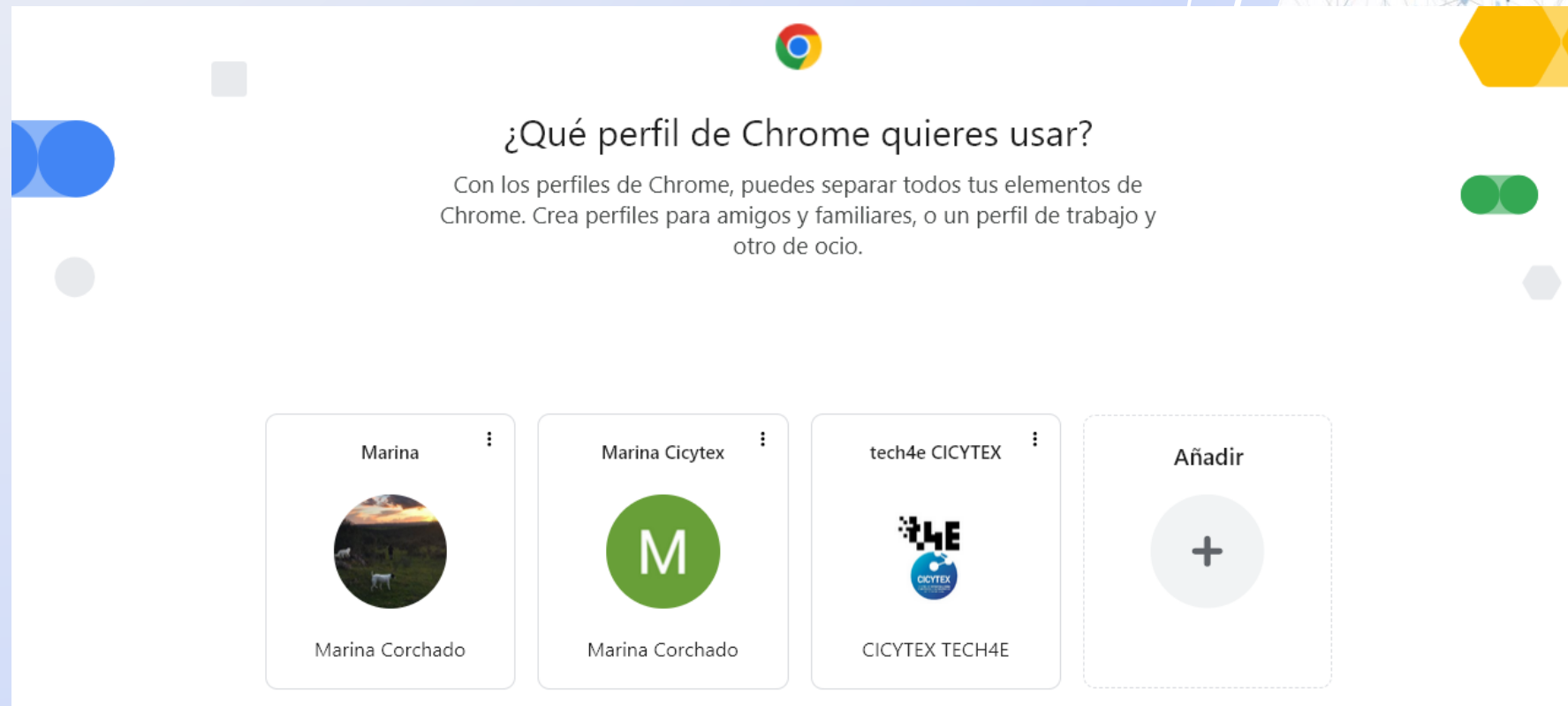
JUNTA DE EXTREMADURA  
Consejería de Educación, Ciencia y Formación Profesional



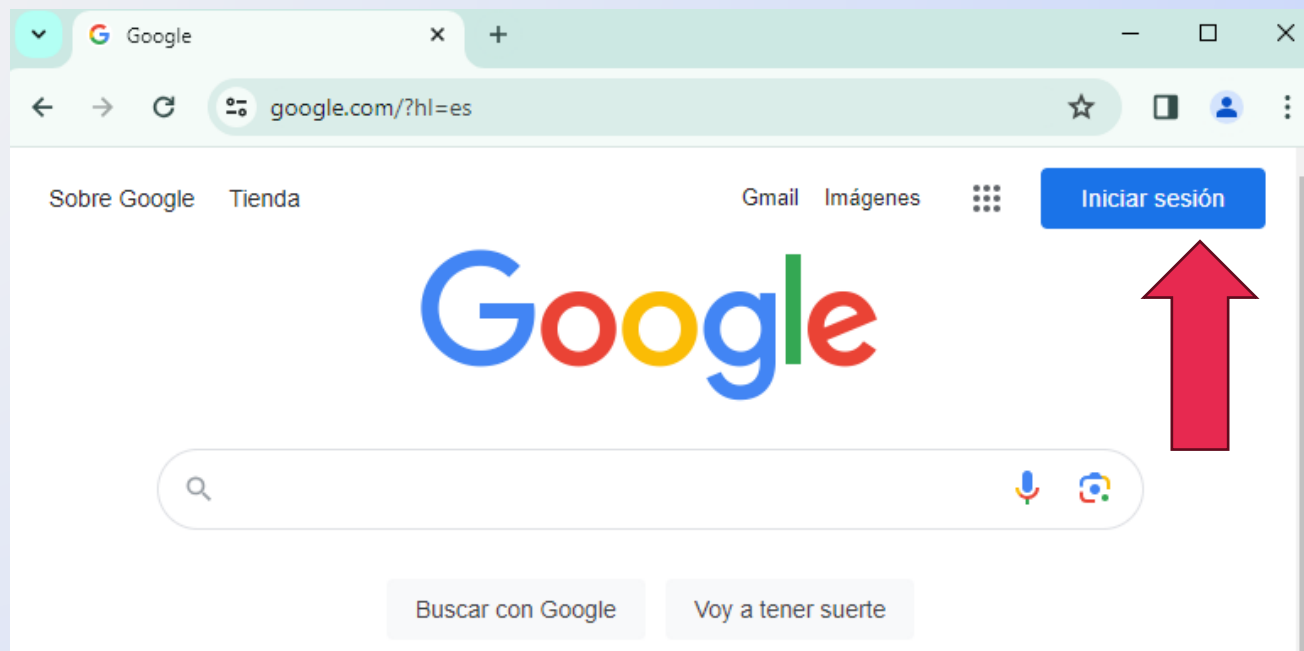
The TECH4E project is financed by European Union under the Agreement - 101083667 of the Project "TECH4E. Tech4efficiencyEDIH" regarding the Call: DIGITAL-2021-EDIH-01 supported by the European Commission through the Digital Europe Program

