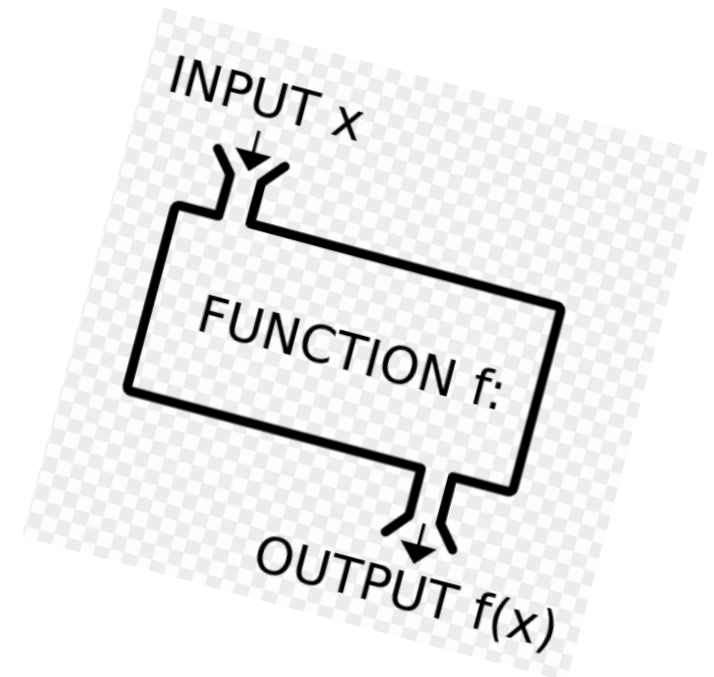
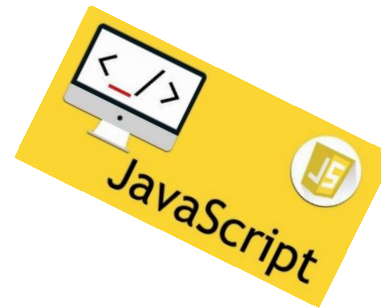


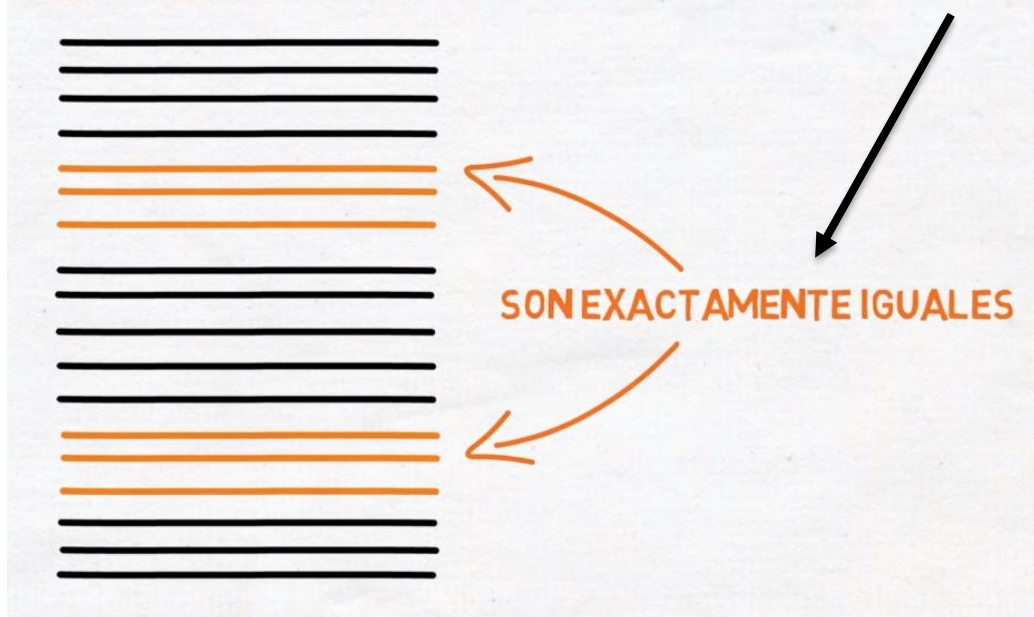
Bloque1: Programaci n b sica con JavaScript (JS)

Funciones

```
void DibujarPuntos( int numPuntos )  
{  
    int i;  
    for ( i = 1; i <= numPuntos; i++ )  
        printf(".");  
}  
void main()  
{  
    DibujarPuntos( 30 );  
}
```



