



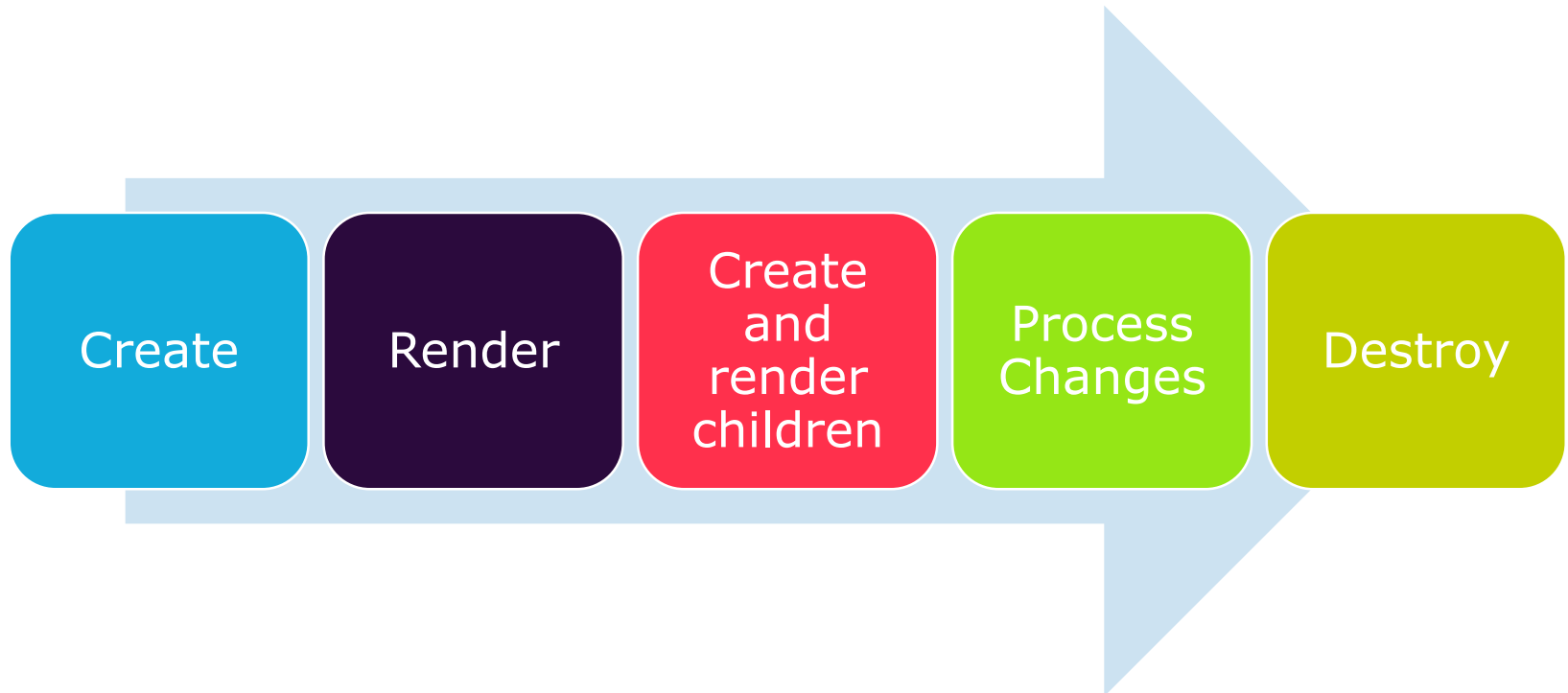
# Angular 4

Component Lifecycle



# Component Lifecycle

- A component has a lifecycle managed by angular.
- The constructor of the component class is called before any other component lifecycle hook.
- As a best practice inputs of a component should not be accessed via constructor. To access the value of an input for instance to load data from server component's life cycle phase should be used





## Lifecycle hooks

- Angular calls life cycle hook methods on directives and components as it creates, changes and destroy them

Creates	Changes	Destroy
<b>OnInit</b>	<b>DoCheck</b>	<b>OnDestroy</b>
<b>AfterContentInit</b>	<b>OnChanges</b>	
<b>AfterViewInit</b>	<b>AfterContentChecked</b>	
	<b>AfterViewChecked</b>	