# Acceder a Google Sheets

Lo primero que tienes que saber sobre el funcionamiento de Google Sheets es que no necesitas de ningún tipo de programa de instalación. Esta hoja de cálculo es gratuita y para empezar a usarlo sólo necesitas un **correo electrónico de Gmail o cuenta de Google**.







Escribe tu correo y pulsa en siguiente para iniciar sesión

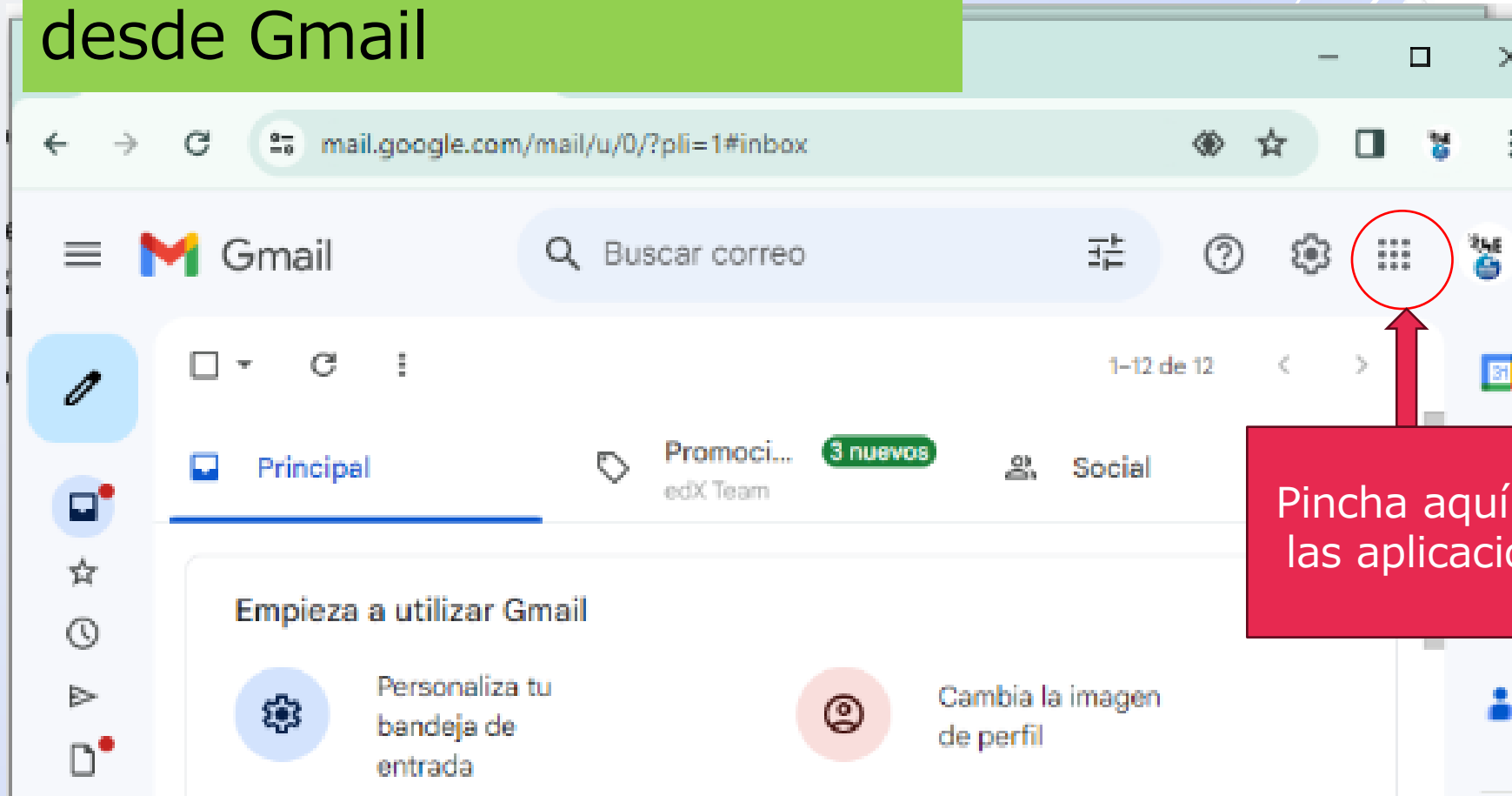
Pincha aquí para crear una cuenta





Pincha aquí para acceder a las aplicaciones de google

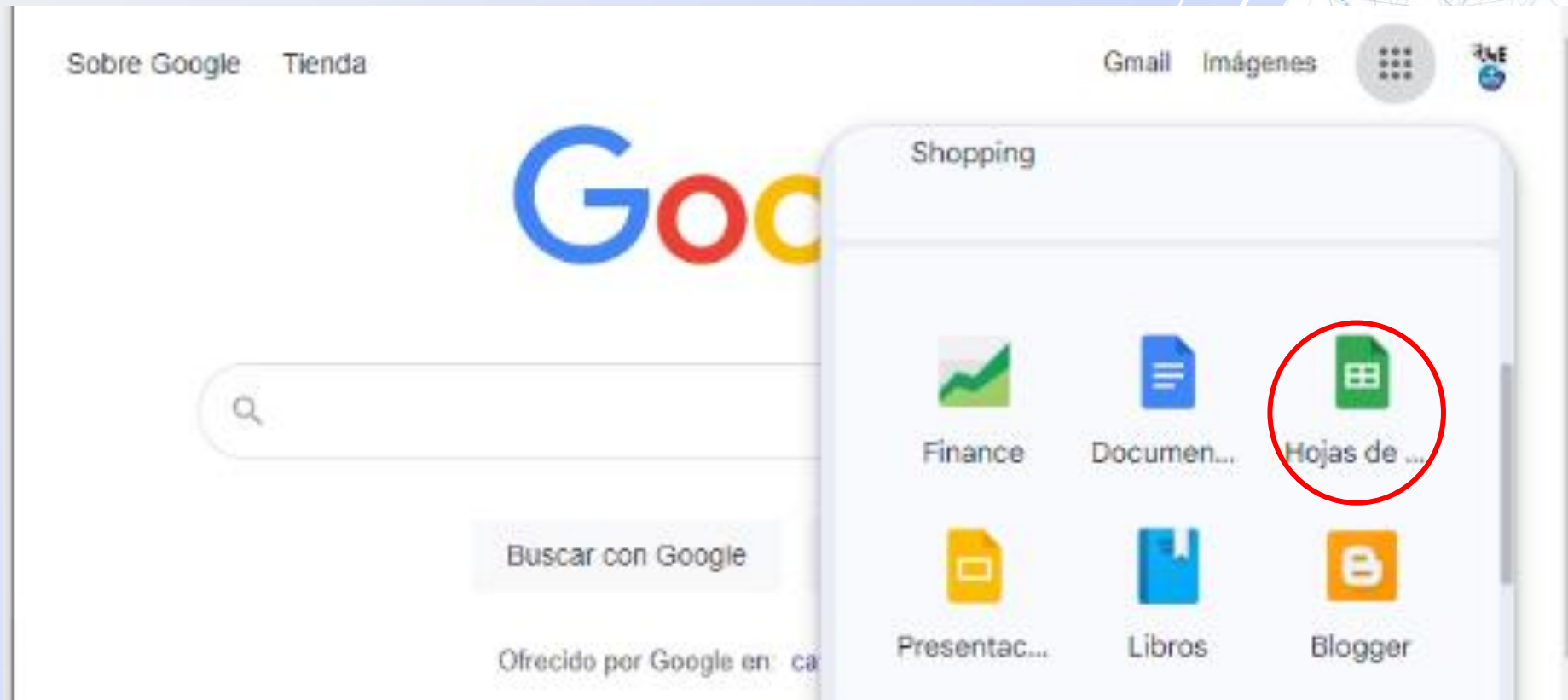