EJEMPLO



es una pérdida de tiempo tener que volver a escribir una y otra vez el mismo código

Ejemplo completo con JavaScript

```
// función que calcula la suma de dos números
// y devuelve el resultado

function suma(num1,num2) {
    let calc;
    calc = num1 + num2;
    return calc;
}

let resultado, x = 10, y = 30;

resultado = suma(x,y);
console.log('La suma es' + resultado);
```

¿Qué son las funciones? (I)

```
function saludar ( ) {
```



```
}
```

```
function saludar (NOMBRE) {
```



```
}
```

Un valor que se le envía a la función

PARÁMETRO

FUNCIONES

Una función es un conjunto de líneas de código que realizan una tarea específica y pueden retornar algo.



```
saludar ( );
```

```
saludar ( "Salva" );
```

¿Qué son las funciones? (II)

Ejemplos básicos con JavaScript (análisis)...

```
// funcion saludar (no recibe parámetros o argumentos)

function saludar() {
  console.log("Hola a todos/as");
}

// Realizo una llamada a la función
saludar();
```

Ejemplo 1

A practicar !!!

Ejemplo 2

A practicar !!!

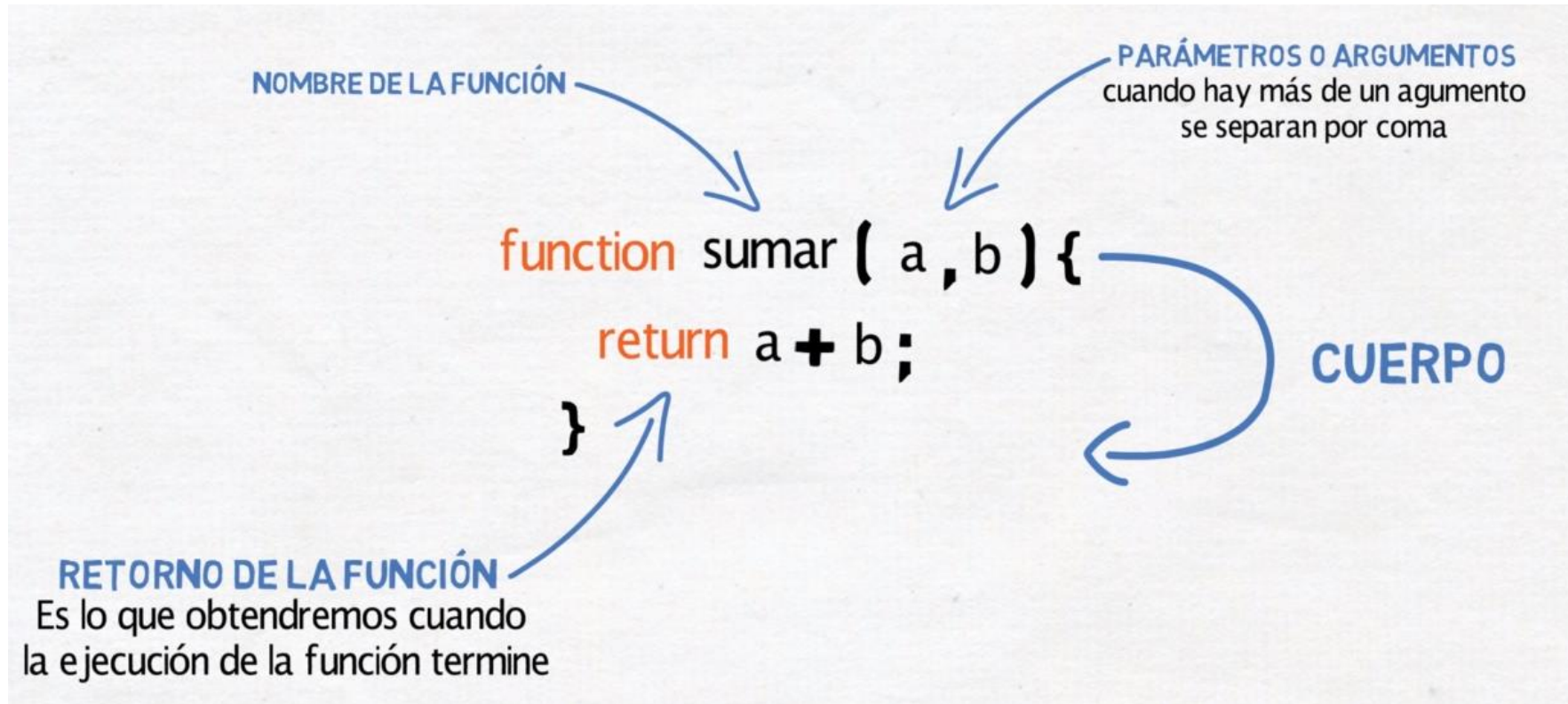
```
// funcion saludar (recibe un parámetro)

function saludar(nombre) {
  console.log("Hola " + nombre);
}

let nom = 'Salva'
// Realizo varias llamadas a la función con diferente parámetro
saludar(nom);
saludar('Pedro');
saludar('Ana');
```

