

UD4 RESUMEN ULTRACONDENSADO



Funciones en JavaScript – Resumen en 1 página

◆ ¿Qué es una función?

Bloque de código con un nombre, que puede recibir parámetros y devolver un resultado.

◆ ¿Por qué se usan?

- Reutilizar código
- Reducir complejidad
- Mejorar legibilidad
- Facilitar mantenimiento

◆ Sintaxis básica

```
function nombre(param) { return valor; }
```

◆ Invocación

- `f()`
- `f(x)`

Si tiene `return`, su resultado se puede:

- guardarse en variables
- usarse en operaciones
- pasarse a otras funciones

◆ Return

- Finaliza la función.
 - Devuelve un valor.
 - Mejor usar **un único return**.
-

◆ Ámbito

- **Local:** dentro de una función.
- **Global:** fuera de funciones.

Variables con el mismo nombre no se pisan entre sí si están en ámbitos distintos.

◆ Parámetros

Por valor:

- Primitivos (números, strings, etc.)

Por referencia:

- Arrays y objetos (pueden modificarse).

Por defecto:

```
function f(a, b = 10) { ... }
```

Variables:

- `arguments`
 - `...rest` (preferido)
-

◆ Tipos de funciones

1. Declarativas

- Soportan hoisting.

2. Por expresión

```
const f = function() {}
```

3. Anónimas

- Sin nombre.

4. Callbacks

- Pasadas como argumentos.

5. Autoejecutables (IIFE)

```
(function(){}())
```

6. Flecha

```
const f = x => x*2;
```

◆ Recursividad

- Función que se llama a sí misma.
- Siempre debe tener:
 - Caso base → detener.
 - Caso recursivo → continuar.

Ejemplo típico: factorial y Fibonacci.

◆ Funciones avanzadas sobre arrays

sort()

- Ordena arrays.
- Puede recibir función de comparación.

forEach()

- Recorre elementos.

map()

- Crea nuevo array transformado.

filter()

- Crea nuevo array filtrado.