Project setup

Keep things organized in folders (Modules are not the default as of 2024). All pages, components, routes, validators, services that belong together should be have the same (grand)parent.

Project setup commands - sample

```
# When setting up a new project think about how
you want it to look. Make a short list of commands
to set up your project. Open the new project in
your IDE, if things do not feel right, adjust your
list and run it again. I have added an example
below.
ng new sample
cd sample
ng g c core/h ome pag e/c ont ain ers /ho mepage
ng g c core/p rod uct s/c ont ain ers /pr oducts
ng g c core/p rod uct s/c ont ain ers /pr odu ct-
edit
ng g c core/p rod uct s/c ont ain ers /pr odu ct-
edit
ng g c core/p rod uct s/c ont ain ers /pr odu ct-
add
ng g c core/p rod uct s/c omp one nts /pr odu ct-
form
ng g s core/p rod uct s/s erv ice s/p roduct
ng g c core/c ont act /co nta ine rs/ contact
# Make many more so you get a good feel of how your
decisions will impact the project. Make changes,
delete the project and run you commands again.
# Add a route to the Produc tsC omp onent at
/products
# Use the Produc tSe rvice in the constr uctor of
the Produc tsC omp onent
```

Life cycle hooks

ngOnCh-anges()	Respond when Angular (re)sets data-bound input properties. The method receives a SimpleChanges object of current and previous property values. Called before ngOnInit() and whenever one or more data-bound input properties change.
ngOnInit()	Initialize the directive/component after Angular first displays the data-bound properties and sets the directive/component's input properties. Called once, after the first ngOnChanges().

Life cycle hooks (cont)

ngDoCheck()	Detect and act upon changes that Angular can't or won't detect on its own. Called during every change detection run, immediately after ngOnChanges() and ngOnInit().
ngAfterContentInit()	Respond after Angular projects external content into the component's view / the view that a directive is in. Called once after the first ngDoCheck().
ngAfterViewInit()	Respond after Angular initializes the component's views and child views / the view that a directive is in. Called once after the first ngAfterContentChecked().
ngAfterViewChecked()	Respond after Angular checks the component's views and child views / the view that a directive is in. Called after the ngAfterViewInit() and every subsequent ngAfterContentChecked().
ngOnDestroy()	Cleanup just before Angular destroys the directive/component. Unsubscribe Observables and detach event handlers to avoid memory leaks. Called just before Angular destroys the directive/component.