¿Qué son las funciones? (III)

¿Qué es el retorno de una funci n?...  Es obligatorio?....



Llamada a la funci n que retorna 33 en la variable **total** (que es definida en la misma l nea)

```
let total = sumar ( 10,23 );
```

01. Funci n que calcule y devuelva/retorne la suma de dos n meros.

02. Funci n que calcule y visualice el  rea de un rect ngulo. Los par metros que recibe para calcular son la base y la altura.

03. Funci n que calcule y retorne el  rea de un rect ngulo. Los par metros que recibe son la base y la altura.

04. Funci n que calcule y visualice la tabla de multiplicar de un n mero (del 1 al 10). Debe recibir el n mero (la tabla a visualizar) como par metro.

05. Funci n que calcule y visualice la tabla de multiplicar de un n mero (del 1 hasta un l mite dado). Debe recibir el n mero y el l mite m ximo como par metros.

A practicar !!!

01. Función que calcule y devuelva/retorne la suma de dos números.

```
// 01. Función que calcule y devuelva/retorne la suma de dos números.
```

```
function suma(a,b) { // recibe dos parámetros  
    return a+b;  
}
```

```
// ejemplo de llamadas y uso de la función
```

```
let resultado;  
resultado = suma(10,20);  
console.log('La suma es '+resultado);
```

```
let x=22; y=12;  
resultado = suma(x,y);  
console.log('La suma es '+resultado);
```

```
let j=22;  
console.log('La suma es ' + suma(j,40));
```

```
public static double suma(double valor1, double valor2)  
{  
    double suma = 0;  
  
    suma = valor1 + valor2;  
  
    return suma;  
}
```

*Ejemplo función
suma con C#.*

Ejecución:

```
La suma es 30  
La suma es 34  
La suma es 62
```

02. Función que calcule y visualice el área de un rectángulo. Los parámetros que recibe para calcular son la base y la altura.

Ejecución:

```
Rectangulo de base 12 y altura 20  
El area del rectángulo es 240  
Rectangulo de base 34 y altura 5  
El area del rectángulo es 170
```

```
// 02. Función que calcule y visualice el área de un rectángulo.  
// Los parámetros que recibe para calcular son la base y la altura.  
  
function calculoArea(base,altura) { // recibe dos parámetros  
    let area = base * altura;  
    console.log('El area del rectángulo es '+area);  
}  
  
// ejemplo 1 de llamada y uso de la función  
let b=12, a=20;  
console.log('Rectangulo de base ' + b + ' y altura ' + a);  
calculoArea(b,a);  
  
// ejemplo2 de llamada y uso de la función  
console.log('Rectangulo de base 34 y altura 5');  
calculoArea(34,5);
```


03. Función que calcule y retorne el área de un rectángulo. Los parámetros que recibe son la base y la altura.

Ejecución:

```
Rectangulo de base 12 y altura 20  
El area del rectángulo es 240  
Rectangulo de base 34 y altura 5  
El area del rectángulo es 170
```

```
// 03. Función que calcule y retorne el área de un rectángulo.  
// Los parámetros que recibe para calcular son la base y la altura.  
  
function calculoArea(base,altura) { // recibe dos parámetros  
    let area = base * altura;  
    return area;  
}  
  
// ejemplo 1 de llamada y uso de la función  
let b=12, a=20, area;  
console.log('Rectangulo de base ' + b + ' y altura ' + a);  
area = calculoArea(b,a);  
console.log('El area del rectángulo es ' + area);  
  
// ejemplo2 de llamada y uso de la función  
console.log('Rectangulo de base 34 y altura 5');  
console.log('El area del rectángulo es ' + calculoArea(34,5));
```

