


estructura básica de un programa informático

programación
javier ferrer

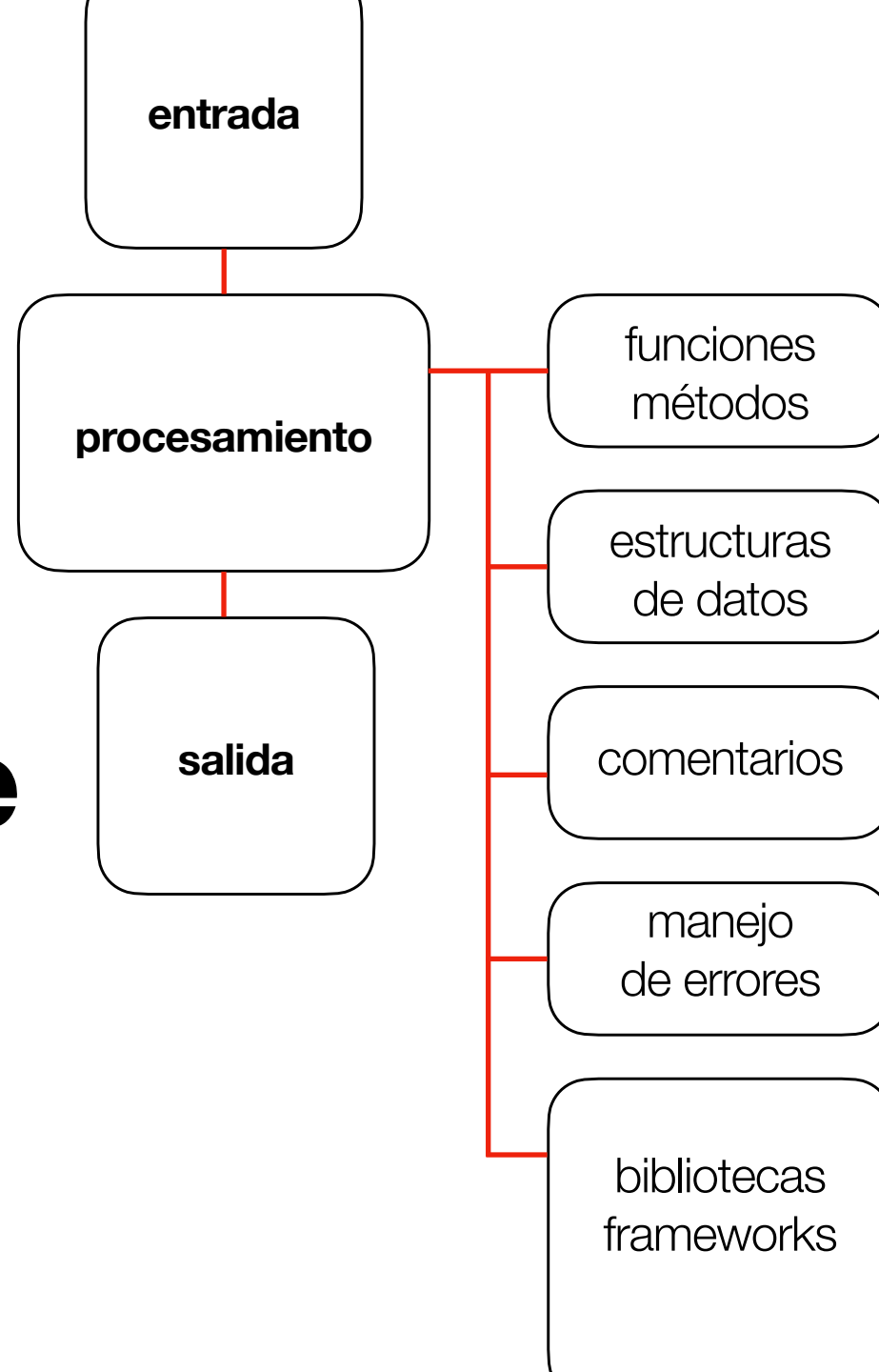
DAW-1 / 0485
RA-1/CE-A

IFP Innovación
en Formación
Profesional

De:

 Planeta Formación y Universidades

la estructura básica de un programa informático tiene que incluir siempre estos elementos básicos



la lógica del programa donde se manipulan los datos

incluye algoritmos, estructuras de control como bucles, condicionales, funciones y objetos

los resultados del procesamiento


puede mostrarse en la pantalla, en la consola de depuración, guardarse como archivos o enviar a otras aplicaciones



entrada

datos que el programa recibe para procesar

pueden ser desde el teclado, archivos, API, BBDD, etc...



