

# Logols Learning

WEEKEND WEB DEVELOPMENT BOOT CAMP

TRAINING: ANGULAR

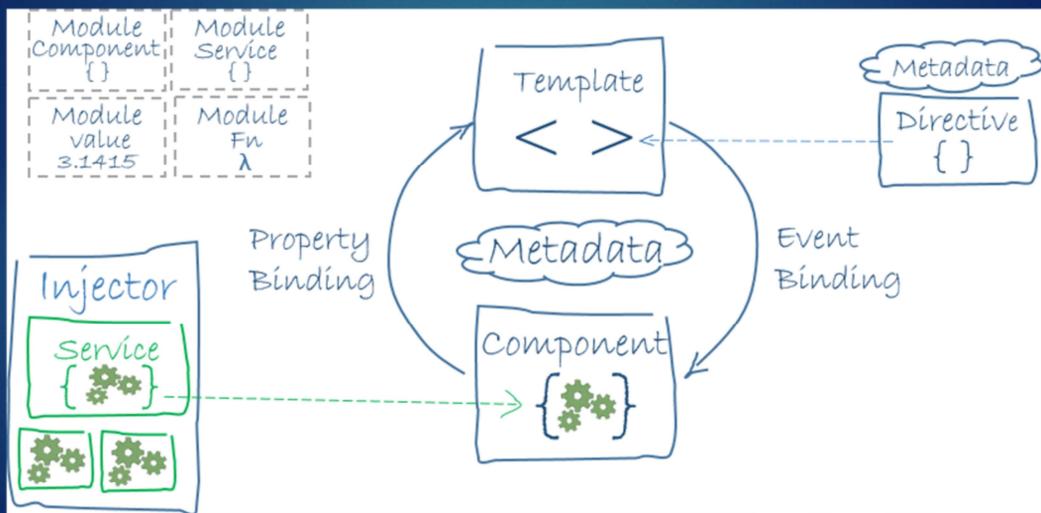
# Architecture of Angular?

- ▶ Modules
- ▶ Component
- ▶ Templates
- ▶ Directives
- ▶ Services



- These are all parts of what make up Angular.
- Angular is built with TypeScript
- It has modules, components, templates, metadata, and services.
- We are going to go through each of these.

# Architecture of Angular



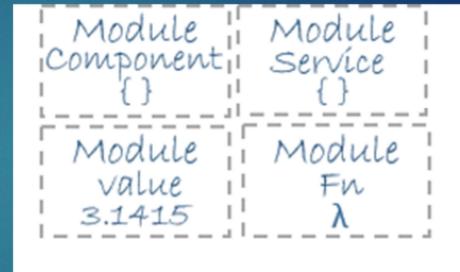
# Angular CLI

- ▶ Install Angular CLI
  - ▶ `npm install -g @angular/cli`
- ▶ Create a new Angular App
  - ▶ `ng new [app-name]`
- ▶ Change Directory
  - ▶ `cd [app-name]`
- ▶ Run the Application
  - ▶ `ng serve`

- The Angular CLI is the command line interface for Angular.
- To install the Angular CLI enter the following command: `npm install -g @angular/cli`
- To create a new application enter the following command: `ng new [app-name]`
- To change the directory to the new application enter the following command: `cd [app-name]`
- To run the application enter the following command: `ng serve`

# Modules

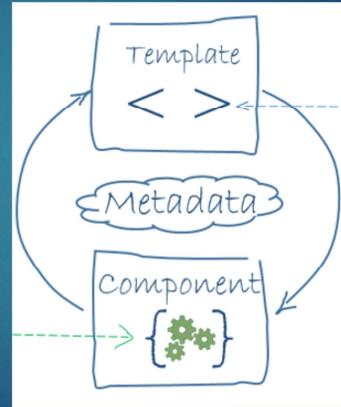
- ▶ Define the Components
- ▶ Root Module
- ▶ Feature Modules



- Modules define the components that live within them.
- There is a root module and there can also be feature modules to break down and separate the code.

# Components

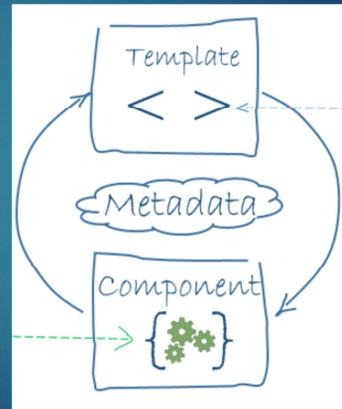
- ▶ Defines the Template
- ▶ Defines the Class
- ▶ Adds Metadata as needed
- ▶ Uses Export and Import keywords to export and import modules



- Components define the template or location of the template.
- They also define the class and adds any metadata needed.
- The Export keyword is used to export a module.
- The Import keyword is used to import an exported module.

# Templates

- ▶ Defines the html
- ▶ Can be inline of the component
- ▶ Or in a separate html file



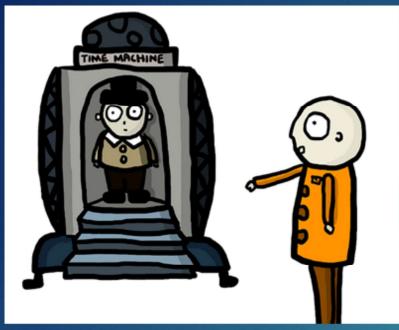
- The template defines the html
- The template can be fully written in the component.
- It is better practice to write the html separately and reference it in the component.