04. Función que calcule y visualice la tabla de multiplicar de un número (del 1 al 10). Debe recibir el número (la tabla) como parámetro.

```
// 04. Función que calcule y visualice la tabla de multiplicar
// (del 1 al 10) de un número. Debe recibir el número como parámetro.

function calculoTabla (numero) { // recibe un parámetro

    for (let i=1;i<=10;i++) {
        console.log(numero + ' x ' + i + ' = ' + numero*i);
    }
}

// ejemplos de llamada y uso de la función

let tabla=8;
calculoTabla(tabla); // visualiza la tabla del 8

tabla = 5;
calculoTabla(tabla); // visualiza la tabla del 5

calculoTabla(4); // visualiza la tabla del 4
```

05. Función que calcule y visualice la tabla de multiplicar de un número (del 1 hasta un límite dado).

Debe recibir el número y el límite máximo como parámetros.

```
// 05. Función que calcule y visualice la tabla de multiplicar
// (del 1 hasta un límite) de un número. Debe recibir el número
// y el límite máximo como parámetros.

function calculoTabla (numero, limite) { // recibe dos parámetros

    for (let i=1;i<=limite;i++) {
        console.log(numero + ' x ' + i + ' = ' + numero*i);
    }
}

// ejemplos de llamada y uso de la función

// visualiza la tabla del 8 hasta el límite de 15 (8x15)
let tabla=8, lim = 15;
calculoTabla(tabla, lim);
// visualiza la tabla del 5 hasta el límite de 20 (5x20)
tabla = 5, lim = 20;
calculoTabla(tabla,lim);
// visualiza la tabla del 4 hasta el límite de 31 (4x31)
calculoTabla(4,31);
```

EJERCICIO 12 de la AP1.

Programa que calcula el mayor de tres números proporcionados por teclado.

Previamente a ello debe comprobar que los tres números suministrados son distintos. En caso de que no lo sea debe avisar que los números suministrados no son válidos.

```
let readlineSync = require('readline-sync');
let num1 = readlineSync.question('Introduce el primer numero: ');
let num2 = readlineSync.question('Introduce el segundo numero: ');
let num3 = readlineSync.question('Introduce el tercer numero: ');

if (num1==num2 || num1==num3 || num2==num3)
  console.log('Los numeros suministrados no son validos');
else if (num1 >= num2 && num1 >= num3)
  // el mayor es num1
  console.log('El numero mayor es',num1);
else if (num2 >= num3)
  // el mayor es num2
  console.log('El numero mayor es',num2);
else
  // el mayor es num3
  console.log('El numero mayor es',num3);
```

Realización utilizando funciones

...Análisis previo...

A practicar !!!

EJERCICIO 12 de la AP1 (con funciones)

```
function sonValidos(n1,n2,n3) {  
  
    if (n1==n2 || n1==n3 || n2==n3)  
        // Los numeros suministrados NO son validos.  
        return false  
    else  
        // Los numeros suministrados son validos  
        return true;  
  
}  
  
function calculaMayor(n1,n2,n3) {  
  
    if (n1 > n2 && n1 > n3)  
        // el mayor es n1 y es el que retorno  
        return n1;  
    else if (n2 > n3)  
        // el mayor es n2 y lo retorno  
        return n2;  
    else  
        // el mayor es n3 y lo retorno  
        return n3;  
  
}
```

sonValidos(n1,n2,n3)

calculaMayor(n1,n2,n3)

```
let readlineSync = require('readline-sync');  
let mayor; // variable para guardar el mayor  
let num1 = readlineSync.question('Introduce el primer numero: ');  
let num2 = readlineSync.question('Introduce el segundo numero: ');  
let num3 = readlineSync.question('Introduce el tercer numero: ');  
  
if (sonValidos(num1,num2,num3)) {  
    mayor = calculaMayor(num1,num2,num3);  
    console.log('El mayor es ' + mayor);  
} else {  
    console.log('Los números no son válidos');  
}
```

EJERCICIO 13 de la AP1

Programa que lea tres números enteros H, M, S que contienen hora, minutos y segundos respectivamente, y comprueba si la hora que indican es una hora válida. Será válida en caso que la hora esté entre 0..24 y los minutos y segundos entre 0..59.

```
let horaValida = true;
let mensajeError = '';

let readlineSync = require('readline-sync');
let horas = readlineSync.question('Introduce las horas: ');
let minutos = readlineSync.question('Introduce los minutos: ');
let segundos = readlineSync.question('Introduce los segundos: ');

if (horas<0 || horas >=24) {
  mensajeError = 'Las horas no son validas. ';
  horaValida = false;
}
if (minutos<0 || minutos>59) {
  mensajeError += 'Los minutos no son validos. ';
  horaValida = false;
}
if (segundos<0 || segundos>59) {
  mensajeError += 'Los segundos no son validos. ';
  horaValida = false;
}
if (horaValida == false)
  console.log(mensajeError);
else
  console.log('La hora completa introducida es valida');
```


A practicar !!!