Si no usas Chrome accede desde Gmail



Pincha aquí para acceder a las aplicaciones de google



Bajamos con el ratón hasta que veamos la app de Google Sheets



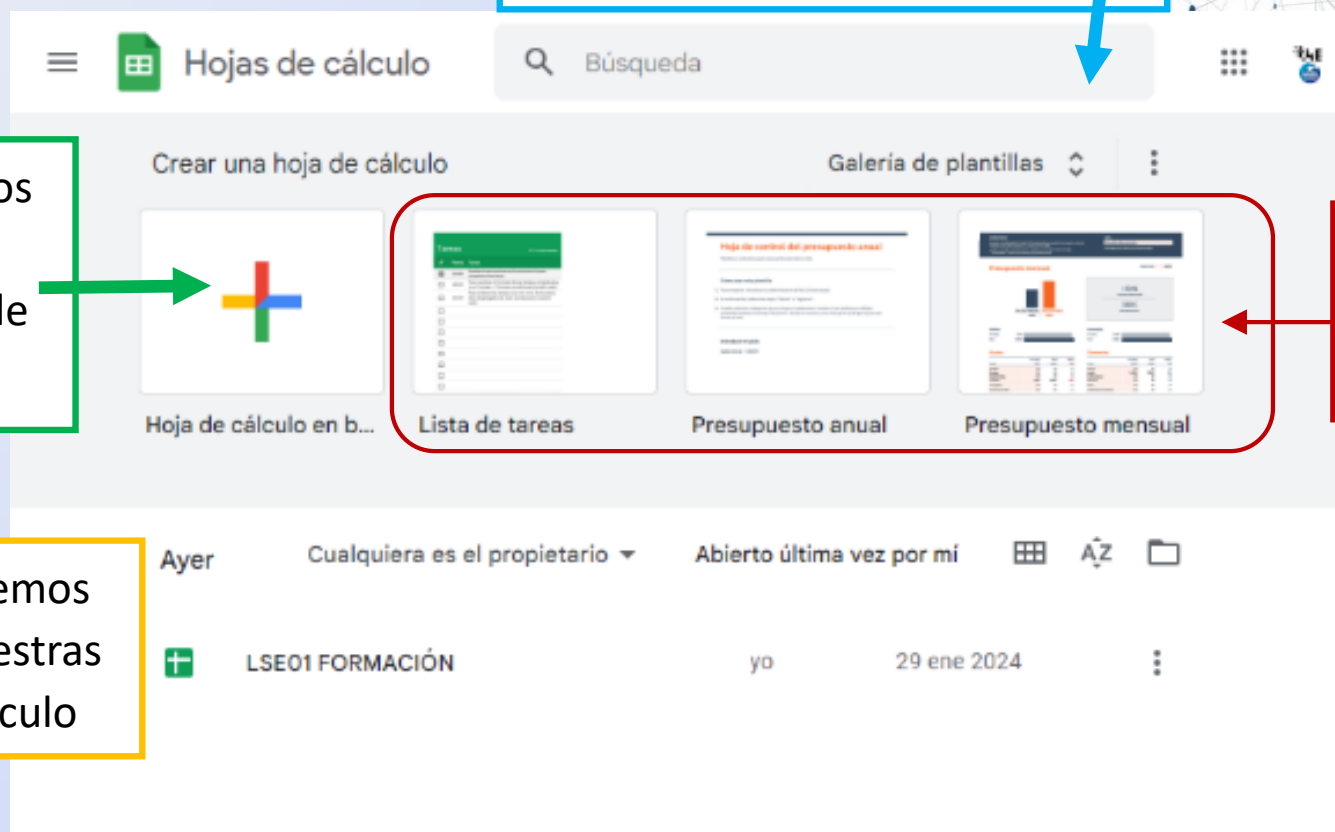
ABRE UNA NUEVA HOJA

Aquí podemos buscar nuestras  
hojas por su nombre

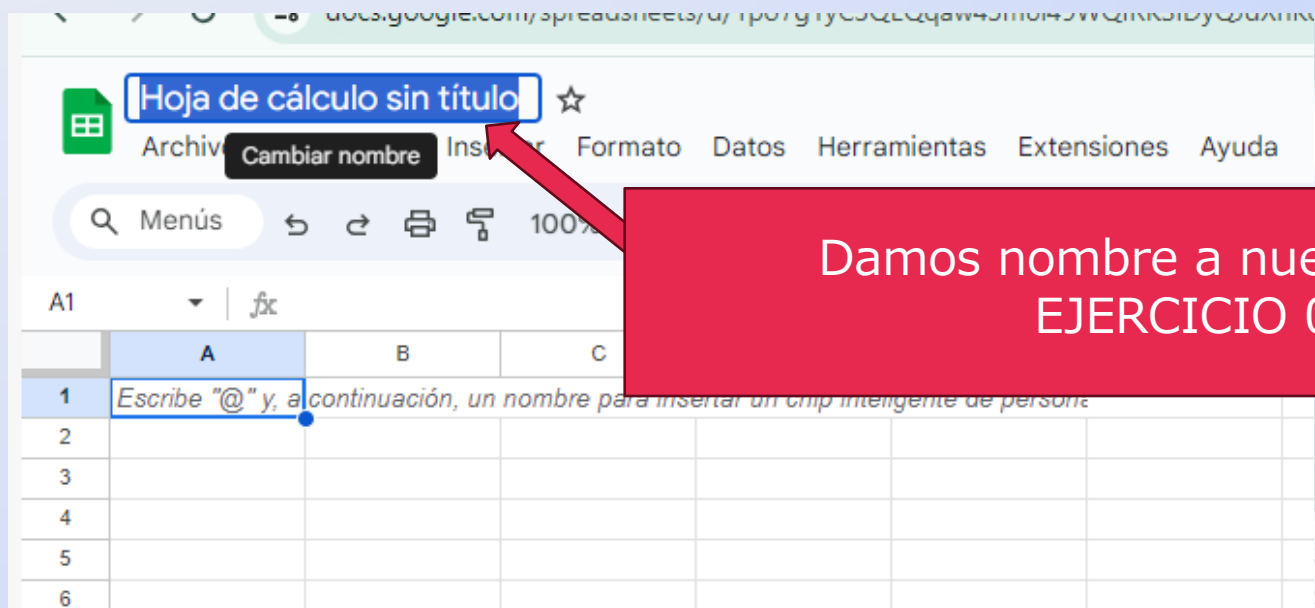
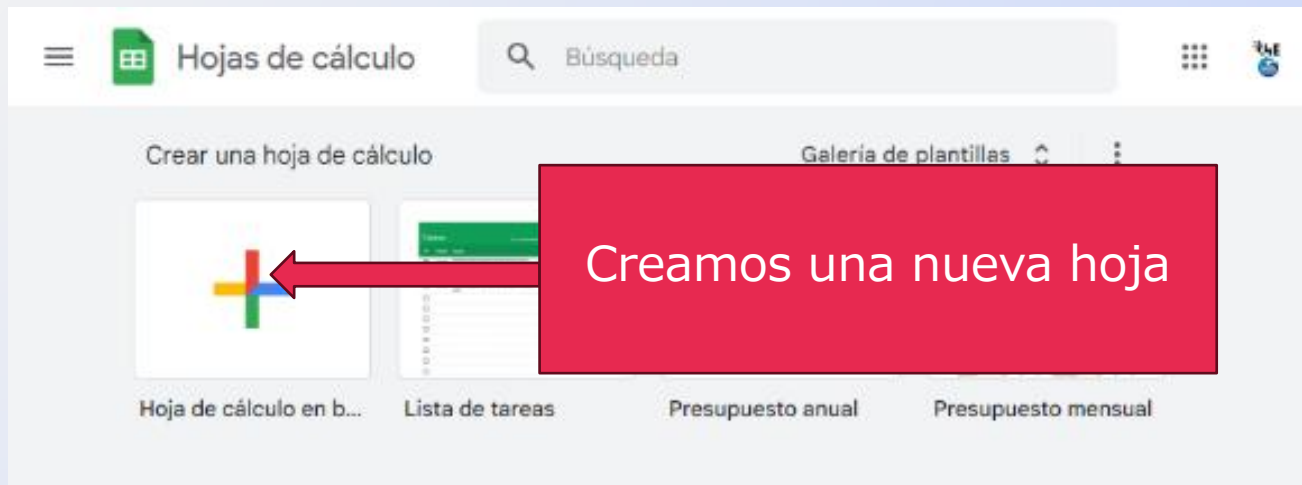
Aquí podemos  
crear una  
nueva hoja de  
cálculo

Aquí podemos  
seleccionar  
plantillas

Abajo tendremos  
la lista de nuestras  
hojas de cálculo







The image shows a Google Sheets spreadsheet titled 'LSE01 FORMACIÓN' and the Google Apps Script editor interface. A red arrow points from a red-bordered box labeled 'Pulsa en extensiones' to the 'Extensiones' menu in the Google Sheets toolbar. A yellow arrow points from a yellow-bordered box labeled 'Abre Apps Script' to the 'Apps Script' option in the 'Extensiones' dropdown menu. Below, a yellow-bordered box labeled 'Cambiar nombre pinchando aquí' points to the 'Proyecto sin título' text in the Apps Script editor's top bar.

