Typescript

- Typescript is a compile-time technology that adds **types** to Javascript.
- Typescript is an *optional* addition to Javascript.

Installation

- Install Typescript using NPM.

```
npm install -g typescript
```

- Test the version works.

```
tsc -v
```

Basic usage

- Create an example file **code.ts**

```
let upper = ( s:string ) => s.toUpperCase();
console.log(upper("seville"));
```

- Open the command line in this folder and run Typescript. The -w argument watches for later file changes.

```
tsc code.ts -w
```

- Typescript will create transpiled code in **code.js**. Run this code using Node.

```
node code.js
```

Basic types

```
let year : number = 2016;
let smoker : boolean = false;
let city : string = "Seville ";
let project : any = true;
```

Arrays

```
    let fruit:string[] = [ "Apples","Pears" ];
    let fruit:Array<string> = [ "Apples","Pears" ];
```

Functions

```
function logger(s) : void { console.log(s); }

let double = (n:number) : number => n*2 ;
```

Enum

```
enum Size { ExtraSmall, Small, Medium, Large, ExtraLarge };
let xl: Size = Size.ExtraLarge;

console.log( Size[0], typeof Size[0] );
console.log( Size.ExtraSmall, typeof Size.ExtraSmall );
```

Classes

- Typescript extends ES6 classes with access modifiers.

```
class Student {

    private name;
    private age;

    constructor( n,a ) {
        this.name = n;
        this.age = a;
    }

    describe() {
        console.log( this.name, this.age );
    }
}

let p = new Student( "Fred", 45 );
p.describe();
```

```
console.log( p.name ); // compile error
```

- Attempting to access p.name triggers a compile-time error.
- The class code can be simplified:

```
class Student {

    constructor( private name, private age ) {}

    describe() {
        console.log( this.name, this.age );
    }
}
```

Interfaces

- Interfaces let you define a minimum set of public properties and methods that a class implements.

```
interface Person{
    name:string;
    age:number;
}

class Student implements Person {

    constructor( public name:string,
    public age:number ) {}

    describe() {
        console.log( this.name, this.age );
    }
}

let p = new Student( "Fred", 45 );
p.describe();
```