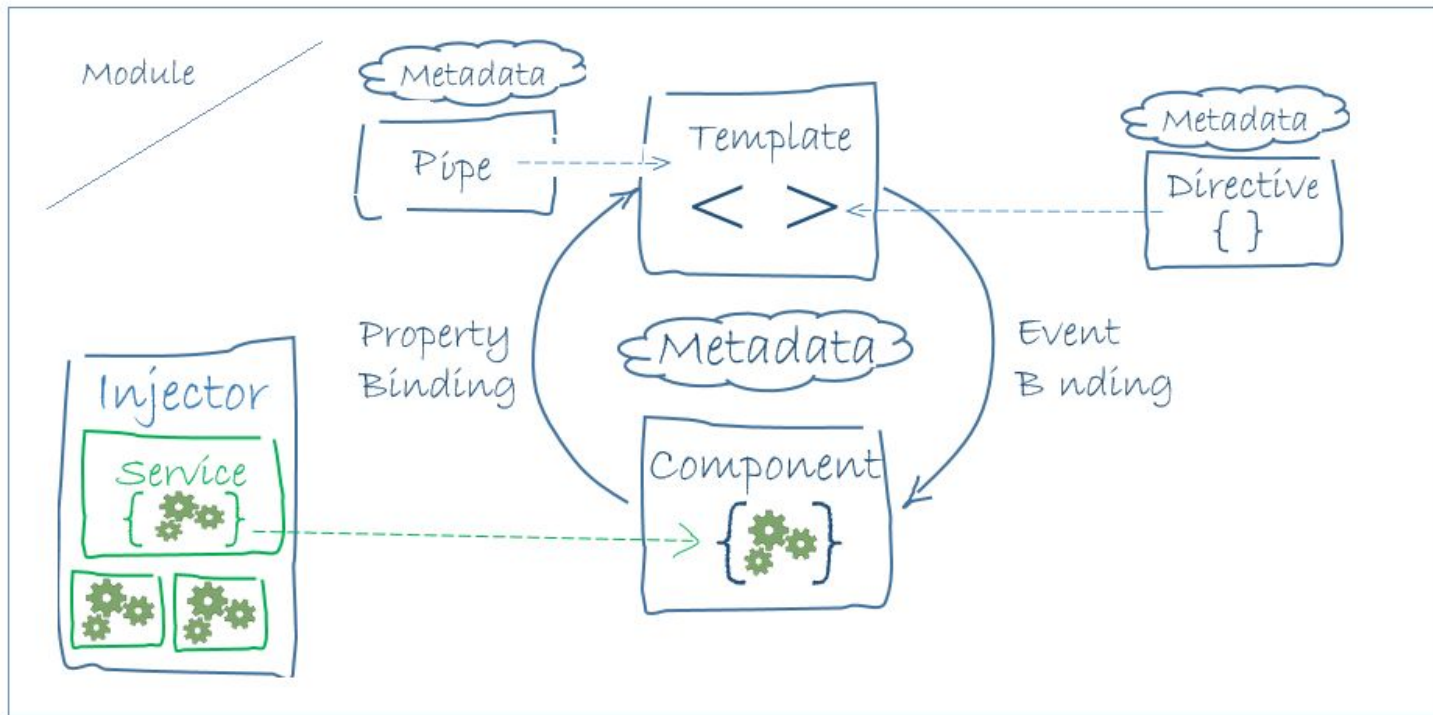


# Lesson 3

---

Directives, Pipes, Lifecycle hooks

# Structure



# Directives

- Directive - class marked with decorator `@Directive`
- Allow to attach behaviour or rearrange content

# Directives

- Relation:

`@Component = @Directive + template`

- Structural directives
- Attribute directives

# Structural Directives

- change internal DOM

```
<li *ngFor="let user of users">{{ user.Name }}</li>
```

```
<user-details *ngIf="user">...</user-details>
```

# Attribute Directives

- Change appearance or behavior

```
<div [ngStyle]="{'font-size' : isSpecial ? '24px' : '12px'}">...</div>
```

```
<div [ngClass]="{'class1': isClickable, 'class2': !isClickable}">...</div>
```

# Pipes

- Format data in templates
- Built-in pipes
  - DatePipe
  - UpperCasePipe
  - LowerCasePipe
  - CurrencyPipe
  - PercentPipe

```
<div class="date">{{ today.Date | date }}</div>
```

```
<div class="header">{{ header | uppercase }}</div>
```

# Pipes

- Usage example

```
<div class="date">{{ today.Date | date: 'shortDate' }}</div>
```

- Chaining

```
<div class="date">{{ today.Date | date: 'shortDate' | uppercase }}</div>
```



# Lifecycle hooks

constructor

**ngOnChanges**

**ngOnInit**

**ngDoCheck**

**ngAfterContentInit**

**ngAfterContentChecked**

**ngAfterViewInit**

**ngAfterViewChecked**

**ngOnDestroy**

Angular calls lifecycle hook methods on directives and components as it creates, changes, and destroys them.

# Lifecycle sequence

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

# Lifecycle sequence

<code>ngDoCheck</code>	<p>Detect and act upon changes that Angular can't or won't detect on its own.</p> <p>Called during every change detection run, immediately after <code>ngOnChanges</code> and <code>ngOnInit</code>.</p>
------------------------	--

# Lifecycle sequence

<code>ngAfterContentInit</code>	<p>Respond after Angular projects external content into the component's view.</p> <p>Called once after the first <code>NgDoCheck</code>.</p> <p>A component-only hook.</p>
<code>ngAfterContentChecked</code>	<p>Respond after Angular checks the content projected into the component.</p> <p>Called after the <code>ngAfterContentInit</code> and every subsequent <code>NgDoCheck</code>.</p> <p><i>A component-only hook.</i></p>

# Lifecycle sequence

ngAfterViewInit	<p>Respond after Angular initializes the component's views and child views.</p> <p>Called <i>once</i> after the first <code>ngAfterContentChecked</code>.</p> <p><i>A component-only hook.</i></p>
ngAfterViewChecked	<p>Respond after Angular checks the component's views and child views.</p> <p>Called after the <code>ngAfterViewInit</code> and every subsequent <code>ngAfterContentChecked</code>.</p> <p><i>A component-only hook.</i></p>

# Lifecycle sequence

ngOnDestroy	<p>Cleanup just before Angular destroys the directive/component. Unsubscribe observables and detach event handlers to avoid memory leaks.</p> <p>Called <i>just before</i> Angular destroys the directive/component.</p>
-------------	--

Example: <https://angular.io/generated/live-examples/lifecycle-hooks/eplnkr.html>