Angular

- Angular is a TypeScript-based framework created by Google for developing client-side web applications for mobile and desktop.
- The Angular framework's fundamental building pieces are Angular components arranged into **modules** which group together similar code to form functional sets.

Angular

- TypeScript is a superset of JavaScript developed by Microsoft that brings static typing, enhanced error checking, and object-oriented programming paradigms to the language before it is converted to JavaScript and processed in web browsers.
- Furthermore, it works more tightly with code editors and integrated development environments (IDEs), improving the developer experience, especially when implementing large JavaScript applications that should scale.

Angular

- Angular enables building frontends with both modular and component-based designs, which helps developers develop application using modules and/or components.
- When designing an Angular application, you may arrange it into modules and/or components that can be utilized across the application

Angular modules

- Module in Angular refers to a place where you can group the components, directives, pipes, and services, which are related to the application.
- To define module, we can use the NgModule.
- We already have two modules in our Angular CLI-generated project: src/app/app.module.ts and src/app/app-routing.module.ts, which export the classes needed to create the main application module and the routing module.

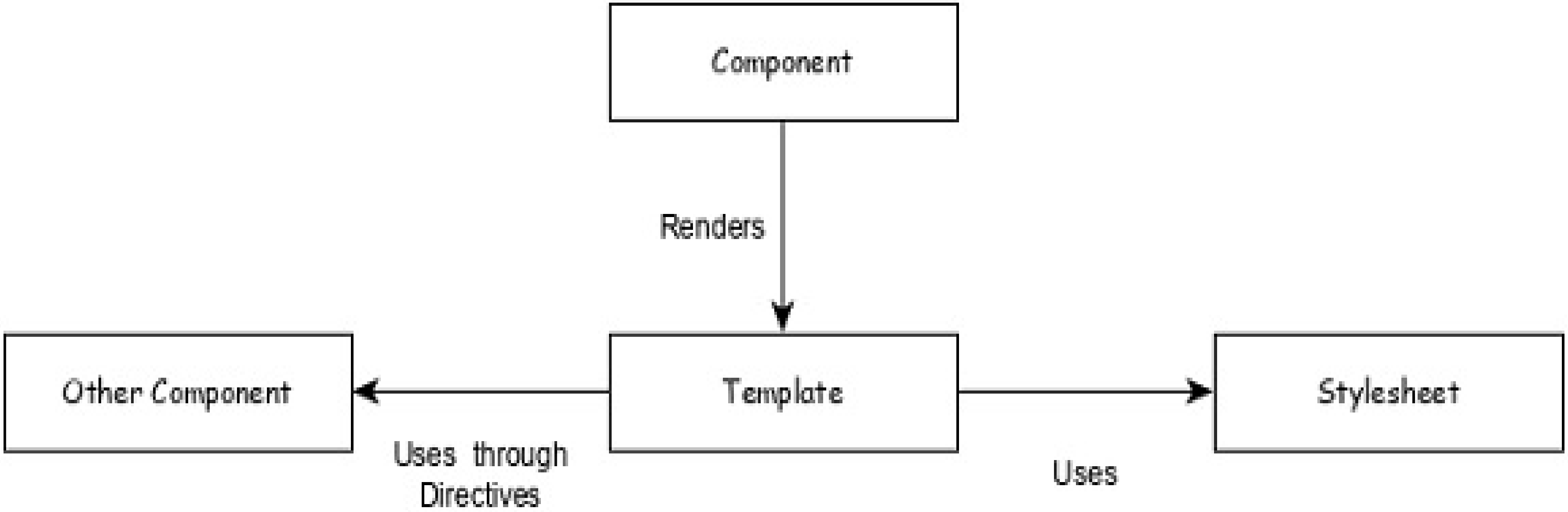
Angular components

- Components are the main building block for Angular applications. Each component consists of an HTML template that declares what renders on the page.
- A component is simply a TypeScript class that has been annotated with the @Component decorator supplied from the Angular core module in terms of code.
- It also supports an HTML template that can be used inline or written as a standalone HTML file to display content and associate properties and events from the component's class.

Angular components

- This is simply a regular TypeScript class with the @Component decorator added to it from the core module. The decorator holds component metadata, such as the following:
- selector: This is used to access the component from an HTML page in the same way that traditional HTML tags are. Angular may be used to create custom HTML tags that are exclusively available inside your application.
- templateUrl: This specifies where Angular should search for the HTML template containing the display. markup.
- styleUrls: This specifies where to obtain component-specific styles that define the component's feel and look.

Angular components



Services

- Services are plain Typescript / JavaScript class providing a very specific functionality.
- Services will do a single task and do it best.
- The main purpose of the service is reusability.
- Instead of writing a functionality inside a component, separating it into a service will make it useable in other component as well.

Dependency injection

- Dependency Injection is used to properly initialize the service in the component so that the component can access the services as and when necessary without any setup.