Realización utilizando funciones

EJERCICIO 13 de la AP1 (con funciones).

```
function validarHora(h,m,s) {  
  if (h<0 || h >=24) {  
    return false;  
  }  
  if (m<0 || m>59) {  
    return false;  
  }  
  if (s<0 || s>59) {  
    return false;  
  }  
  return true;  
}
```

`validarHora(horas,minutos,segundos)`



```
let readlineSync = require('readline-sync');  
let horas = readlineSync.question('Introduce las horas: ');  
let minutos = readlineSync.question('Introduce los minutos: ');  
let segundos = readlineSync.question('Introduce los segundos: ');  
  
let esValida; //Almacena el retorno de la función  
  
esValida = validarHora(horas,minutos,segundos);  
  
if (esValida) {  
  console.log('La hora tiene un formato correcto');  
} else {  
  console.log('La hora NO tiene un formato correcto');  
}
```

EJERCICIO 14 de la AP1.

Programa que, tras pedir la medida de los tres lados de un triángulo identifique de qué tipo es (escaleno, equilátero o isósceles).

A practicar !!!

Realización utilizando funciones

...Análisis previo...

```
/* CLAVES PARA LA LÓGICA DE PROGRAMACIÓN
Un triángulo ESCALENO tiene los tres lados diferentes.
Un triángulo EQUILÁTERO tiene los tres lados diferentes.
Un triángulo ISÓSCELES tiene dos lados iguales y uno diferente.
*/

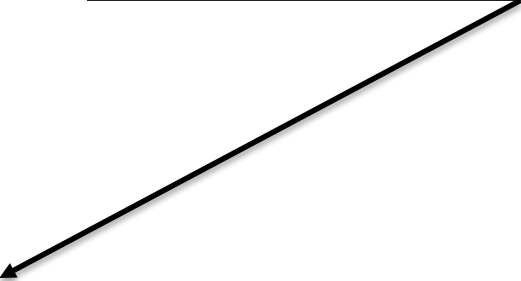
let readlineSync = require('readline-sync');
let lado1 = readlineSync.question('Medida de 1er lado: ');
let lado2 = readlineSync.question('Medida de 2do lado: ');
let lado3 = readlineSync.question('Medida de 3er lado: ');

if (lado1 == lado2 && lado2 == lado3)
  console.log('El triangulo es EQUILATERO (tres lados iguales)');
else if (lado1 == lado2 || lado1 == lado3 || lado2==lado3)
  console.log('El triangulo es ISÓSCELES (dos lados iguales y uno diferente)');
else
  console.log('El triangulo es ESCALENO (los tres lados diferentes)');
```

EJERCICIO 14 de la AP1 (con funciones).

```
function tipoTriangulo(l1,l2,l3) {  
  
    let tipo = '';  
    if (l1 == l2 && l2 == l3) {  
        // tres lados iguales  
        tipo='EQUILATERO';  
    } else if (l1 == l2 || l1 == l3 || l2==l3) {  
        // dos lados iguales y uno diferente  
        tipo='ISÓSCELES';  
    } else {  
        // los tres lados diferentes  
        tipo='ESCALENO';  
    }  
    return tipo;  
}
```

`tipoTriangulo(lado1,lado2,lado3)`



```
let readlineSync = require('readline-sync');  
let lado1 = readlineSync.questionInt('Medida de 1er lado: ');  
let lado2 = readlineSync.questionInt('Medida de 2do lado: ');  
let lado3 = readlineSync.questionInt('Medida de 3er lado: ');  
  
let result;  
result = tipoTriangulo(lado1,lado2,lado3);  
console.log('El triángulo es ' + result);
```

ACTIVIDAD PRACTICA 3 (AP3)

Título

Ejercicios básicos de programación (funciones)

Objetivos

- Conocer el uso de funciones (con y sin parámetros, con y sin retorno de valor) de una forma básica para utilizarlas en cualquier lenguaje de programación.
- Saber aplicar una programación modular mediante el uso de funciones utilizando la sintaxis de JavaScript.

Temporalización

Previsto: Utilización de dos sesiones presenciales + trabajo en casa.