Understanding the Building Blocks

David Mann

@MannD | www.HeirloomSoftware.com

Topics

Component

- @Input
- @Output

Directives

Pipes

Building Blocks

Components

```
@Component({
  selector: 'my-component',
  templateUrl: './my.component.html',
  styleUrls: ['./my.component.scss']
})
```

Directives

```
<div myDirective>{{content}}</div>
```

Pipes

```
{{contents | myPipe}}
```

Angular Basics

Angular: Getting Started

Deborah Kurata

Angular Fundamentals

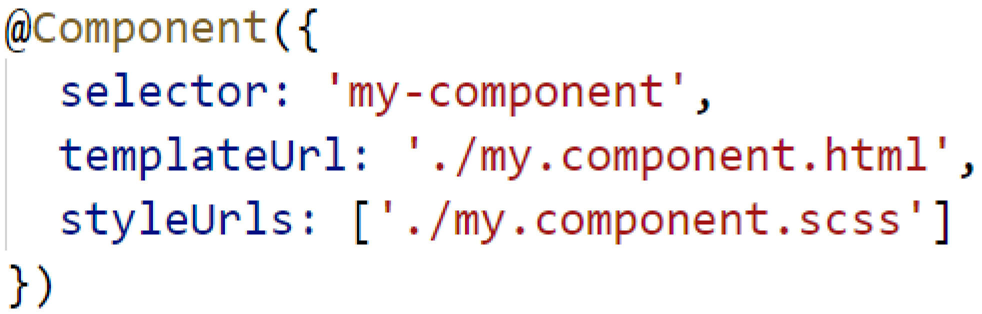
Joe Eames

Jim Cooper

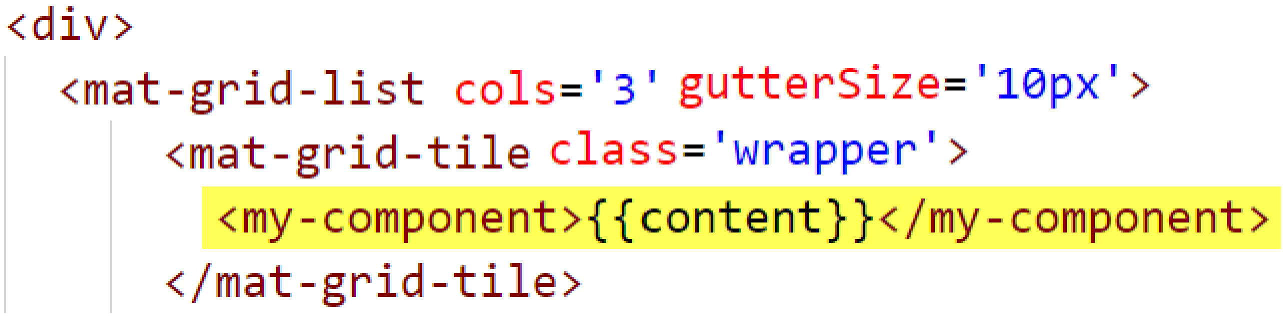
Angular: First Look

John Papa

Components



```
@Component({
  selector: 'my-component',
  templateUrl: './my.component.html',
  styleUrls: ['./my.component.scss']
})
```



```
<div>
  <mat-grid-list cols='3' gutterSize='10px'>
    <mat-grid-tile class='wrapper'>
      <my-component>{{content}}</my-component>
    </mat-grid-tile>
```

