**2. tsconfig.build.json**

This is the TypeScript configuration file specifically used for building the NestJS application. The tsconfig.build.json file usually contains build-specific configurations, such as excluding test files from the build output.

**Example of tsconfig.build.json**

json

CopyEdit

{

"extends": "./tsconfig.json",

"compilerOptions": {

"outDir": "./dist",

"rootDir": "./src",

"noEmit": false,

"sourceMap": true,

"removeComments": true

},

"include": ["src/\*\*/\*"],

"exclude": ["node\_modules", "test/\*\*/\*.ts"]

}

**Explanation:**

* **extends**: Inherits the settings from the tsconfig.json file. This is useful for reusing common settings.
* **compilerOptions**:
  + **outDir**: Specifies where the compiled JavaScript files will go (dist directory in this case).
  + **rootDir**: The root directory of the source code (src directory).
  + **noEmit**: Prevents the compiler from generating output files (set to false to allow compilation).
  + **sourceMap**: Generates .map files that map the compiled JavaScript code to the TypeScript code for debugging.
  + **removeComments**: Removes comments from the output code to keep the compiled files clean.
* **include**: Specifies which files should be included in the compilation (src/\*\*/\* to include all files in the src directory).
* **exclude**: Specifies which files or directories should be excluded from the compilation (e.g., test/\*\*/\*.ts to exclude test files).

**How It Works:**

* The tsconfig.build.json file allows you to customize how TypeScript should behave during the build process.
* When you run nest build, the NestJS CLI uses this configuration to transpile TypeScript files into JavaScript, and the files are outputted to the dist folder.