Node.js versions

Dated august 2023, Compodoc is tested and compatible with only last [active versions](https://nodejs.dev/fr/about/releases/) of Node.js (see this [link](https://angular.io/guide/versions) for more information) : v14.x, v16.x, v18.x, v20.x

Angular-CLI

Dated august 2023, Compodoc supports last Angular-CLI version : 16.x

Just run Compodoc in a fresh or existing project.

Global installation

Install from npm :

npm install -g @compodoc/compodoc

If you use PowerShell on Windows, add quotes :

npm install -g "@compodoc/compodoc"

Local installation

Install with Angular CLI : npm scripts + special tsconfig.doc.json file will be created.

ng add @compodoc/compodoc

or directly

npm install --save-dev @compodoc/compodoc

Run

Create a file named tsconfig.doc.json, containing a key include pointing to src folder, you can also use exclude key :

{

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

"exclude": ["src/test.ts", "src/\*\*/\*.spec.ts", "src/app/file-to-exclude.ts"]

}

Define a script task for it in your package.json (with npm 6.x) :

"scripts": {

"compodoc": "npx compodoc -p tsconfig.doc.json"

}

and run it like a normal npm script :

npm run compodoc

or with npx :

npx @compodoc/compodoc ...

See [usage](https://compodoc.app/guides/usage.html) for more details.

Position of tsconfig file in codebase

Compodoc start at the folder level of the tsconfig file provided with -p option.

Example for an Angular CLI project :

.

├── src

│ ├── app

│ │ ├── app.component.ts

│ │ └── app.module.ts

│ ├── main.ts

│ └── ...

├── tsconfig.app.json

├── tsconfig.doc.json

└── tsconfig.json