Components

Modular

Atomic

Self-contained

Component Boundaries

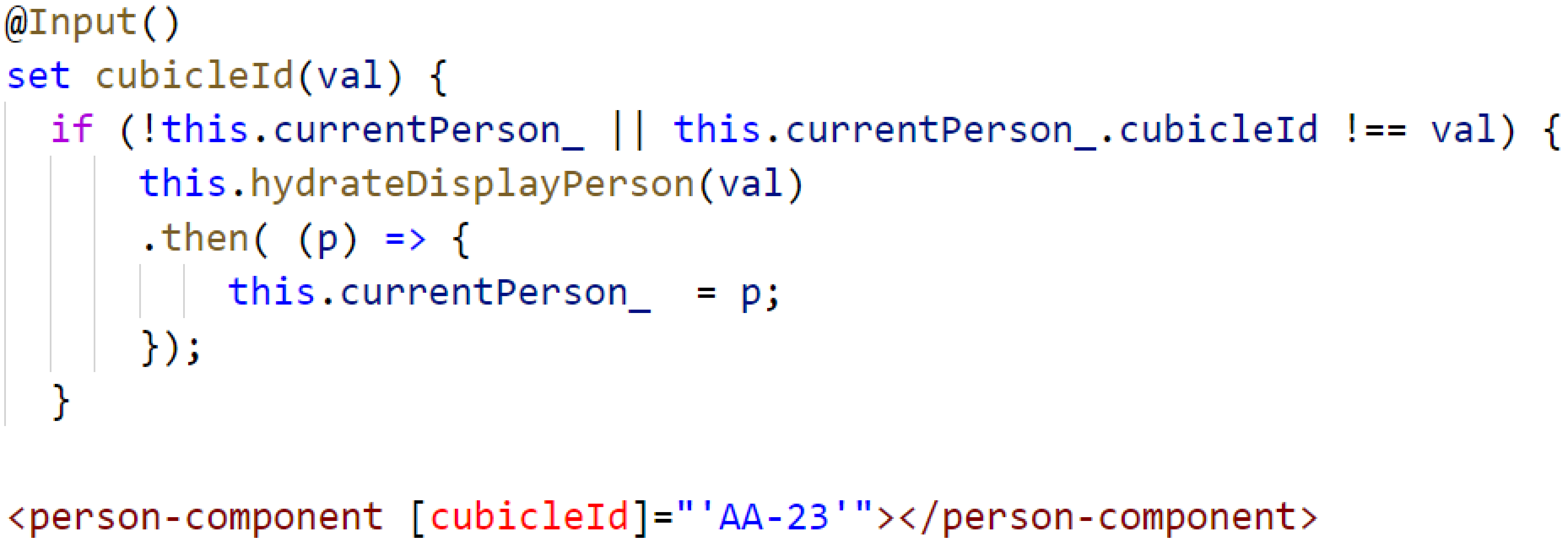
- ✓ Part of the same requirements change
- ✓ Used elsewhere
- ✓ Visually related
- ✓ Routing target
- ✓ Change together

@Input Decorator

```
@Input() cubicleId: string;
```

```
<person-component [cubicleId]="'AA-23'"></person-component>
```

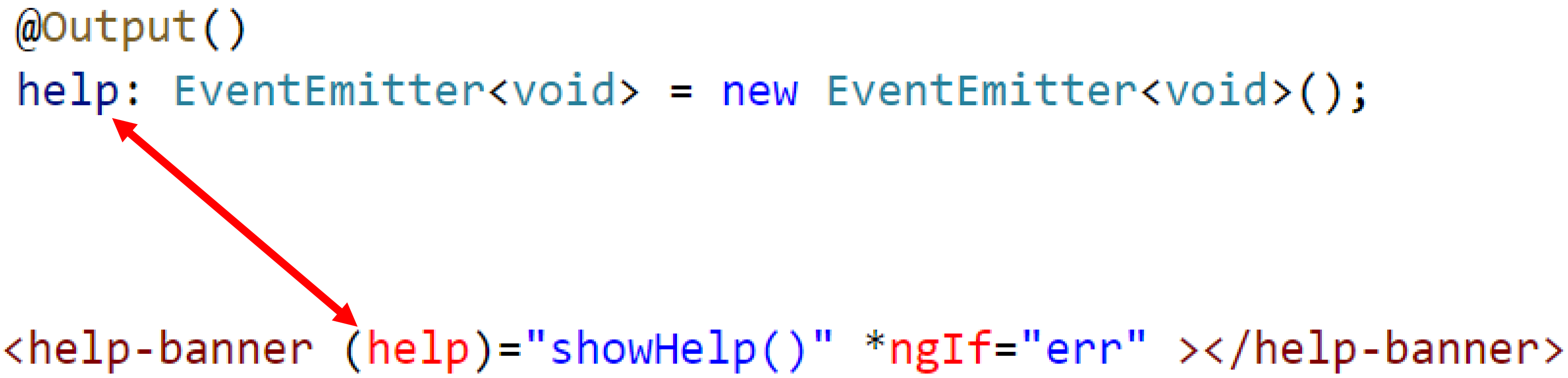
@Input Decorator



```
@Input()
set cubicleId(val) {
  if (!this.currentPerson_ || this.currentPerson_.cubicleId !== val) {
      this.hydrateDisplayPerson(val)
      .then( (p) => {
          this.currentPerson_  = p;
      });
  }

<person-component [cubicleId]="'AA-23'"></person-component>
```

