

# Bases TypeScript

## Installation

Installer TypeScript

```
npm install --save-dev typescript
```

Vérifier l'installation :

```
npx tsc --version
```

## Création de projet

Dans un nouveau dossier, initialiser TypeScript :

```
npx tsc --init
```

Un fichier tsconfig.json est créé :

```
{
  "compilerOptions": {
    "target": "ES6",
    "module": "CommonJS",
    "strict": true,
    "outDir": "./dist",
    "rootDir": "./src",
    "esModuleInterop": true
  }
}
```

- **target**: Spécifie la version ECMAScript de sortie.
- **module**: Définit le système de modules (CommonJS pour Node.js, ESNext pour les modules modernes).
- **strict**: Active les vérifications strictes.
- **outDir**: Indique où compiler les fichiers .ts (ex: dist).
- **rootDir**: Définit le dossier source (ex: src).
- **esModuleInterop**: Active l'interopérabilité avec les modules CommonJS.

## Structuration

A créer :

```
/mon-projet
├── src/
│   ├── index.ts
│   └── utils.ts
├── dist/ (généré après compilation)
├── tsconfig.json
├── package.json
└── node_modules/
```

## Test index.ts

```
const greet = (name: string): string => {
  return `Bonjour, ${name}!`;
};

console.log(greet("Alice"));
```

## Compilation

Le code **ts** est transpilé en **js**.

```
npx tsc
```

## Exécution

```
node dist/index.js
```

Pour une exécution directe, il faut installer ts-node :

```
npm install --save-dev ts-node
```

puis exécuter :

```
npx ts-node src/index.ts
```

## Exercices

- [TD JS-TS](#)

From:  
<http://slamwiki2.kobject.net/> - **SlamWiki 2.1**

Permanent link:  
<http://slamwiki2.kobject.net/web/ts>

Last update: **2025/08/12 02:35**

