

Basic Angular overview

Typescript essentials

Vital terminology

- Framework
- Compilation
- Single-page application (SPA)
- DOM, rendering
- Strict typing
- Application build process



Meet the Angular

Angular is a modern TypeScript-based frontend framework for building scalable maintainable enterprise-grade single-page applications.

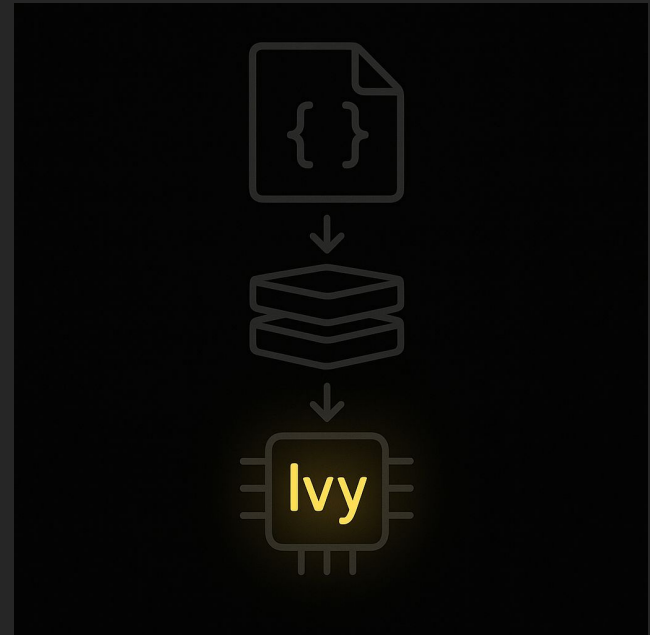


A short deep dive

How does Angular frontend app work? It passes a long compilation journey.

- Source code (TS)
- Bundle (JS) - affected by what you write in TS classes.
- Rendering logic (JS, Ivy instructions) - affected by what you write in the components' templates.
- Rendering / painting logic (Plain JS DOM interaction instructions)

Thus, the decisions taken in the Source code are affecting the resulting app both explicitly and silently.



First steps

After installing [Angular CLI](#) just a few commands needed to see your first Angular app running:

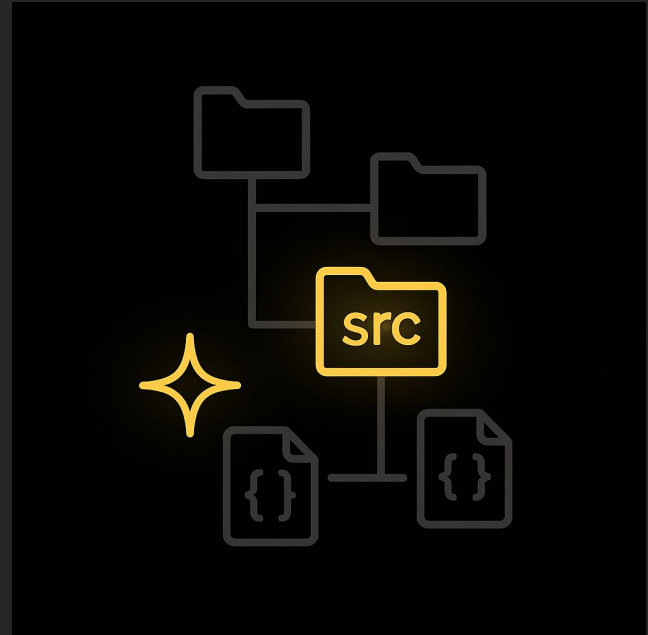
- [ng_new](#) my-angular-app
- [ng_generate](#) component my-app-component
- [ng_serve](#) / [ng_build](#)



Project structure

[Read more about general structure](#)

[Read more about angular.json](#)

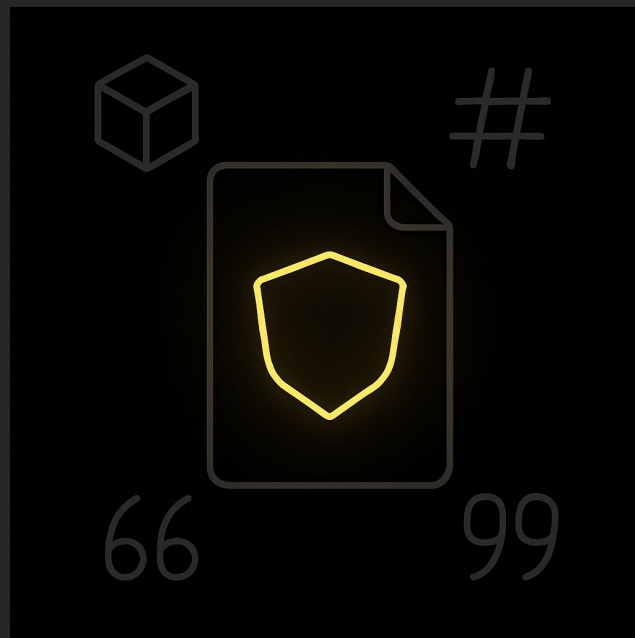


TypeScript

TypeScript brings clarity to the source code so eerie developer should utilize its power.

Here are some basic hints in case it's your first TypeScript encounter:

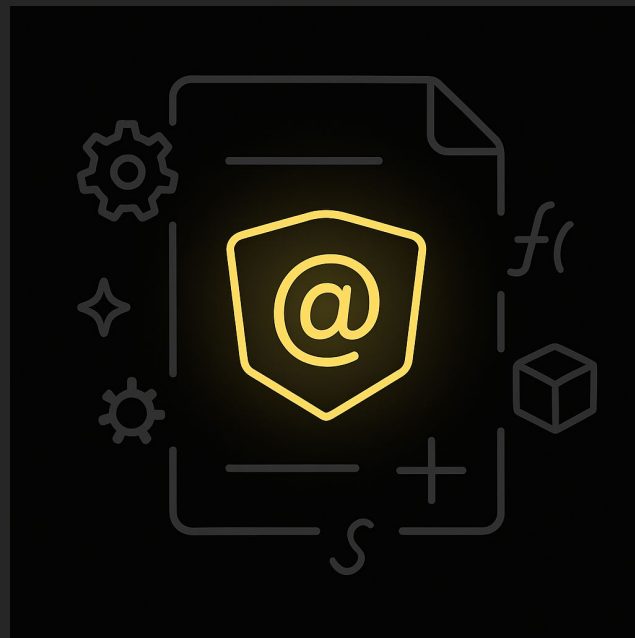
- Don't use **any** data type!
- Don't use enums, consider having a constant object instead
- There can't be too much of explicitness and generics!



Decorators

Sometimes decorators may be perceived as Angular's feature, but this is a delusion. This is a [powerful TS feature](#) that facilitates sophisticated organisation of Angular entities.

It is basically enough to treat decorators as the tools for metadata allocation



H/A

- Create your first Angular app with **ng new** and run it with **ng serve** (if you haven't tried this out yet).
- Ask this custom [ChatGPT agent](#) for materials, quizzes and practical tasks for the preparation for the topic #2 - Angular entities, components and pipes. You can also include module vs. standalone approach to this prep. Feel free to ask this agent any kind of Angular related questions but be ready for the fact it will also propose to practise the things with quiz or refactoring tasks as it is instructed to.