

What do Double Question Marks mean in TypeScript?

Posted At : May 7, 2020 9:00 AM | Posted By : Jeffry Houser

Related Categories: Professional, TypeScript

I find it fascinating when lesser experienced devs on my team do something incredibly awesome that I knew nothing about. It is great to be in a role where we can share knowledge both up and down the chain, without too much of a bruised ego. I was reviewing a PR that used a double quotation mark syntax, like this:

```
let x = foo ?? 'Some Default Value'
```

Huh! What does this code do?

Turns out this is a process called "Nullish Coalescing". I hope I never have to say that out loud. If the value of foo is null; then x will point to the value after the double question marks. If the value of foo is not null; then x will point to the value of foo.

In this sample:

```
let foo;  
const x = foo ?? 'Some Default Value';  
console.log(x);
```

X will output as "Some Default Value", because foo is undefined.

We can continue the code:

```
foo = 'some non default value';  
const y = foo ?? 'Some Default Value';  
console.log(y);
```

In this case, the value of y will be "Some non default value", pointing at foo.

If you ever need a quick shorthand way to set defaults, this is a great way to do it.

I have a few quick samples of this in the sample project I used for [Optional Chaining and Safe Navigation](#).