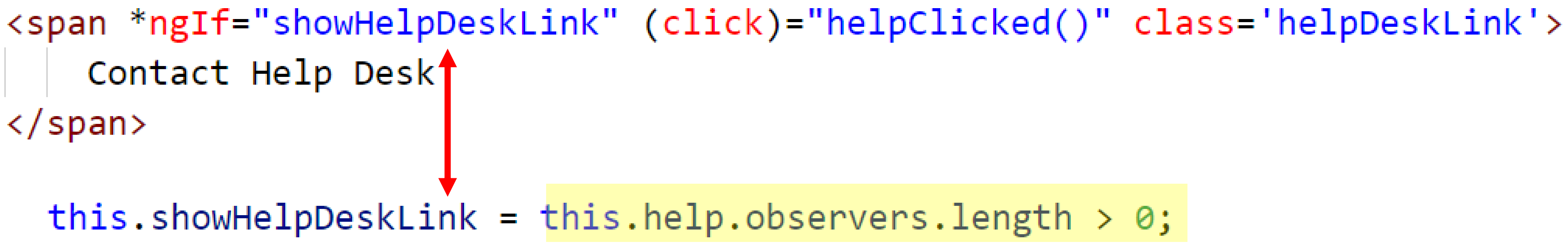
@Output Decorator



```
@Output()
help: EventEmitter<void> = new EventEmitter<void>();
```

```
<help-banner (help)="showHelp()" *ngIf="err" ></help-banner>
```

@Output Decorator



```
<span *ngIf="showHelpDeskLink" (click)="helpClicked()" class='helpDeskLink'>
    Contact Help Desk
</span>

  this.showHelpDeskLink = this.help.observers.length > 0;
```

Types of Directives

~~Component~~

Attribute Directive

~~Structural Directive~~

An Attribute directive changes the appearance or behavior of a DOM element

Source: https://angular.io/guide/attribute-directives

Why Directives?

Dynamic HTML

Responding to run-time environments

Quick & easy reuse

Pipes modify data for display (only)

