**TypeScript**

**Nunca.**

**Documentación:**

**<https://www.typescriptlang.org/docs/handbook/basic-types.html#never>**

[**https://www.typescriptlang.org/docs/handbook/2/everyday-types.html**](https://www.typescriptlang.org/docs/handbook/2/everyday-types.html)

Otro tipo de dato bastante interesante es el nunca (never). Hay que entender que el tipo de dato nunca, no es ni undefined, ni null, etc.

Al utilizar el tipo de dato nunca en una función, esta función no puede tener un punto de conexión alcanzable, es decir, significa que la función no debe terminar exitosamente o que la función no debe de tener un punto de retorno. Cuando nosotros miremos una función así, sabemos de antemano que después de llamar esa función que retorna el tipo de dato nunca, nuestro código de la aplicación ya no seguirá corriendo. En pocas palabras reventará el código.

**Ejemplo de función con nunca usada correctamente:**

const error = (mensaje: string): never => {

    throw new Error("Mensaje del error: " + mensaje )

}

error("Auxilio");

console.log("Hola Mundo!") // Este código jamás se ejecutará.

**Ejemplo de función con nunca usada incorrectamente:**

const error = (mensaje: string): never => {

    if( false ){

        throw new Error("Mensaje del error: " + mensaje )

    }

}

error("Auxilio");

console.log("Hola Mundo!") // Este código si se ejecutará.

**Nota Importante.**

**En las funciones también podemos especificar que se pueden retornar mas de 1 tipo de dato. La siguiente función puede retornar un tipo de dato nunca o número. Ejemplo:**

const error = (mensaje: string): (never | number) => {

    let condicion: boolean = true;

    let retorno: number = 200;

    if( condicion ){

        throw new Error("Mensaje del error: " + mensaje )

    }

    return retorno;

}

error("Auxilio");

console.log("Hola Mundo!") // Este código jamás se ejecutará.