Checkout form

Below is an example form built entirely with Bootstrap's form controls. Each required form group has a validation state that can be triggered by attempting to submit the form without completing it.

Billing address

First nameLast name

Username

@ Username

Email (Optional)

you@example.com

Address

1234 Main St

Address 2 (Optional)

Apartment or suite

CountryStateZip

Your cart

3

Product name	\$12
Brief description	
Second product	\$8
Brief description	
Third item	\$5
Brief description	
Promo code	-\$5
EXAMPLECODE	
Total (USD)	\$20

Promo code

Redeem

la lógica del programa donde se manipulan los datos

incluye algoritmos, estructuras de control como bucles, condicionales, funciones y objetos

procesamiento

```
let edad = 18;
if (edad >= 18) {
    console.log("Eres mayor de edad.");
} else {
    console.log("Eres menor de edad.");
}
```

este es un sencillo ejemplo en JavaScript sobre un condicional de los más usados **if/else**, en el asignamos el valor 18 a la variable edad, y el condicional comprueba si edad >=18 entonces nos muestra por consola “eres mayor de edad”, en caso contrario nos muestra “eres menor de edad”

funciones métodos

pueden ser bloques de código (normalmente reutilizables) que realizan tareas específicas

```
function saludar(nombre){  
    return 'Hola, ${nombre}!'  
}
```

el código de la izquierda es una función llamada saludar, la cual devuelve un String con un valor fijo que es Hola, y una variable que hemos entrado anteriormente.

```
console.log(saludar("Carlos"));
```

Una vez llamemos a la función, mediante en este caso con un console.log, nos mostrara la cadena de caracteres **“Hola, Carlos!”**



estructuras de datos

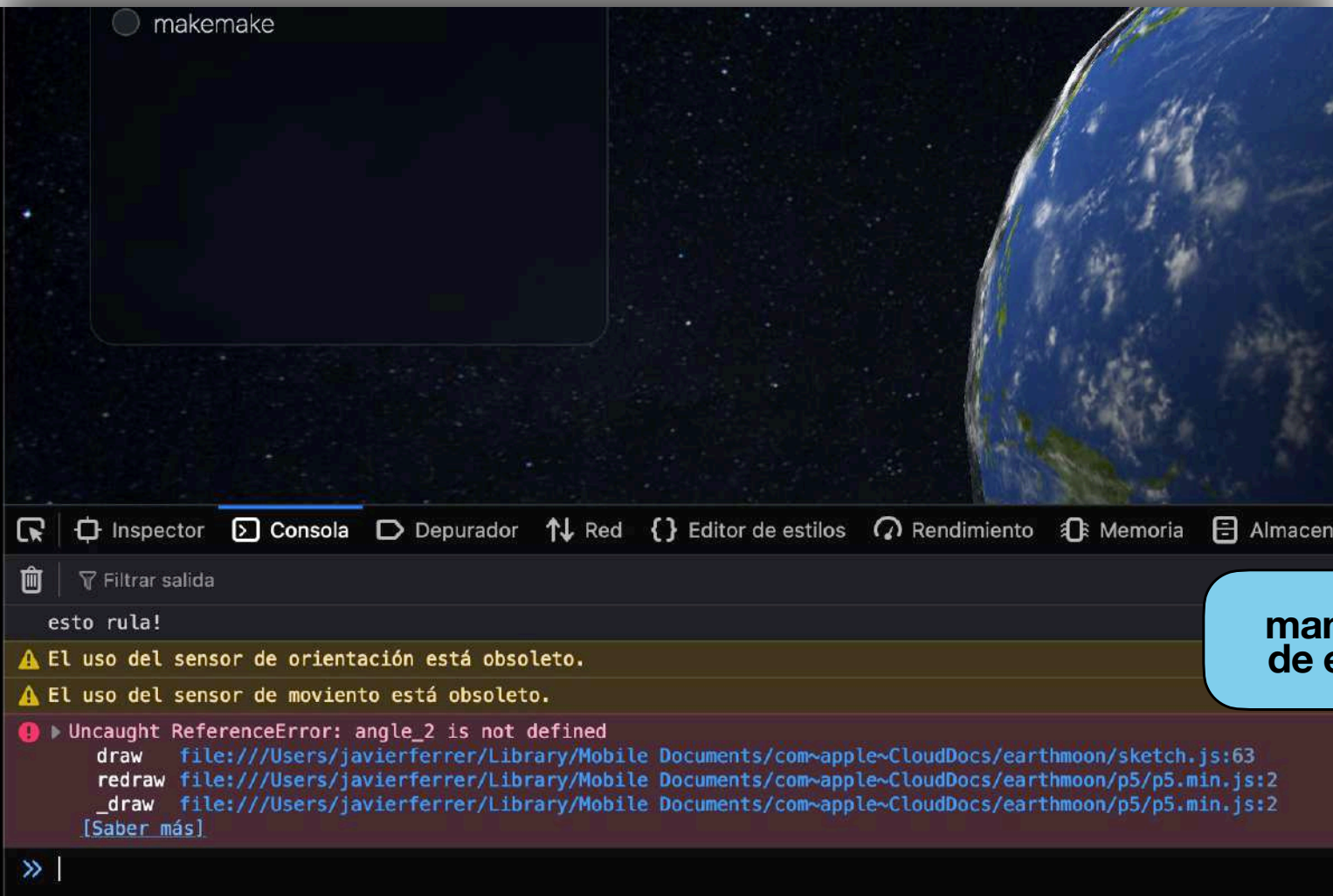
formas de organizar y almacenar datos, como listas u objetos

```
let superHeroes = ["batman", "saitama",  
  "superman", "wonder Woman", "black panther",  
  "Flash"]
```

```
10 //aquí empieza el header
20 <header>
30 <h1>Hola, soy el header</h1>
40 </header>
50 //aquí finaliza el header
```

