Angular creates components – e.g. each time it founds our selector it will instantiate new component and add it to the DOM. Then Angular goes for a couple additional phases (lifecycle hooks). We can add methods and if and only if they present Angular will call them in appropriate time. The order like in PDF and screenshot files.

Some comments on a few stages on top on basic provided by documentation.

ngOnInit() – will run only once while object creation after constructor – component won’t yet be injected in DOM and shown.

ngDoCheck() – will run a lot of times – each time **change detection run** (it doesn’t mean change had place but angular needs to check it) on the template of the component or more properly say inside the component – so it needs to change somehow template view. But important ngDoCheck will be executed for each check even no changes are detected for instance user clicked in field but didn’t change yet anything - ngDoCheck will run anyway to define whether check was made or not. Usually it is advanced use case to implement something for that method – like example you want to inform angular manually about some custom change was detected angular is not aware of. We should not do any heavy operations cause these hook called a lot and as result will lead to some performance issues.

ngOnDestroy() – e.g. if we removed component from the view for instance \*ngIf turns false and component should not be shown and removed from dom any more then onDestroy will be called on component – used usually to clear resources.