

Voici les commandes pour initialiser un projet **TypeScript** avec **Express**, en suivant les meilleures pratiques :

---

## 1. Initialiser le projet

bash  
Copier le code  
`npm init -y`

---

## 2. Installer les dépendances

### Installations principales :

bash  
Copier le code  
`npm install express`

### Installations de développement :

bash  
Copier le code  
`npm install --save-dev typescript ts-node @types/node @types/express nodemon`

---

## 3. Initialiser TypeScript

bash  
Copier le code  
`npx tsc --init`

Cela génère un fichier **tsconfig.json**. Vous pouvez ajuster les options importantes comme suit :

json  
Copier le code  

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

- **outDir** : Où placer les fichiers JavaScript compilés (par défaut **dist**).
- **rootDir** : Où se trouvent les fichiers TypeScript sources (par défaut **src**).

---

## 4. Configurer le dossier principal

Créez le dossier `src` pour vos fichiers sources et un fichier de démarrage `src/app.ts`.

---

## 5. Configurer Nodemon

Ajoutez un fichier `nodemon.json` pour que Nodemon surveille les fichiers `.ts` :

json

Copier le code

```
{
  "watch": ["src"],
  "ext": "ts",
  "exec": "ts-node ./src/app.ts"
}
```

---

## 6. Ajouter les scripts dans `package.json`

Ajoutez les scripts suivants dans votre `package.json` :

json

Copier le code

```
"scripts": {
  "start": "nodemon",
  "build": "tsc",
  "serve": "node dist/app.js"
}
```

- **start** : Lance Nodemon avec `ts-node` pour un développement rapide.
  - **build** : Compile le TypeScript en JavaScript.
  - **serve** : Démarre l'application avec les fichiers compilés (utile en production).
- 

## 7. Lancer le serveur

En mode développement (avec Nodemon) :

bash

Copier le code

```
npm start
```

**Compiler le code TypeScript :**

bash

Copier le code

```
npm run build
```

### Exécuter les fichiers compilés :

```
bash
```

Copier le code

```
npm run serve
```

---

### Votre structure finale :

```
csharp
```

Copier le code

```
project-folder/
```

```
├─ node_modules/
```

```
├─ src/
```

```
│   └─ app.ts
```

```
├─ dist/                                # Généré après compilation
```

```
├─ package.json
```

```
├─ tsconfig.json
```

```
├─ nodemon.json
```

```
└─ package-lock.json
```