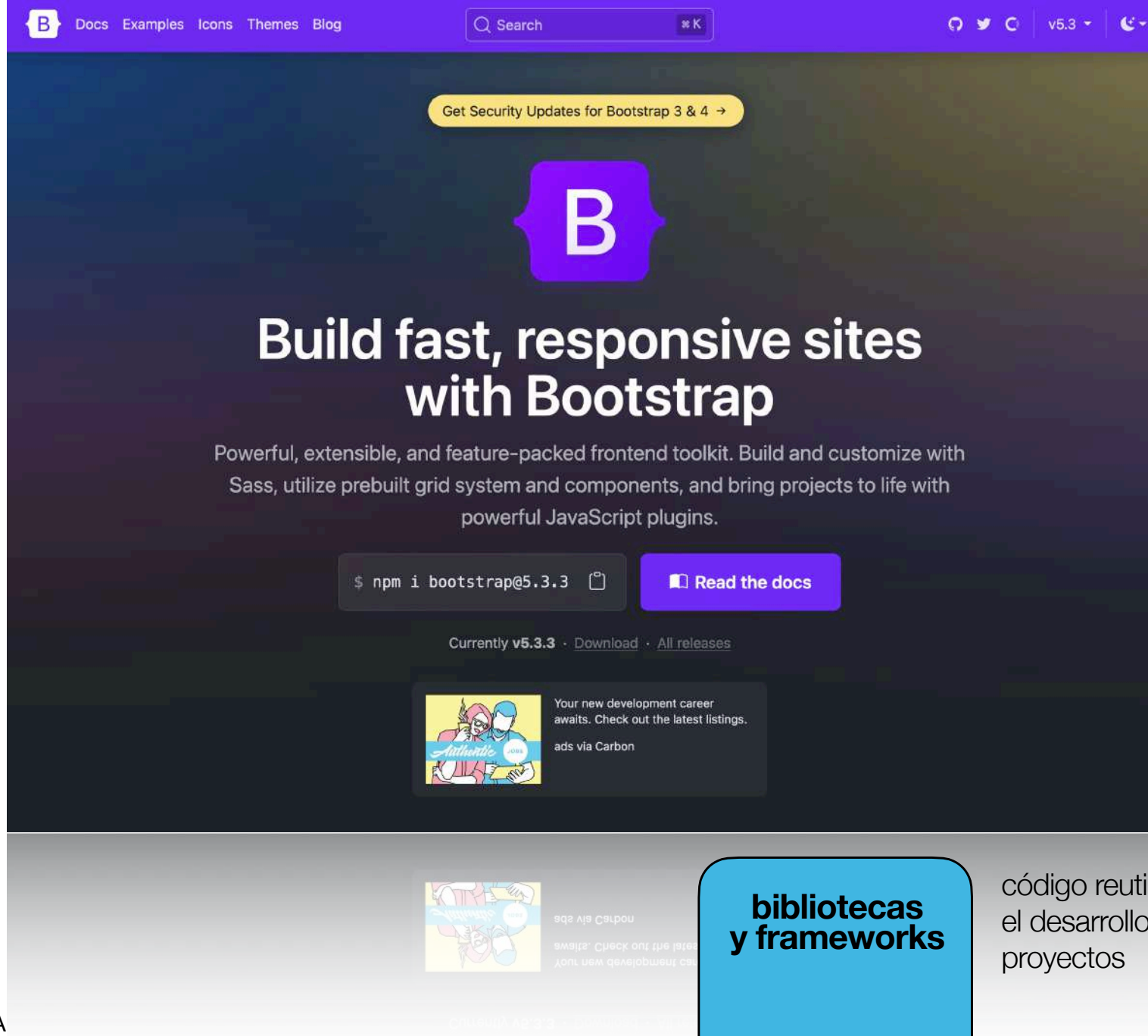
comentarios

explicaciones en el código para
facilitar la comprensión



manejo de de errores

mecanismos y herramientas para visualizar errores en tiempo de ejecución



**bibliotecas
y frameworks**

código reutilizable que facilita
el desarrollo de nuestros
proyectos

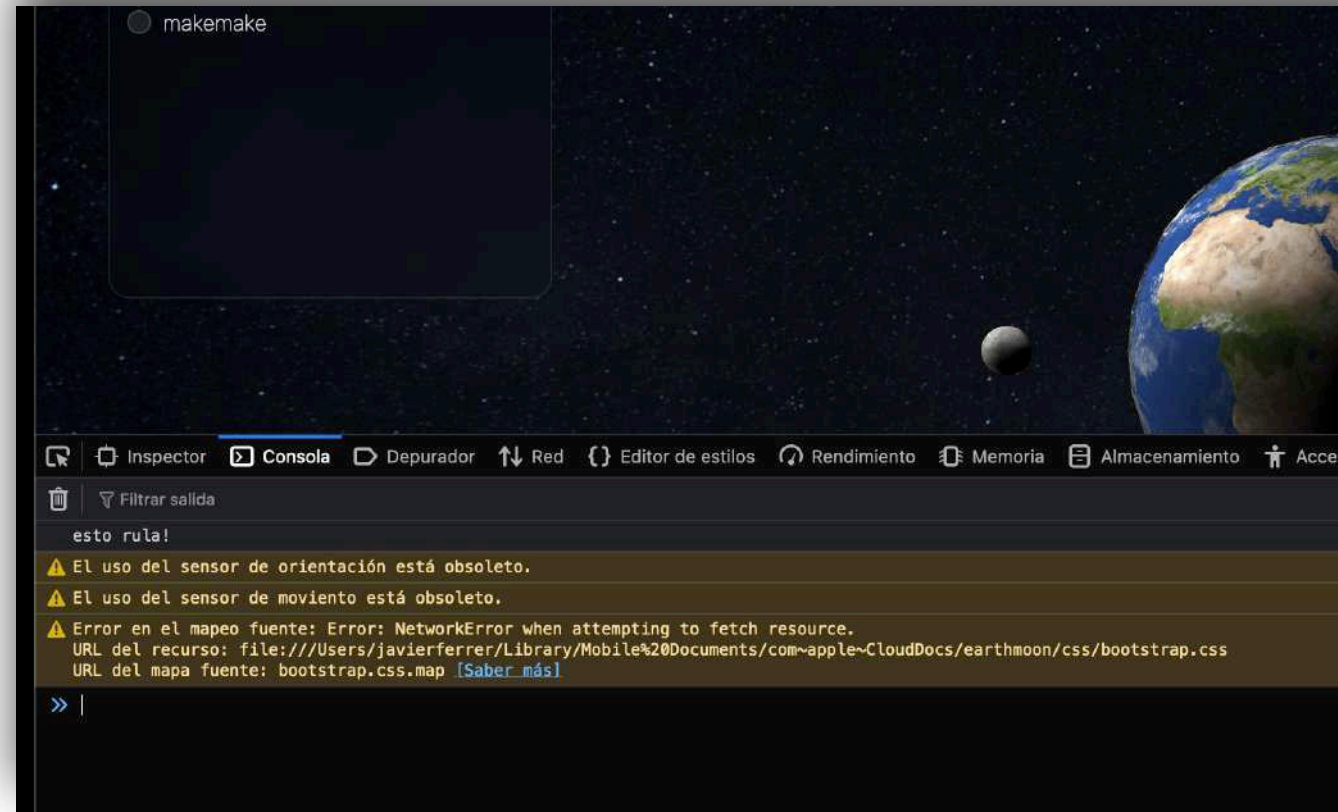
los resultados del procesamiento

puede mostrarse en la pantalla, en la consola de depuración, guardarse como archivos o enviar a otras aplicaciones

salida

este ejemplo de salida nos muestra el mensaje **“esto rula!”** mediante un comando **console.log** que sirve para mostrar mensajes en la consola de depuración.

Estos mensajes colocados a lo largo del código son esenciales para un poder disfrutar de un desarrollo dinámico y eficaz cuando creemos nuestros programas





¡Gracias por participar!

www.ifp.es

IFP Innovación
en Formación
Profesional

De:

 Planeta Formación y Universidades