LSE01 FORMACIÓN

Archivo Editar Ver Insertar Formato Datos Herramientas Extensiones Ayuda

Complementos  
Macros  
Apps Script  
AppSheet

Pulsa en extensiones

Abre Apps Script

Apps Script Proyecto sin título

Cambiar nombre pinchando aquí

## PRIMEROS EJERCICIOS

Cambiar nombre del proyecto

Título del proyecto\*

Proyecto sin título

Cancelar

Cambiar nombre



Por defecto viene la estructura de una función en Apps Script.  
Esas líneas de código vamos a borrarlas.



Archivos       Ejecutar  Depuración No hay funciones

 Código.gs

 Bibliotecas 

```
1 function myFunction() {  
2  
3 }  
4
```

DEJAR ABIERTO EL EDITOR PARA PROBAR LAS FUNCIONALIDADES BÁSICAS A MEDIDA QUE AVANZAMOS

# Funciones

## DEFINICIÓN:

Una función es un conjunto de instrucciones que realizan una tarea específica.

## ESTRUCTURA:

```
function miFuncion() {  
    //tareas que realizará la función  
}
```

NOTA: Cuando colocamos // sirve para dar una descripción en el código. Todo lo que escribamos con // delante no será ejecutado por nuestro script.

## Nombre de la función:

La estructura protocolaria que se suele utilizar es empezar en minúsculas, separar las palabras con una letra mayúscula y evitar caracteres especiales como acentos:

- nuevaFuncion,
- alertaHumedad,
- etc



European  
Digital Innovation  
Hubs Network



European  
Commission

JUNTA DE EXTREMADURA  
Consejería de Educación, Ciencia y Formación Profesional



The TECH4E project is financed by European Union under the Agreement – 101083667 of the Project "TECH4E .Tech4efficiencyEDIH" regarding the Call: DIGITAL-2021-EDIH-01 supported by the European Commission through the Digital Europe Program



# Funciones internas para registrar en la consola de ejecución

## Logger.log:

Especifica de Apps Script

## console.log:

Válida en Java Script y Apss Script

## EJEMPLO:

```
Logger.log("Hola mundo");  
console.log ("Hola mundo");  
Logger.log(8);  
Logger.log("Arriba aparece un " + 8);
```

### Concatenar:

Si queremos mostrar más de un elemento dentro de un log, entre cada elemento pondremos el signo

+



European  
Digital Innovation  
Hubs Network



European  
Commission

JUNTA DE EXTREMADURA  
Consejería de Educación, Ciencia y Formación Profesional



The TECH4E project is financed by European Union under the Agreement – 101083667 of the Project "TECH4E .Tech4efficiencyEDIH" regarding the Call: DIGITAL-2021-EDIH-01 supported by the European Commission through the Digital Europe Program

# Llamar a una función

## DEFINICIÓN:

Una vez definida una función podemos activarla dentro de otras funciones

## EJEMPLO:

```
function saludo() {  
    Logger.log("Hola mundo");  
}  
  
function cuenta() {  
    Logger.log("6 + 3 = " + (6 + 3));  
}  
  
function despedida() {  
    Logger.log("Hasta la próxima");  
}
```

```
function ejecutarFuncion() {  
    saludo();  
    cuenta();  
    despedida();  
}
```

En el ejemplo de la izquierda hay tres funciones diferentes.  
En la derecha está una función que llama a las tres anteriores en el orden indicado.

## Ejercicio 0

Vamos a ver cómo funciona **Logger.log** en apps script



European  
Digital Innovation  
Hubs Network



European  
Commission

JUNTA DE EXTREMADURA  
Consejería de Educación, Ciencia y Formación Profesional



The TECH4E project is financed by European Union under the Agreement - 101083667 of the Project "TECH4E .Tech4efficiencyEDIH" regarding the Call: DIGITAL-2021-EDIH-01 supported by the European Commission through the Digital Europe Program



# Variables

## DEFINICIÓN:

En Google Apps Script, una variable es un espacio de almacenamiento que tiene un nombre y se utiliza para almacenar y manipular datos en un script.

**NOTA:** Cada vez que escribamos una línea de código debe acabar en punto y coma ( ; ) antes de pasar a la siguiente.

## TIPOS DE VARIABLES:

**SIMPLES:** son variables que contienen un único dato.

**COMPLEJAS:** son variables que contienen varios datos

- Arrays
- Objetos
- Array de objetos

## Nombre de la variable:

La estructura protocolaria que se suele utilizar es empezar en minúsculas, separar las palabras con una letra mayúscula y evitar caracteres especiales como acentos:

- nombreParcela,
- humedad,
- conductividadElectrica

La variable se designará con mayúsculas cuando son constantes

- PI

```
var PI = 3.1416;
```



European  
Digital Innovation  
Hubs Network



European  
Commission

JUNTA DE EXTREMADURA  
Consejería de Educación, Ciencia y Formación Profesional



The TECH4E project is financed by European Union under the Agreement - 101083667 of the Project "TECH4E .Tech4efficiencyEDIH" regarding the Call: DIGITAL-2021-EDIH-01 supported by the European Commission through the Digital Europe Program

# Variables simples.

## ESTRUCTURA:

Declaramos la variable: **var**  
Damos **nombre** a la variable.  
Ponemos un **=**  
Escribimos el valor  
Cerramos la línea de código con **;**

```
var cultivoParcela = "Pimiento";  
var numeroParcelas = 6;
```

