diff between var and dynamic?

https://www.dotnettricks.com/learn/csharp/differences-between-object-var-and-dynamic-type

BrowserModule @angular/platform-browser When you want to run your app in a browser

componentModule

NgModule

Injectable

Directive

Pipe

CommonModule @angular/common When you want to use NgIf, NgFor

FormsModule @angular/forms When you want to build template driven forms (includes NgModel)

ReactiveFormsModule @angular/forms When you want to build reactive forms

RouterModule, Routes @angular/router When you want to use RouterLink, .forRoot(), and .forChild()

HttpClientModule @angular/common/http When you want to talk to a server

needed for reactive forms : import{FormGroup,FormControl,Validators,FormBuilder} from from '@angular/forms';

Routing:

Routing helps in directing users to different pages based on the option they choose on the main page. Hence,

based on the option they choose, the required Angular Component will be rendered to the user.

<router-outlet> </router-outlet>.--container for display route views, it is directive from routermodule

forRoot creates a module that contains all the directives, the given routes, and the router service itself.

forChild creates a module that contains all the directives and the given routes, but does not include the

router service. It registers the routers and uses the router service created at the root level.

Angular 2:

Angular 2 is components based

used for develop mobile driven apps

no longer controllers and $scope

Support for TypeScript

multimple browser support

Angular 4:

Faster and smalle

template is now ng-template

httpclient

useses routermodule- forroute, forchild for routing

Angular 5:

Router Life Cycle

compiler improvements

RxJs Support

Angular 6:

Angular Elements

RxJs 6 Support

Subjects are specific kind of observable and observer.

A subject is observable,so we can get values from the observable stream via subscribe i.e We can subscribe to them.

they support multiple subscriptions. In other words, they are multicast.but observables are singlecast.

A subject is observer,so we can update the observable stream with next(),error, complete method

but regular observable does not.

Behavior Subjects: When we subscribe to a behavior subject, it will give you the last emitted value right away.

Replay Subjects: Replay Subjects keep a given number of historical values so that those values can be replayed to new subscribers.

https://coursetro.com/posts/code/149/RxJS-Subjects-Tutorial---Subjects,-BehaviorSubject,-ReplaySubject-&-AsyncSubject

https://medium.com/@luukgruijs/understanding-rxjs-behaviorsubject-replaysubject-and-asyncsubject-8cc061f1cfc0