



```
1  (()=>{
2    const calc
3  })();
```

Avanzados y Funcione...



Curso de Fundamentos de TypeScript



→OBJETOS EN FUN...

Retorno de funciones 18/24



RECURSOS

MARCADORES

En TypeScript podemos especificar el tipo de dato del valor que nos retornará una función o indicar si no se devolverá dato alguno:

Retornos tipados en TypeScript

El tipo de retorno se especificará después de los paréntesis en los que se encuentran los argumentos de la función:

1. Void: funciones sin retorno

Este tipo de función ejecuta ciertas instrucciones, pero no devuelve dato alguno. Estas son conocidas como funciones de tipo `void`. Se definen así:

```
//TypeScript
function imprimirNombre(yourName: string): void {
    console.log(`Hello ${yourName}`);
}
```

2. Funciones con retorno

Por el contrario, si en la función devolveremos algún valor, podemos especificar el tipo de dato de este:

```
//TypeScript
function suma(a: number, b: number): number {
    return a + b;
}

function holaMundo(): string {
    return "Hello, World!";
}
```

También los retornos pueden ser más de un tipo de dato:

```
//TypeScript
function devolverMayor(a: number, b: number): number | string {
    if(a > b){
        // Retorna un número
    }
}
```

```
    return a;
  } else if( b > a ) {
    // Retorna un número
    return b;
  } else {
    // Retorna un string
    return `Los números ${a} y ${b} son iguales`;
  }
}
```

TypeScript también lo infiere

Si no indicamos en nuestra declaración de la función el tipado del retorno, TypeScript, al igual que con las variables, lo puede inferir según si retornas datos (sea `string` , `number` , etc.) o si nada es devuelto (tipo `void`).

Contribución creada por: Martín Álvarez.