## NOTA:

Los decimales se separan con punto, no con coma

**8.14**

Las cadenas de texto se ponen entre comillas

**"Hola, mundo"**

Los booleanos se escriben en minúsculas

**true**

## TIPOS DE DATOS:

Apps script es muy flexible en cuanto al tipo de datos que contiene la variable

### Número entero:

```
var numero = 42;
```

### Número decimal:

```
var decimal = 8.14;
```

### Cadena de texto:

```
var texto = "Hola, mundo";
```

### booleano:

Los booleanos se pueden utilizar directamente sin necesidad de ponerlos en una variable

```
var esVerdadero = true;
```

```
var esFalso = false;
```



European  
Digital Innovation  
Hubs Network



European  
Commission

JUNTA DE EXTREMADURA  
Consejería de Educación, Ciencia y Formación Profesional



The TECH4E project is financed by European Union under the Agreement – 101083667 of the Project "TECH4E .Tech4efficiencyEDIH" regarding the Call: DIGITAL-2021-EDIH-01 supported by the European Commission through the Digital Europe Program



# Variables globales o locales.

## variables globales

se definen fuera de las funciones y por lo tanto son accesibles desde cualquier parte del script

## variables locales

se definen dentro de una función y por lo tanto sólo esa función tiene acceso y no el resto del script

```
1  //Variable numéricas globales
2  // estas variables están fuera de cualquier función
3  //por lo que pueden se utilizadas por cualquier funcio del script
4
5  var a = 1;
6  var b = 3;
7
8  function matematicas() {
9
10     //Variable numéricas locales
11     // estas variables están dentro de la función matematicas
12     var c = 12;
13     var d = 6;
```

# ejercicio1.

Vamos a abrir el editor de apps script y a copiar la función matemáticas.

```
function matematicas() {  
  
  var suma = a + b;  
  var resta = a - b;  
  
  //Logger.log es una de las funciones internas que sirve para mostrar el resultado de una variable en la pantalla de ejecución. es una con la que  
  podemos consultar información sobre los resultados de nuestro script  
  
  Logger.log('El resultado de la suma es: ' + suma);  
  Logger.log("el resultado de la resta es: " + resta);  
  
  //Ejercicio1: ESCRIBE LAS LINEAS DE CODIGO NECESARIAS PARA QUE SE DE EL RESULTADO DE LA DIVISIÓN y la multiplicación  
}
```

Si guardamos y ejecutamos matemáticas nos muestra en pantalla el resultado de la suma y de la resta

Escribe ahora las líneas de código necesarias para que la función dé el resultado de la multiplicación y la división



# Parámetros de función.

Cuando defines una función puedes especificar uno o más parámetros entre paréntesis dentro de la declaración de la función.

```
function calculos(numero1,numero2){  
    var suma = numero1 + numero2;  
    var resta = numero1 - numero2;
```

Estos parámetros actúan como variables locales dentro del cuerpo de la función y toman los valores que se les pasan cuando la función es llamada.

```
    calculos(6,3);
```

Así en el ejemplo numero1 tiene el valor 6 y numero2 el valor 3

## Ejercicio 2 – parámetros de función

Vamos a poner esto en práctica con el ejercicio 2



European  
Digital Innovation  
Hubs Network



European  
Commission

JUNTA DE EXTREMADURA  
Consejería de Educación, Ciencia y Formación Profesional



The TECH4E project is financed by European Union under the Agreement – 101083667 of the Project "TECH4E .Tech4efficiencyEDIH" regarding the Call: DIGITAL-2021-EDIH-01 supported by the European Commission through the Digital Europe Program

# Variables complejas. Arrays

## DEFINICIÓN:

- Un array es una estructura de datos que permite almacenar múltiples valores bajo un solo nombre.
- Estos valores se almacenan en posiciones consecutivas y se accede a ellos mediante un índice.

## ESTRUCTURA:

Declaramos la variable: **var**

Damos **nombre** a la variable.

Ponemos un **=**

Abrimos corchetes **[**

Escribimos los valores separados por comas

Cerramos los corchetes **]**

Cerramos la línea de código con **;**

```
var numeros = [1, 2, 3, 4, 5];
```

```
var frutas = ["Manzana", "Banana", "Cereza"];
```



European  
Digital Innovation  
Hubs Network



European  
Commission

JUNTA DE EXTREMADURA  
Consejería de Educación, Ciencia y Formación Profesional



The TECH4E project is financed by European Union under the Agreement - 101083667 of the Project "TECH4E .Tech4efficiencyEDIH" regarding the Call: DIGITAL-2021-EDIH-01 supported by the European Commission through the Digital Europe Program



# Variables complejas. Objetos

## DEFINICIÓN:

- Un objeto es una colección de pares clave-valor, donde cada valor está asociado a una clave única.
- Los objetos permiten organizar y estructurar datos de manera más compleja que los arrays.

## ESTRUCTURA:

- El objeto se engloba dentro de {}

```
var persona = {nombre: "Juan", edad: 30, ciudad: "Badajoz"};  
var parcela = {ciudad: "Lobón", polígono: 30, cultivo: "Maíz"};  
var sensor = {marca: "Dragino", modelo: "LSE01"};
```

# Variables complejas. Array de objetos

## DEFINICIÓN:

- Es un array, por lo que se define con [ ]
- En lugar de contener datos simples contiene objetos con pares clave-valor englobadas con { }
- En el ejemplo se muestra un array que contiene 3 objetos

## ESTRUCTURA:

```
var personas = [  
    {nombre: "Juan", edad: 30, ciudad: "Badajoz"};  
    {nombre: "María", edad: 31, ciudad: "Cáceres"};  
    {nombre: "Alicia", edad: 36, ciudad: "Mérida"};  
]
```



European  
Digital Innovation  
Hubs Network



European  
Commission

JUNTA DE EXTREMADURA  
Consejería de Educación, Ciencia y Formación Profesional



The TECH4E project is financed by European Union under the Agreement - 101083667 of the Project "TECH4E .Tech4efficiencyEDIH" regarding the Call: DIGITAL-2021-EDIH-01 supported by the European Commission through the Digital Europe Program



