

TYPESCRIPT : PRÉSENTATION



semifir

LES FAIBLESSES DE JAVASCRIPT

- Typage dynamique peu sécurisé et parfois peu lisible,
- Sur gros projet : difficile à structurer
- N'est pas orienté objet :
 - Pas de classes mais des prototypes,
 - Héritages possibles, mais pas polymorphisme
 - Tout est publique

SOLUTION : TYPESCRIPT !

- Il reprend la syntaxe de Javascript,
- Ajoute la notion d'objets,
- Un typage fort,
- Notions de classes proche du Java ou C#
- Conserve les fonctionnalités d'ES6

AVANTAGES DE TYPESCRIPT

Grâce à la POO, on peut :

- Mieux structurer le code,
- Rendre le code plus robuste,
- Mettre en place des design-patterns

AVANTAGES DE TYPESCRIPT

Développé et maintenu par Microsoft :

- Sous licence Apache 2.0
- Utilisable commercialement
- Modifiable et distribuable

le code source [ici](#) !

AVANTAGES DE TYPESCRIPT

- Comparable à NodeJS en terme de performances
- Certaines tâches sont même plus rapides
- Utilisé par de nombreux frameworks (Angular, NetsJS ...)

AVANTAGES DE TYPESCRIPT

- TypeScript utilise une extension '.ts'
- Similaire à '.js' pour JavaScript
- Un syntaxe proche du js

AVANTAGES DE TYPESCRIPT

- TypeScript doit être transpilé en JavaScript
- On peut utiliser du code JS dans du TS

INSTALLATION



INSTALLATION

Pour être utilisé dans un projet, il doit être installé avec NPM.

```
npm install -g typescript
```

EXECUTER DU CODE

Comme pour le JavaScript, il n'est normalement pas possible d'exécuter le code, ce dernier devant être interprété par le navigateur.

Il est toutefois possible de transpiler le code en javascript, avant de l'exécuter avec node.

```
1 # transpiler le code en js :  
2 tsc hello  
3 # Executer le code avec node :  
4 node hello.js
```

MODULE

- Pour éviter de multiplier les commandes, on peut déléguer cette gestion au module ts-node

```
npm i -g ts-node
```

Ce faisant, on pourra démarrer les projets avec

```
ts-node <nomFichier>
```

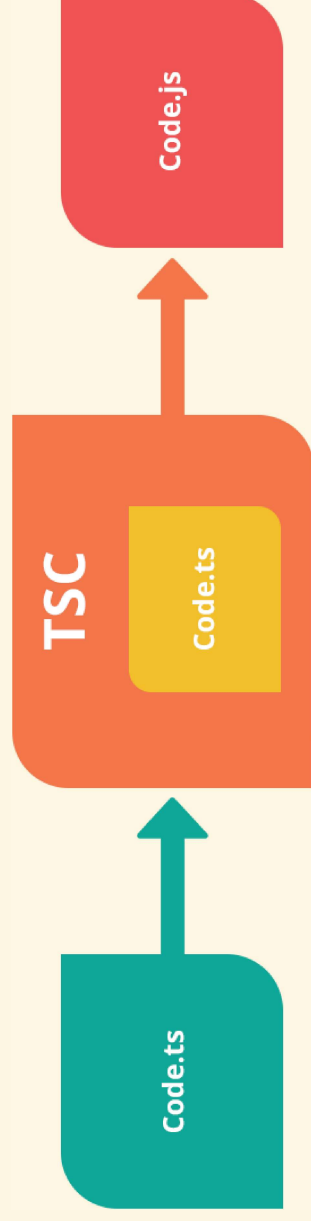
DEMONSTRATION

Hello world !

Note : ./Hello-world.md

TRANSPILATION





- Le navigateur ne peut pas lire le TypeScript
- Le projet doit être 'traduit' en JavaScript pour être utilisable

TRANSPILATION

- On dit alors qu'il est 'Transpilé'
- Transpile = **Transform** + **Compile**

TRANSPILATION

- Pour réaliser la transpilation, il faut utiliser un fichier de configuration
- La config par défaut est utilisée lorsque l'on exécute la commande **'tsc monScript.ts'**

TRANSPILATION

- Il est possible d'affiner les options de la transpilation

```
tsc --init
```

Créé un fichier `tsconfig.json` à la racine du répertoire courant.

TRANSPIRATION

```

{
  "compilerOptions": {
    /* Visit https://aka.ms/tsconfig.json to read more about this file */

    /* Projects */
    // "incremental": true,
    // "composite": true,
    // "tsBuildInfoFile": "./",
    // "disableSourceOfProjectReferenceRedirect": true,
    // "disableSolutionSearching": true,
    // "disableReferencedProjectLoad": true,

    /* Language and Environment */
    "target": "es2016",
    // "lib": [],

    /* Set the JavaScript compiler to use to compile JavaScript files */
    // "specify a set of

```

TRANSPILATION

- Le fichier comporte tout un tas d'options que l'on peut décommenter pour activer.
- On peut notamment y changer la version d'**ECMAScript** qui est utilisée (**ES2016** par défaut)
- Ou encore désactiver le mode strict.

TRANSPILATION

*Il existe beaucoup d'options
supplémentaires qui ne seront pas
abordées en détail ici.*

LA SUITE

Par ici !

