@NgModule

This is a decorator used to help the compiler in identifying collections of components grouped together inside one entity(module). It helps the compiler identify the modules own components, any imports or exports, dependencies and so on.

Declarations

imports makes the exported declarations of other modules available in the current module.

declarations are to make directives (including components and pipes) from the current module available to other directives in the current module. Selectors of directives, components or pipes are only matched against the HTML if they are declared or imported.

providers are to make services and values known to DI (dependency injection). They are added to the root scope and they are injected to other services or directives that have them as dependency.

exports makes the components, directives, and pipes available in modules that add this module to imports. exports can also be used to re-export modules such as CommonModule and FormsModule, which is often done in shared modules.

EntryComponent

Entry Components registers components for offline compilation so that they can be used with ViewContainerRef.createComponent(). Components used in router configurations are added implicitly. They allow components to be used as modals or pop ups

Bootstrap

Bootstrap is a frontend components library that has hundreds of components that fully customizable to be used by developers in their frameworks to build responsive web apps

package.json vs package-lock json

The key difference between the files is that package.json defines the minimum requirement of the dependencies of the project. It also defines other details such as name, author, description, version, license etc. Package-lock.json on the other hand solely serves to identify the exact version of dependencies used in the project to allow for an exact replica of the project dependency tree to be built and the project to run on the same library versions.

Subscribers

Subscribers follow the same principles as Pub/Sub design patterns. A function can be declared as an observable and subscribers can subscribe to the result. The subscribers will have certain functions such as next, error and complete to take of any values being delivered by the publisher/observable.

Observables

In Observables, computation does not start until it has a subscriber. Observables can perform computations and emit any amount of values of different types which the subscriber can listen to.

RxJS library

RxJS stands for reactive extensions for javascript and the main use case is asynchronous programming. The above described concepts of observables and subscribers fall under the umbrella of rxjs. It is defined as a library for composing asynchronous and event-based programs by using observable sequences. It provides one core type, the Observable, satellite types (Observer, Schedulers, Subjects) and operators inspired by Array#extras (map, filter, reduce, every, etc.) to allow handling asynchronous events as collections.

Injectable

This decorator is used to define dependencies that are required to be compiled by a components constructor. Without this decorator, the compiler will not know how to gather the dependency injections for the constructor.

Normal http & Native http (difference)

Http is used to make requests over the internet in our apps. Normal HTTP is angular's trusted way of making requests, and is based on observables and is more secure. Native Http is based on cordova library and is used for special cases such as services which are not available in browsers.