**TypeScript**

**Cadenas.**

**Documentación:**

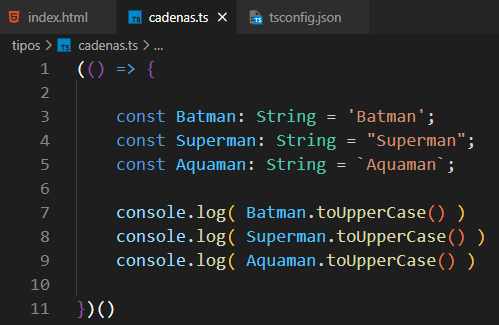
[**https://www.typescriptlang.org/docs/handbook/basic-types.html#string**](https://www.typescriptlang.org/docs/handbook/basic-types.html#string)

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

Para las cadenas de caracteres existen 3 formas de definirlas.

1. Comillas Simples ‘’
2. Comillas Dobles “”
3. Backquote / Backticks ``

Ejemplo:



Al momento de trabajar con las cadenas de texto se recomienda utilizar siempre los apostrofes normales comillas simples (‘’), ya que se supone que son más ligeras que las comillas dobles (“”). Pero si se necesita usar las comillas simples en una cadena de texto, entonces se recomienda utilizar las comillas dobles en la variable.

**Concatenaciones**

Una forma de hacer las concatenaciones un poco más interesantes es utilizar los backticks y dentro de la cadena de texto poner el código de ${ NombreVariable } para concatenar el valor de la variable en la cadena de texto.

Ejemplo: console.log( `Yo soy ${Batman}` )

**Control de error con NullCheck**

Hay que recordar que TS no es algo que nos va asegurar a que no cometamos errores.

Por ejemplo en el siguiente código: Digamos que vamos a utilizar la variable de Batman declarada en la parte de arriba y queremos obtener el carácter en la posición 11 y pasar eso a un toUpperCase.

Haríamos lo siguiente:

console.log( Batman[10].toUpperCase() )

Para el TS ese código tiene sentido a pesar de que sabe de Batman no tiene más de 10 caracteres, al ejecutar ese código en el JS tendríamos un error. Para controlar ese error, hay una cosa interesante que se llama NullCheck que podemos poner con un símbolo de interrogación.

Ejemplo:

console.log( Batman[10]?.toUpperCase() || 'No hay carácter' )

Esta forma se valida de la siguiente forma, si Batman en la posición 10 tiene algo “?”, entonces realiza el toUpperCase. Sino entonces manda el mensaje ‘No hay carácter’…