

WBE-Praktikum 6

TypeScript

Aufgabe 1: Funktion definieren

Einstiegsübung

Schreiben Sie die Fakultätsfunktion aus Praktikum 3 (Aufgabe 2) neu mit TypeScript. Die Funktion soll mit *number* und mit *bigint* arbeiten:

```
> factorial(10)
3628800
> factorial(50n)
30414093201713378043612608166064768844377641568960512000000000000n
```

Aufgabe 2: Übung zu Strings (Abgabe)

Schreiben Sie eine TypeScript-Funktion *findTag*, welche aus einem String das erste Tag (also ein Text in spitzen Klammern, darf keine Leerzeichen enthalten) findet und den Tagnamen zurückgibt:

```
> findTag("<header>Text</header>")
"header"
> findTag("blabla <br> blabla")
"br"
> findTag("123245 </header> bla")
"/header"
> findTag("123245 <hea der> bla")
undefined
> findTag("123245 <hea<der> bla")
"der"
> findTag("123245 <hea<der bla")
undefined
```

Sie können mit `"blabla"[n]` auf das n-te Zeichen des Strings zugreifen und eine for-Schleife verwenden (es gibt aber auch andere Möglichkeiten).

Abgabe

Geben Sie die Funktion *findTag* ab. Da unser Abgabeserver noch kein TypeScript verarbeiten kann, kommentieren Sie den TypeScript-Code aus und hängen den vom Transpiler erzeugten JavaScript-Code für den Jasmine-Test an:

```
/*
```

```
TypeScript-Funktion...
```

```
*/
```

Generiertes JavaScript...

Abgabeserver: <https://radar.zhaw.ch/python/UploadAndCheck.html>

Name Praktikum: WBE7

Dateiname: find-tag.js

Funktionsname: findTag

Export im Script: `module.exports = { findTag }`

Aufgabe 3: Übungen zu TypeScript (fakultativ)

Falls Sie noch etwas TypeScript üben möchten, finden Sie hier einige Übungen, beginnend mit sehr einfachen Aufgaben und dann ansteigendem Schwierigkeitsgrad:

<https://typescript-exercises.github.io>