

Angular Tutorial for Beginners – Part 4 : Interfaces

- *Tuba Mansoor*

This is part 4 of the Angular Tutorial for Beginners series.

Part 1 of this series discussed how to get started with Angular.

Part 2 discussed about the basic tenets of Angular Modules, Components and Directives.

Part 3 discusses data binding.

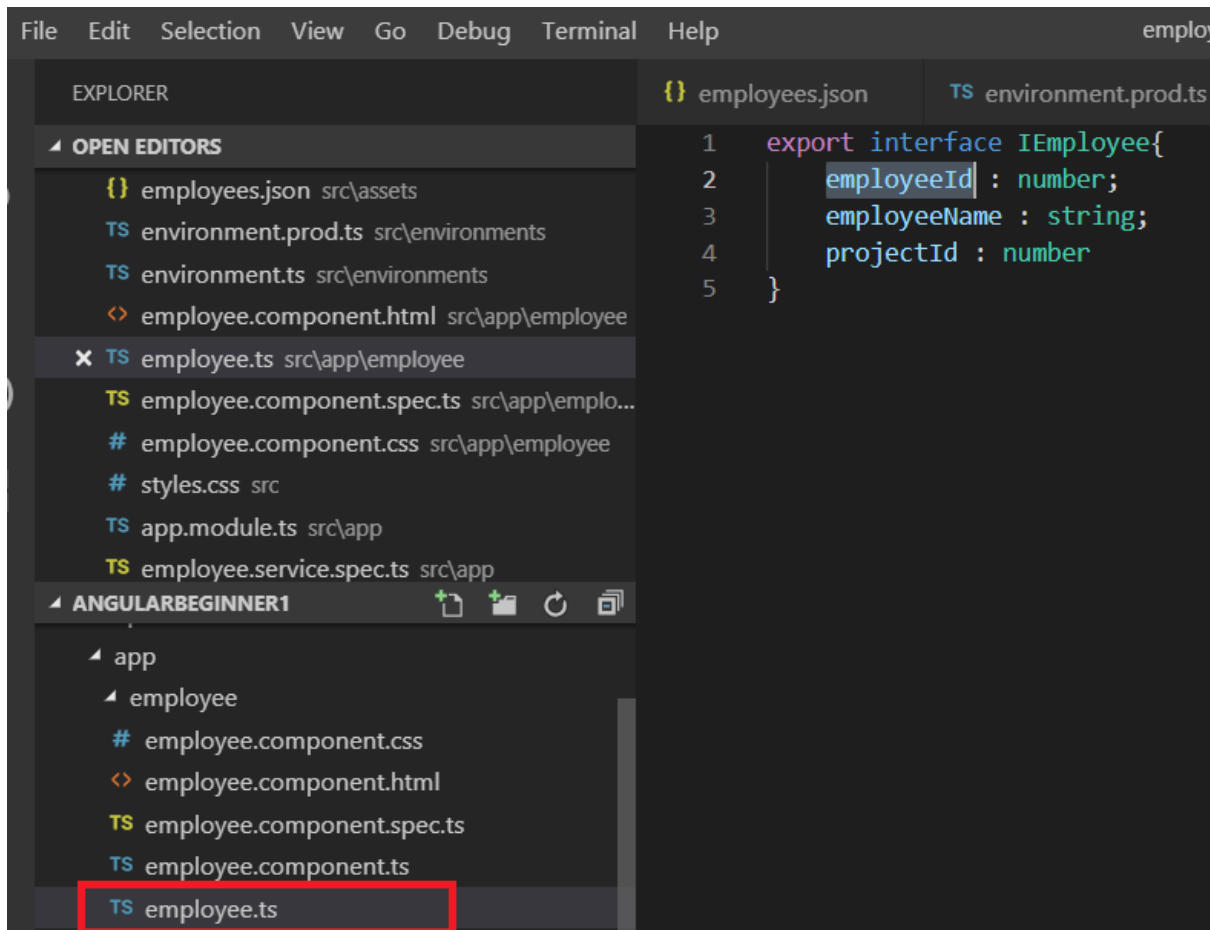
Part 4 here builds up on the demo used in the earlier parts to explain the concept of interface. Hence it is strongly advised that you go through them before proceeding.