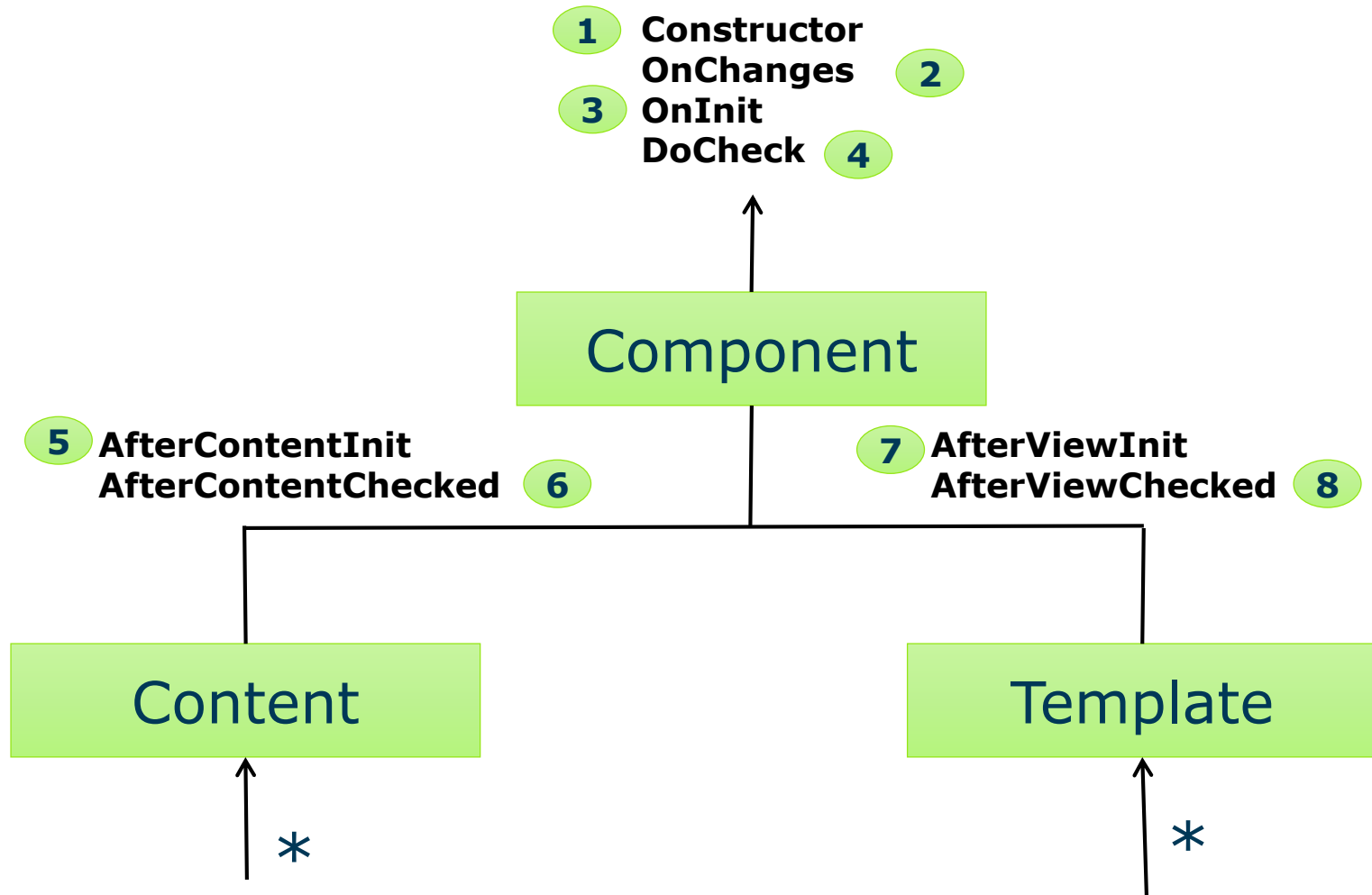
## Accessing Component Lifecycle hooks

- To respond to certain events in the life cycle of a component angular provides various interface which can be implemented.
- Angular application finds the hook functions only when the Component class assigns the associated interface.
- For each hook a corresponding interface exists without the prefix **ng**

Interface	Hook	Interface	Hook
<b>OnChanges</b>	<b>ngOnChanges</b>	<b>OnInit</b>	<b>ngOnInit</b>
<b>DoCheck</b>	<b>ngDoCheck</b>	<b>AfterContentInit</b>	<b>ngAfterContentInit</b>
<b>AfterContentChecked</b>	<b>ngAfterContentChecked</b>	<b>AfterViewInit</b>	<b>ngAfterViewInit</b>
<b>AfterViewChecked</b>	<b>ngAfterViewChecked</b>	<b>OnDestroy</b>	<b>ngOnDestroy</b>



# Lifecycle hooks execution order



# Demo



## ComponentLifecycle