Contents

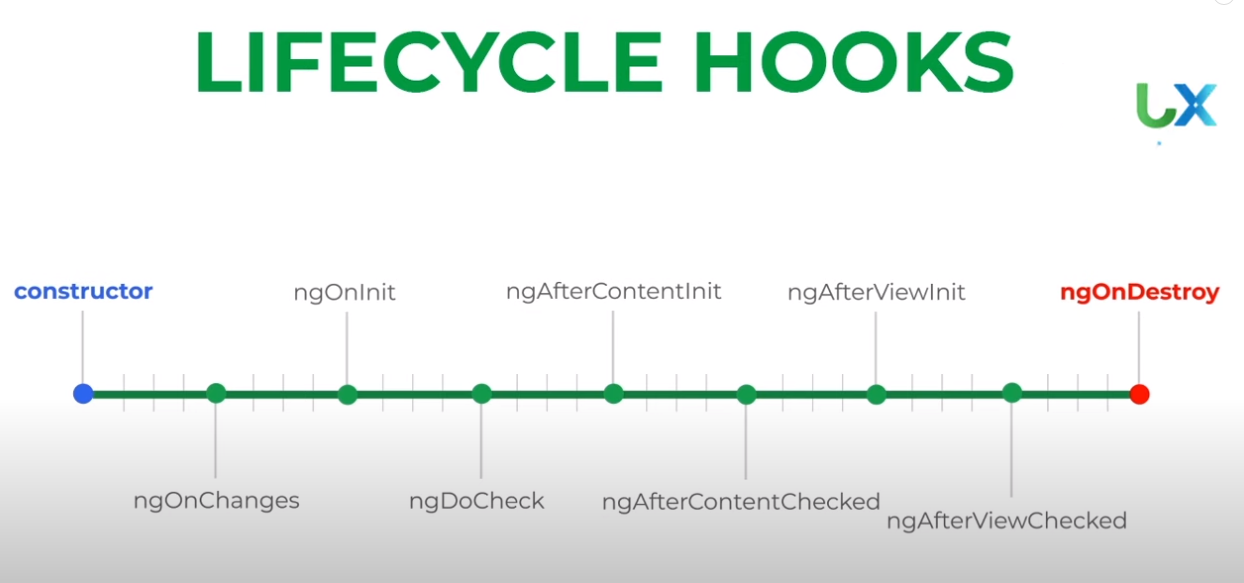
* Life cycle sequence
* Change Detection Strategy



**Life Cycle Sequence**

* **A component instance has a lifecycle that starts when the angular instantiates the component class and renders the component view along with its child views.**
* **Angular executes hook methods in the following sequence**
* **https://www.freecodecamp.org/news/angular-lifecycle-hooks/**
  1. **ngOnChanges()**
     + PURPOSE:
       - Respond when angular sets or resets data-bound input properties.
       - The method receives a **SimpleChanges** and **SimpleChange** object of current and previous values.
     + TIMING:
       - Called before **ngOnInit().**
  2. **ngOnInit()** 
     + PURPOSE:
       - Initialize the directives or component after angular first displays the data-bound properties and sets the directive or component’s input properties.
     + TIMING:
       - Called **once**, after the first **ngOnChanges().**
  3. **ngDoCheck()**
     + PURPOSE:
       - Detect and act upon changes that angular can’t or won’t detect on its own.
     + TIMING:
       - Called immediately after **ngOnChanges()** on every changes detection run, and immediately after **ngOnInit()** on the first run.
  4. **ngAfterContentInit()**
     + PURPOSE:
       - Respond after Angular projects external content into the component’s view, or into the view that a directive is in.
     + TIMING:
       - Called **once** after the first ngDoCheck().
  5. **ngAfterContentChecked()**
     + PURPOSE:
       - Respond after angular checks the content projected into the directive or component.
     + TIMING:
       - Called after **ngAfterContentInit()** and every Subsequent **ngDoCheck().**
  6. **ngAfterViewInit()**
     + PURPOSE:
       - Respond after angular initializes the component’s views and child views, or the view that contains the directive.
     + TIMING:
       - Called **once** after the first **ngAfterContentChecked()**
  7. **ngAfterViewChecked()**
     + PURPOSE:
       - Respond after angular checks the component’s view and child views, or the view that contains the directive.
     + TIMING:
       - Called after the **ngAfterViewInit()** and every subsequent **ngAfterContentChecked()**
  8. **ngOnDestroy()**
     + PURPOSE:
       - Cleanup just before angular destroys the directive or component.
       - Unsubscribe Observables and detach event handlers to avoid memory leaks.
     + TIMING:
       - Called immediately before angular destroys the directives or components.

https://angular.io/guide/cheatsheet



**Change Detection Strategy**

1. **onPush:**
   1. Use the **CheckOnce** strategy, meaning that automatic change detection is deactivated until reactivated by setting the strategy to Default **(CheckAlways).**
2. **Default:**
   1. Use the default **CheckAlways** strategy, in which change detection is automatic until explicitly deactivated.