

Introduction

Ce cours est une introduction au langage TypeScript. Nous le présentons ici comme un module à part entière car, Angular est écrit en TypeScript.

Le TypeScript permet d'écrire du code typé de manière statique ou non. Il possède également tous les paradigmes d'un langage objet. Il est donc très adapté à la conception d'application.

TypeScript est un « sur-ensemble » au Javascript. Le code TypeScript sera transcompilé, ré-écrit en code Javascript compatible pour tous les navigateurs. Il faudra cependant préciser la cible ou la norme que l'on souhaite lors de cette phase (ES5, ES6, ...).

Nous allons décrire dans un premier temps la configuration de base à faire sur votre PC pour suivre les exercices.

Nous vous donnons également un lien vers la documentation officielle :
<https://www.typescriptlang.org/docs/home.html> (Handbook)

TypeScript permet de définir des types avancés, ce n'est pas un typage statique, mais lorsque vous travaillez avec ce langage nous vous invitons à définir les types de toutes vos variables.

En TypeScript vous verrez parfois un type particulier « any », ce type est à éviter. En effet, le typage vous apportera de la sécurité dans vos développements. Un type « any » laissera passer n'importe quel type dans vos méthodes ou fonctions au niveau des paramètres de ces dernières.

Le typage n'est donc pas obligatoire dans TypeScript, mais il est indispensable à la conception d'application complexe.

Voici quelques exemples de typage de variable dans TypeScript :

Types simples exemples

```
// type boolean
let isDone: boolean = false;

// exemples de type number
let hex: number = 0xf00d;
let decimal: number = 6;

// type string
let color: string = "blue";

// Type de Array de number
let list: number[] = [1, 2, 3];
```

Type de retour pour une fonction

```
// type de retour void la fonction ne retourne rien
function sayHello(): void {
  console.log("Hello World");
}
```

Type de retour plus technique

```
class User {
  status: number | string;
}

function showStatus(user: User): string | number {
  return user.status;
}
```

On peut préciser le type d'une variable a posteriori

```
let message: any = "this is a string";
let strMessage: number = (<string>message).length;
```

Ce typage est équivalent à

```
let message: any = "this is a string";
let strMessage: number = (message as string).length;
```

Installation

Il faut bien sûr avoir node et npm installés (versions récentes).

```
npm install -g typescript
```

Faites un **tsc -v** en console pour vérifier la version. Créez un dossier **TypeScript** sur votre bureau

Par la suite vous utiliserez comme éditeur de code **Visual Studio Code**.

Notons qu'Angular nécessite TypeScript dans une version $\geq 2.7.2$