# Acceder a la información de las variables complejas

## Arrays

- Para acceder a un elemento concreto de un array lo haremos a través de su índice.
- Para ello ponemos el nombre de la variable
- Abrimos corchetes [ ]
- Y dentro de los corchetes ponemos su índice
- LOS INDICES EMPIEZAN EN CERO

```
var frutas = ["Manzana", "Banana", "Cereza"];
```

`frutas[0]` → devuelve "Manzana"

`frutas[1]` → devuelve "Banana"

`frutas[2]` → devuelve "Cereza"



European  
Digital Innovation  
Hubs Network



European  
Commission

JUNTA DE EXTREMADURA  
Consejería de Educación, Ciencia y Formación Profesional



The TECH4E project is financed by European Union under the Agreement - 101083667 of the Project "TECH4E .Tech4efficiencyEDIH" regarding the Call: DIGITAL-2021-EDIH-01 supported by the European Commission through the Digital Europe Program

Logo for the Digital Europe Program, featuring a stylized blue and white design.

# Consultar índices de un array:

```
var cultivos = ["tomate", "patata", "Olivo", "Ciruelo", "almendro"]
```

```
function consultarArray() {  
  for (var i = 0; i < cultivos.length; i++) {  
    console.log("Índice " + i + ": " + cultivos[i]);  
  }  
}
```

- Si tenemos dudas sobre el índice de los elementos de un array podemos utilizar la función consultar array
- Esta función utiliza un bucle for. Los bucles los explicaremos con más detalle más adelante

## Ejercicio 3 – Array e índices

Vamos a poner esto en práctica con el ejercicio 3



European  
Digital Innovation  
Hubs Network



European  
Commission

JUNTA DE EXTREMADURA  
Consejería de Educación, Ciencia y Formación Profesional



The TECH4E project is financed by European Union under the Agreement – 101083667 of the Project "TECH4E .Tech4efficiencyEDIH" regarding the Call: DIGITAL-2021-EDIH-01 supported by the European Commission through the Digital Europe Program

Logo of the project supported by the European Union under the Digital Europe Program

# Acceder a la información de las variables complejas

## Objetos

- Para acceder a un elemento concreto de un objeto lo haremos a través de su índice.
- Para ello ponemos el nombre de la variable seguida de un punto y el nombre de la clave del valor al que queremos acceder

```
var persona = {nombre: "Juan", edad: 30, ciudad: "Badajoz"};  
var parcela = {ciudad: "Lobón", polígono: 30, cultivo: "Maíz"};  
var sensor = {marca: "Dragino", modelo: "LSE01"};
```

`persona.nombre` → devuelve "Juan"

`parcela.cultivo` → devuelve "Maíz"

`sensor.modelo` → devuelve "LSE01"



# Consultar claves de un objeto:

```
var persona = {nombre: "Juan", edad: 30, ciudad: "Badajoz"};  
var parcela = {ciudad: "Lobón", polígono: 30, cultivo: "Maíz"};  
var sensor = {marca: "Dragino", modelo: "LSE01"};
```

```
function consultarObjeto(objeto) {  
  console.log(objeto);  
  for (var clave in objeto) {  
    console.log("Clave: " + clave + ", Valor: " + objeto[clave]);  
  }  
}
```

- Si tenemos dudas sobre las claves que hay en un objeto podemos utilizar la función consultar objeto
- Esta función utiliza un bucle for in. Los bucles los explicaremos con más detalle más adelante

## Ejercicio 4 – Objeto y claves

Vamos a poner esto en práctica con el ejercicio 4



European  
Digital Innovation  
Hubs Network



European  
Commission

JUNTA DE EXTREMADURA  
Consejería de Educación, Ciencia y Formación Profesional



The TECH4E project is financed by European Union under the Agreement – 101083667 of the Project "TECH4E .Tech4efficiencyEDIH" regarding the Call: DIGITAL-2021-EDIH-01 supported by the European Commission through the Digital Europe Program

Logo of the project supported by the European Union through the Digital Europe Program

# Acceder a la información de las variables complejas

## Array de Objetos

```
var personas = [  
    {nombre: "Juan", edad: 30, ciudad: "Badajoz"}; //indice0  
    {nombre: "María", edad: 31, ciudad: "Cáceres"}; //indice1  
    {nombre: "Alicia", edad: 36, ciudad: "Mérida"}; //indice2  
]
```

- Para acceder a un objeto concreto de un array de objetos lo haremos a través de su índice.
- Para acceder a el valor de una de las claves de un objeto determinado necesitamos el índice del objeto y el nombre de la clave

personas[0].nombre → "Juan"

personas[1].edad → 31

personas[2].ciudad → "Mérida"

# Consultar claves de un array de objetos:

```
var personas = [  
  {nombre: "Juan", edad: 30, ciudad: "Badajoz"},  
  {nombre: "María", edad: 31, ciudad: "Cáceres"},  
  {nombre: "Alicia", edad: 36, ciudad: "Mérida"}  
];
```

```
function consultarArrayDeObjetos(array) {  
  //bucle for para acceder al índice  
  for (var i = 0; i < array.length; i++) {  
    //creamos una variable objeto para los objetos según su índice i  
    var objeto = array[i];  
    //mostramos en la consola objeto de índice i  
    console.log("Objeto de índice " + (i) + ":");  
    //abrimos un bucle for in para recorrer las claves de cada objeto  
    for (var clave in objeto) {  
      console.log("  Clave: " + clave + ", Valor: " + objeto[clave]);  
    }  
  }  
}
```

## Ejercicio 5 – Array de objetos- índices y claves

Vamos a poner esto en práctica con el ejercicio 5



European  
Digital Innovation  
Hubs Network



European  
Commission

JUNTA DE EXTREMADURA  
Consejería de Educación, Ciencia y Formación Profesional



The TECH4E project is financed by European Union under the Agreement – 101083667 of the Project "TECH4E .Tech4efficiencyEDIH" regarding the Call: DIGITAL-2021-EDIH-01 supported by the European Commission through the Digital Europe Program