Stable

Async Unwraps a value from an asynchronous primitive

Currency Formats a number as currency using locale rules

Date Formats a date according to locale rules

Decimal Formats a number according to locale rules

JSON Converts JSON object into JSON string

LowerCase Transforms text to lowercase

Percent Formats a number as a percentage according to locale rules

Slice Creates a new list or string containing a subset of the elements

TitleCase Transforms text to title case

UpperCase Transforms text to uppercase

https://angular.io/api/common/*Name*Pipe

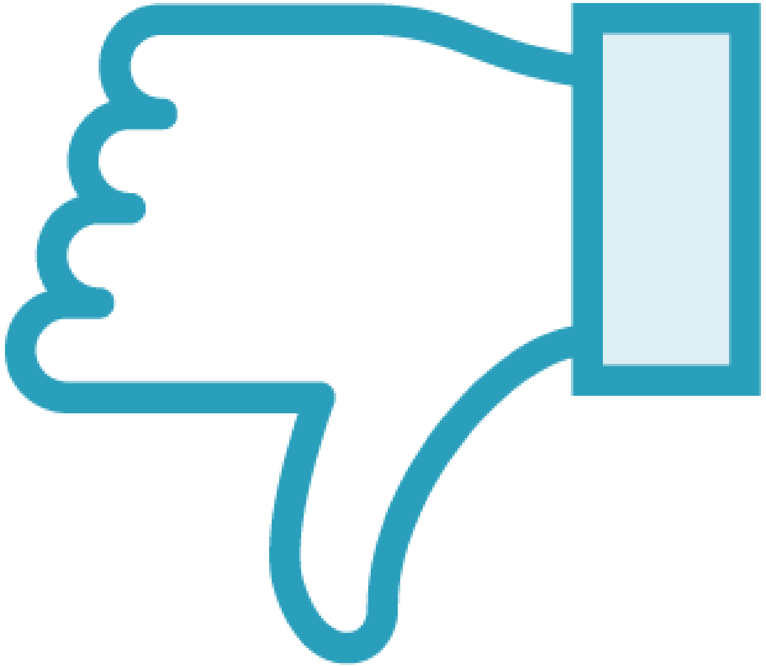
Experimental

I18nPlural Maps a value to a string that pluralizes the value according to locale rules

I18nSelect Generic selector that displays the string that matches the current value

https://angular.io/api/common/*Name*Pipe

(*Soon to Be*) Deprecated?



DeprecatedCurrency Formats a number as currency using locale rules

DeprecatedDate Formats a date according to locale rules

DeprecatedDecimal Formats a number according to locale rules

DeprecatedPercent Formats a number as a percentage according to locale rules

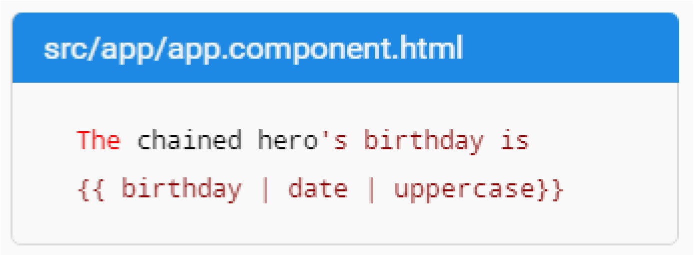
All require Internationalization API:

- Not in all browsers
- May require polyfill

https://angular.io/api/common/*Name*Pipe

Things to Know

Chaining

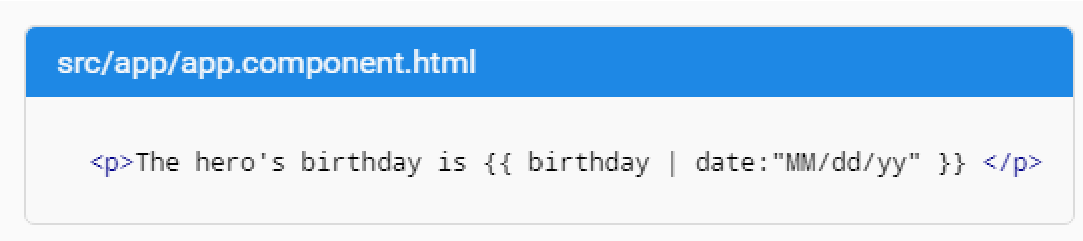


src/app/app.component.html

```
The chained hero's birthday is
{{ birthday | date | uppercase}}
```

FRIDAY, APRIL 15, 1988

Parameters



src/app/app.component.html

```
<p>The hero's birthday is {{ birthday | date:"MM/dd/yy" }} </p>
```

04/15/88

Performance

Pure vs. Impure

Source: https://angular.io/guide/pipes

Key Takeaways

What makes a good Component?

@Input & @Output

Directives

Pipes

- Chaining, Parameters, Performance

Next Up

Querying the UI