05 Angular Routing / Service / Form

Prepared by Surajai Chamroensat

Agenda

- Services
- Routing
- Observables and RxJS basics
- Form controls and validation
- Material Design

Services

Angular services provide a way for you to separate Angular app data and functions that can be used by multiple components in your app. To be used by multiple components, a service must be made injectable. Services that are injectable and used by a component become dependencies of that component. The component depends on those services and can't function without them.

Dependency Injection

Dependency Injection (DI) is a software design pattern that focuses on achieving Inversion of Control (IoC) by separating the creation of objects from their usage. Instead of an object creating its own dependencies, those dependencies are "injected" into the object by an external entity, often referred to as an "injector" or "container."

Dependency injection is the mechanism that manages the dependencies of an app's components and the services that other components can use.

Routing

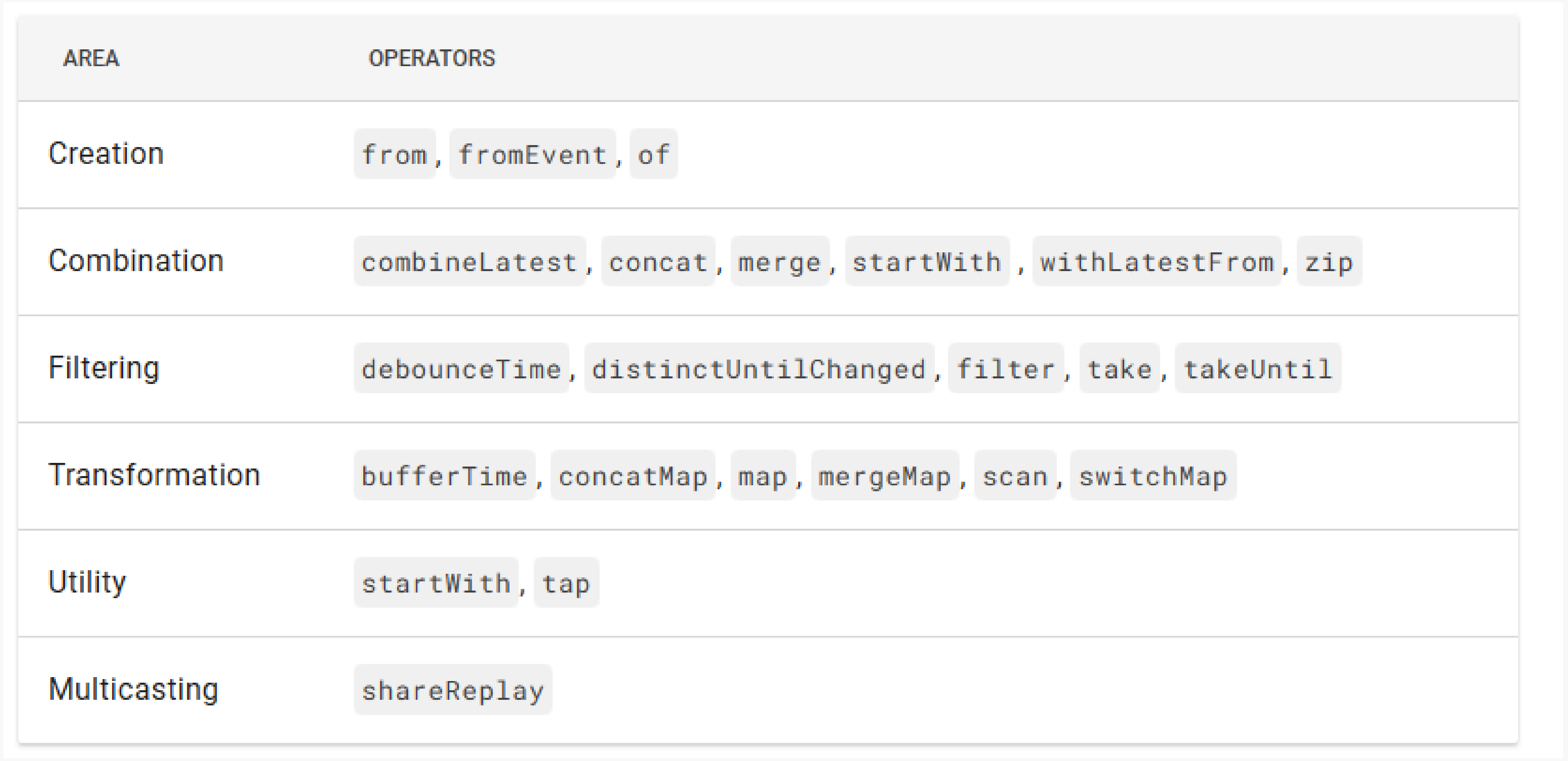
- Routing is the ability to navigate from one component in the application to another.

Observable and RxJS

- Observable
  - Angular makes use of observables as an interface to handle a variety of common asynchronous operations. For example:
    - The HTTP module uses observables to handle AJAX requests and responses
    - The Router and Forms modules use observables to listen for and respond to user-input events

Observable and RxJS

- RxJS



AREA	OPERATORS
Creation	`from`, `fromEvent`, `of`
Combination	`combineLatest`, `concat`, `merge`, `startWith` , `withLatestFrom`, `zip`
Filtering	`debounceTime`, `distinctUntilChanged`, `filter`, `take`, `takeUntil`
Transformation	`bufferTime`, `concatMap`, `map`, `mergeMap`, `scan`, `switchMap`
Utility	`startWith`, `tap`
Multicasting	`shareReplay`

Observable and RxJS

- toSignal
- toObservable

Form controls and validation

- formGroup
- validation

Material Design

- npm install -g @angular/cli
- ng add @angular/material
- Components
    - Button
    - Card
    - Autocomplete
    - Badge
    - Dialogue
    - Input
    - Table
    - Tab

Angular Bootstrap

- ng add @ng-bootstrap/ng-bootstrap
- Grid
- Alert
- Modal

References

- https://angular.dev/
- https://material.angular.dev/
- https://ng-bootstrap.github.io/#/home