

## **Differences between Angular services in various versions and Components.**

### **Components**

Components are responsible for defining the UI (User Interface) and behavior of a part of the application. Each component encapsulates a part of the user interface along with its associated logic.

Components consist of three parts: the template (HTML), the component class (TypeScript), and the component decorator (metadata). These parts work together to define the component's functionality and appearance.

Components are reusable and can be used multiple times across the application. They promote modularization and encapsulation of functionality.

### **Services**

Services are used for encapsulating reusable logic or data that can be shared across multiple components or throughout the application. They promote the DRY (Don't Repeat Yourself) principle by centralizing common functionality.

Services are simple TypeScript classes with a specific purpose. They do not have templates or decorators like components.

Angular services are typically singleton instances, meaning there's only one instance of a service throughout the application. This ensures that data and state are shared consistently across components.

## **Differences between various versions of angular**

AngularJS (1.x):

This was the first version of Angular released by Google in 2010.

It introduced concepts like two-way data binding, directives, and dependency injection.

AngularJS used JavaScript for development.

Angular 2+ (also known as Angular 2, Angular 4, Angular 5, etc.):

Angular 2 was a complete rewrite of AngularJS and introduced a component-based architecture.

It used TypeScript as the primary language for development, providing benefits like static typing and better tooling.

Introduced features like modularization, improved performance, and enhanced dependency injection.

Angular 2 was followed by Angular 4, Angular 5, and so on. These were incremental updates that introduced improvements, bug fixes, and new features. The version numbering was aligned for core packages.

Angular (starting from version 6):

After Angular 5, the Angular team switched to semantic versioning (major.minor.patch) and dropped the 'JS' from the name.

Introduced Angular CLI (Command Line Interface) for easier project setup, scaffolding, and deployment.

Introduced Angular Elements, allowing Angular components to be used in non-Angular environments.

Angular (versions 10, 11, 12, etc.):

Continued focus on improving developer experience, performance, and stability.

Regular updates to Angular CLI, Angular Material, and other libraries.

Introduction of new features like stricter type checking, improved handling of dependency injection, and updates to the Angular Router.

Angular (latest):

Continues to evolve with regular updates and releases, addressing developer feedback and industry trends.

Focus on performance optimizations, developer productivity, and improving compatibility with modern web standards.