## String.Format like C# in typescript

Asked 1 year, 2 months ago Active 3 months ago Viewed 3k times



Is it possible to use similar function like String. Format of C# in TypeScript?



My idea is to make some string like:



url = "path/{0}/{1}/data.xml"



depending of the logical I set {0} and {1}. Obiously I can replace them but I think String. Format is a clear function.

typescript





## 5 Answers



Expanding on the comment that I made on the response from vivekkurien, declaring a function which, in turn, interpolates is probably the largest "bang for your buck" approach. I use this, frequently, for generating chunks of repetitive HTML with minor varying properties, for example.



The answer from vivekkurien, however, does not work. It returns a literal string, instead. Here is a modified sample, based on the original answer:



```
const pathFn = (param1, param2) => `path/${param1}/${param2}/data.xml`;
let param1 = "student";
let param2 = "contantdetails";
let resultPath = pathFn(param1, param2);
alert(resultPath);
```

A runnable example of the above code can be found here: Function-Based Interpolation at TypeScript Playground

answered May 19 '18 at 20:49



String.format you can use an array and position 0 i the format and the next one are the values but all in an array of basic objects. Is it possible to do in Typescript? – David May 20 '18 at 12:20

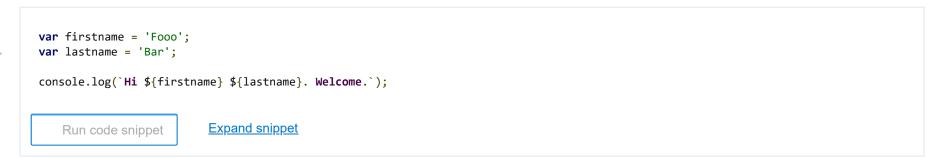
Not natively, that I am aware of. That being said, check out the second response to this question, which sounds very close to what you are looking for, there: <a href="mailto:stackoverflow.com/questions/20070158/...">stackoverflow.com/questions/20070158/...</a> – Joseph Ferris May 20 '18 at 17:43



I think you are looking for back quote: "







You can find the back quote on the tilde key.



ComputerHope.com

answered May 19 '18 at 18:58



String.format you can use an array and position 0 i the format and the next one are the values but all in an array of basic objects. – David May 20 '18 at 12:20



1

```
const StringFormat = (str: string, ...args: string[]) =>
    str.replace(/{(\d+)}/g, (match, index) => args[index] || '')

let res = StringFormat("Hello {0}", "World!")
    console.log(res) // Hello World!
    res = StringFormat("Hello {0} {1}", "beautiful", "World!")
    console.log(res) // Hello beautiful World!
    res = StringFormat("Hello {0},{0}!", "beauty")
    console.log(res) //Hello beauty, beauty!
    res = StringFormat("Hello {0},{0}!", "beauty", "World!")
    console.log(res) //Hello beauty, beauty!
```

## Try in TypeScript Playgroud

edited Nov 25 '18 at 5:57

answered Nov 25 '18 at 5:51



Edward Young 291 3 5



You need to make the path as a function which can accept 2 parameter. Then it will return as string if we call the function with required parameters.





```
const pathFn= (param1, param2) => "path/${param1}/${param2}/data.xml";
let param1 = "student";
let param2 = "contantdetails";
let resultPath = pathFn(param1,param2);
```

edited May 19 '18 at 18:47

answered May 19 '18 at 18:27



You would want to use bat quotes for the string interpolation to kick in, as well as use the interpolation marker (\$) for the replacement to happen. See: typescriptlang.org/play/... – Joseph Ferris May 19 '18 at 18:43



This worked for me. Tried with different scenarios.





format(text: string, ...args: string[]): string {
 return text.replace(/{(\d+)}/g, (match, num) => {
 return typeof args[num] !== 'undefined' ? args[num] : match;
 });
}

answered Apr 11 at 22:25