Interfaces in Typescript provide type checking. Here we have defined an array of **any** as below. But this does not provide any type checking.

```
employees : any[] =[ {  
  "employeeId" : 1,  
  "employeeName" : "Tuba",  
  "projectId":100  
},  
{  
  "employeeId" : 2,  
  "employeeName" : "Atul",  
  "projectId":101  
},  
{  
  "employeeId" : 3,  
  "employeeName" : "Theran",  
  "projectId":101  
}  
]
```

For type checking we will create, an interface of employee as below.

We will create typescript file employee.ts and export an interface as below.



The screenshot shows the Visual Studio Code interface. The Explorer sidebar on the left shows the project structure with 'employee.ts' highlighted in red. The Open Editors sidebar shows 'employee.ts' as the active file. The main editor area displays the following TypeScript code:

```
1 export interface IEmployee{
2     employeeId : number;
3     employeeName : string;
4     projectId : number
5 }
```

```
1 export interface IEmployee{
2     employeeId : number;
3     employeeName : string;
4     projectId : number
5 }
```

Let's import the employee.ts in employee.component.ts.

```
TS employee.component.ts x TS app.module.ts TS employee.ts TS employo
1 import { Component, OnInit } from '@angular/core';
2 import { IEmployee } from './employee'
3 @Component({
4   selector: 'app-employee',
5   templateUrl: './employee.component.html',
6   styleUrls: ['./employee.component.css']
7 })
```

Now we replace the array of any with an array of IEmployee in employee.component.ts as below.

```
employee.component.ts x TS environment.pro
4
5 employees : IEmployee[] = [ {
6   "employeeId" : 1,
7   "employeeName" : "Tuba",
8   "projectId":100
9 },
10 {
11   "employeeId" : 2,
12   "employeeName" : "Atul",
13   "projectId":101
14 },
15 {
16   "employeeId" : 3,
17   "employeeName" : "Theran",
18   "projectId":101
19 }
20 ]
21 ngOnInit() {
```

Using an interface like IEmployee will provide me with a type checking. If I write 'employeeNames' instead of 'employeeName', I will be notified immediately as below. So, this helps in compile time checking and helps me to catch errors earlier on.

```
26     "employeeId" : 3,  
27     "employeeNames" : "Theran",  
28     "projectId" : 101  
29   }  
30 ]  
31 ngOnInit() {  
32  
33  
34 }  
35  
36  
37 information = '';  
38 onClickInformation() { this.information = 'This is a list of employees'; }  
39
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL 1: node

```
i [wdm]: Compiling...  
Date: 2019-05-13T15:01:22.880Z - Hash: 1f15b88bdf7c45a8d43e - Time: 329ms  
5 unchanged chunks  
chunk {main} main.js, main.js.map (main) 17 kB [initial] [rendered]  
i [wdm]: Compiled successfully.  
  
ERROR in src/app/employee/employee.component.ts(27,7): error TS2322: Type '{ "employeeId": number; "employeeNames": string; "proj  
assignable to type 'IEmployee'.  
  Object literal may only specify known properties, and "employeeNames" does not exist in type 'IEmployee'.
```

So, what if we use a class instead of an interface for type checking. There is no concept of interfaces in JavaScript. Hence when interfaces are transpiled, there is no code generated. But if we use classes, Javascript code is generated. Using interface, you can avoid this extra javascript code. If you only need to perform type checking, then using interfaces instead of classes is good option. is a good read to know more about Typescript classes & interfaces.

In the **next part** of the Angular tutorial for Beginners series we will learn how to use services in angular.

This article was originally published on my [website](#).