



Florida
Universitària

Introducción a TypeScript

Curso 2025/2026

Paco Segura

Introducción a TypeScript

- TypeScript \approx JavaScript mejorado. Tiene las características de JavaScript y otras nuevas. Es tipado.
 - Tipos:
 - boolean, number, string, null, undefined.
-

TypeScript:

- **Variables:**

- Regla general: cuando definimos una variable con un valor inicial, ya no se puede reasignar con un tipo diferente.
- Excepción: si la declaramos sin valor. Le asignara 'any'.

```
let hola;  
hola = 1;  
hola = false;
```

TypeScript:

- **Variables:**

- Las variables pueden tener declaraciones de tipo añadidas después de sus nombres. Se indican con : y el tipo.

```
let saludo : string;  
saludo = 'hola';  
  
saludo = 12; // error, tiene que ser de tipo string.
```

TypeScript:

- **Funciones:**

- También se puede realizar declaraciones de tipo a los parámetros de las funciones:

```
function myFunction(paramOne: number, paramTwo: string){  
    //...  
}
```

TypeScript:

- **Funciones:**

- Con un interrogante indicamos que el parámetro es opcional –es decir, se puede no pasar nada por parámetro-:

```
function check(state?: string) {  
  console.log(`${state} || 'no está listo...'`);  
}  
  
check();  
check('¿Listo?');
```

TypeScript:

- **Funciones:**

- Se puede indicar el valor por defecto de los parámetros:

```
function show(word = 'Mundo') {  
  console.log(`Hola, ${word}!`);  
}
```

TypeScript:

- **Funciones:**

- Si no devuelve nada, se puede indicar 'void':

```
function myFunction(param: string): void{  
    //...  
}
```

TypeScript:

- **Arrays:**
 - Podemos indicar el tipo obligatorio para un array -no se podrán poner elementos de otros tipos-:

```
let names: string[] = ['Dani', 'Ximo'];  
  
//otra forma equivalente:  
let names: Array<string> = ['Dani', 'Ximo'];
```

TypeScript:

- **Arrays:**
 - Para arrays de varias dimensiones:

```
let names: string[][] = [['Dani', 'Ximo'], ['Laura', 'Maria']];
```

TypeScript:

- **Arrays:**
 - Si se tienen varios tipos:

```
let myArray: [string, number, string, boolean] = ['Roberto', 7, 'Albacete', false];
```

TypeScript:

- **Unión de tipos:**
 - Más específico que any. Permite utilizar cualquiera de los tipos que se indica:

```
let id: string | number;
```

TypeScript:

- **Unión de tipos:**
 - Con arrays:

```
const myArray: (string | number)[] = [12, 'Jueves'];
```