Logo of the project supported by the European Union under the Digital Europe Program



# Añadir elementos a variables complejas: (push)

- Con el método “push” podemos agregar elementos al array o nuevos objetos al array de objetos
- Para añadir pares clave-valor a un objeto no es necesario el método push.

```
// Crear un array vacío
var miArray = [];

// Agregar elementos al array
miArray.push("dato1");
miArray.push("dato2");
miArray.push("dato3");
```

```
// Crear un objeto vacío
var miObjeto = {};

// Agregar propiedades al objeto
miObjeto.propiedad1 = "valor1";
miObjeto.propiedad2 = "valor2";
miObjeto.propiedad3 = "valor3";
```

```
// Crear un array de objetos vacío
var arrayDeObjetos = [];
```

```
// Crear un objeto
var objeto1 = {
  propiedad1: "valor1",
  propiedad2: "valor2"
};
```

```
// Agregar el objeto al array
arrayDeObjetos.push(objeto1);
```

```
// Crear otro objeto
var objeto2 = {
  propiedad1: "otro valor",
  propiedad2: "otro valor2"
};
```

```
// Agregar el segundo objeto al array
arrayDeObjetos.push(objeto2);
```

## Ejercicio6- agregar elementos en variables complejas



European  
Digital Innovation  
Hubs Network



European  
Commission

JUNTA DE EXTREMADURA  
Consejería de Educación, Ciencia y Formación Profesional



The TECH4E project is financed by European Union under the Agreement – 101083667 of the Project “TECH4E .Tech4efficiencyEDIH” regarding the Call: DIGITAL-2021-EDIH-01 supported by the European Commission through the Digital Europe Program

Logo for the project, supported by the European Union under the Agreement – 101083667 of the Project “TECH4E .Tech4efficiencyEDIH” regarding the Call: DIGITAL-2021-EDIH-01 supported by the European Commission through the Digital Europe Program

# condicionales

## DEFINICIÓN:

- Las estructuras condicionales se utilizan para para tomar decisiones dentro del código.

## TIPOS:

- **If – else** → ejecuta un bloque de código si una condición es verdadera
- **Else - if** → Se encadena con el anterior para evaluar condiciones adicionales
- **Swich - case** → para evaluar una variable con múltiples valores posibles

# Condicionales if - else

## ESTRUCTURA :

```
function mayorEdad(){  
  
    var edad = 18;  
  
    if (edad >= 18) {  
        | Logger.log("Eres mayor de edad");  
    } else {  
        | Logger.log("Eres menor de edad");  
    }  
  
}
```



European  
Digital Innovation  
Hubs Network



European  
Commission

JUNTA DE EXTREMADURA  
Consejería de Educación, Ciencia y Formación Profesional



The TECH4E project is financed by European Union under the Agreement - 101083667 of the Project "TECH4E .Tech4efficiencyEDIH" regarding the Call: DIGITAL-2021-EDIH-01 supported by the European Commission through the Digital Europe Program



# Condicionales else - if

## ESTRUCTURA :

```
function calificaciones(){  
    var puntuacion = 75;  
  
    if (puntuacion >= 90) {  
        | Logger.log("A");  
    } else if (puntuacion >= 80) {  
        | Logger.log("B");  
    } else if (puntuacion >= 70) {  
        | Logger.log("C");  
    } else {  
        | Logger.log("D");  
    }  
  
}
```



European  
Digital Innovation  
Hubs Network



European  
Commission

JUNTA DE EXTREMADURA  
Consejería de Educación, Ciencia y Formación Profesional



The TECH4E project is financed by European Union under the Agreement - 101083667 of the Project "TECH4E .Tech4efficiencyEDIH" regarding the Call: DIGITAL-2021-EDIH-01 supported by the European Commission through the Digital Europe Program

# Condicionales switch case

## Ejercicio7- condicionales

### ESTRUCTURA :

```
function diaSemana(){  
    var dia = "Lunes";  
    //swich abre la estructura de control que evaluará una expresión o valor y ejecutará la parte del código asociada  
    //entre parentesis pondremos la expresión a evaluar  
    switch (dia) {  
        case "Lunes": //case especifica un posible valor de la función  
            Logger.log("Comienzo de semana");  
            break; //si esta parte del código se ha ejecutado, break fuerza la salida de la expresión switch  
        case "Martes":  
        case "Miércoles":  
        case "Jueves":  
            Logger.log("Mitad de semana");  
            break;  
        case "Viernes":  
            Logger.log("¡Viernes, fin de semana cerca!");  
            break;  
        case "Sábado":  
        case "Domingo":  
            Logger.log("¡fin de semana!");  
            break;  
        default: // en el caso de que la expresión no coincida con ninguno de los otros casos expuestos  
            Logger.log("No es un día de la semana válido");  
    }  
}
```

# bucles

Existen diferentes bucles, pero de momento utilizaremos sólo el bucle for

## El bucle for

- Es un bucle de control de flujo que se utiliza para iterar sobre una secuencia de valores.
- Se compone de tres partes: la inicialización, la condición de continuación y la expresión de incremento.
- El bucle for ejecuta el bloque de código contenido dentro de él repetidamente hasta que la condición de continuación se evalúa como falsa.

```
for (inicialización; condición de continuación; expresión de  
    incremento) {  
    // código a ejecutar en cada iteración  
}
```

## Ejercicio8- Bucles y condicionales



European  
Digital Innovation  
Hubs Network



European  
Commission

JUNTA DE EXTREMADURA  
Consejería de Educación, Ciencia y Formación Profesional



The TECH4E project is financed by European Union under the Agreement – 101083667 of the Project "TECH4E .Tech4efficiencyEDIH" regarding the Call: DIGITAL-2021-EDIH-01 supported by the European Commission through the Digital Europe Program