## Assets Folder

- ▶ Contains all assets for application.
- ▶ Non Angular assets
- ▶ Images
- ▶ CSS Files not tied to component
- ▶ Reference files as:
- ▶ /assets/.../img.jpg



- The template defines the html
- The template can be fully written in the component.
- It is better practice to write the html separately and reference it in the component.



## EXAMPLE

ANGULAR COMPONENTS AND TEMPLATES

# ASSESSMENT

ANGULAR COMPONENTS AND TEMPLATES



- Write on the board the command to create a new angular application.
- What are 5 main pieces of the Angular architecture?
- Which piece does html go in?
- Which piece defines the class and links to the template?
- Write on the board a component definition.

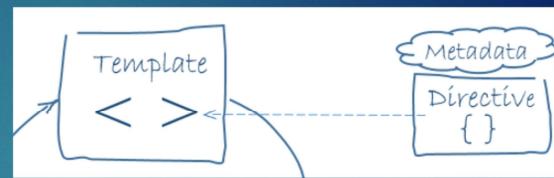
## Assignment

- ▶ Add the status report into Angular.
- ▶ Create a new component and html template for your zombie report.
- ▶ Add the report html that you created before into the html template.



# Directives

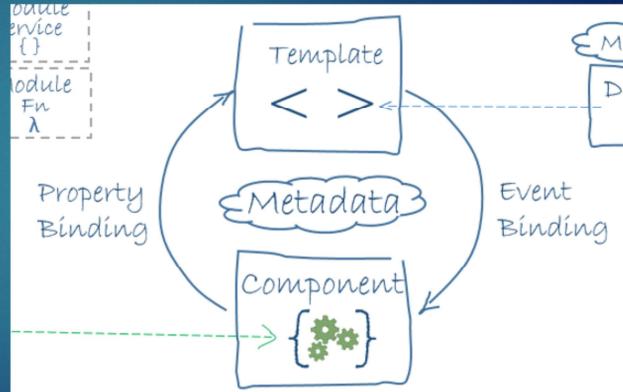
- ▶ Custom extended HTML element
- ▶ ngIf – conditional elements
- ▶ ngFor – elements created in a loop
- ▶ ngClass – conditionally add classes



- Directives are custom extended html elements
- They are for things like conditions using ngIf and loops using ngFor

# Data Binding

- ▶ Interpolation
- ▶ One-way binding (unidirectional)
- ▶ Two-way binding
- ▶ Event binding



- There are different types of data binding.
- Let's look at the examples to understand these better.
- Examples: <http://www.c-sharpcorner.com/article/data-binding-in-angular-2/>

# Data Binding Examples

- ▶ Interpolation

```
Hello {{ name }}
```

- ▶ One-way binding

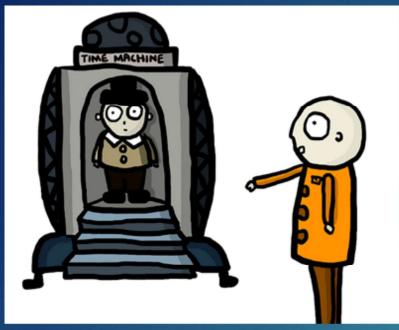
```
<input type = 'text' [value] = "firstName" />
```

- ▶ Two-way binding

```
<input [(ngModel)] = "firstName" />
```

- ▶ Event binding

```
<button (click) = "onSaveClick()" > Save </button>
```

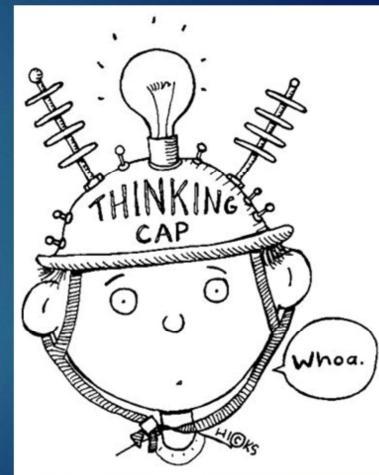


## EXAMPLE

ANGULAR BINDING

# ASSESSMENT

ANGULAR BINDING



- Which Angular piece extends HTML and makes it dynamic?
- What are the four types of binding?
- Write on the board interpolation for a property called name.
- Write on the board a text field with a two way binding to a property called name.
- Write on the board a button with an event binding to an event called onButtonClick.

# Assignment

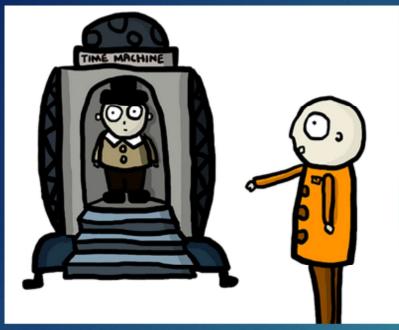
- ▶ Add a property into the component for the title.
- ▶ Use interpolation to show the value of the property for the title.
- ▶ Create a class that defines a status report line named PersonStatus with properties: firstName, lastName, statusId, statusDescription.
- ▶ Create a property with an array of PersonStatus objects.
- ▶ Use the ng-for directive to display array data in the table.



# Routing/Navigation

- ▶ Define routes in the module file
- ▶ Use the following syntax in the template to link to a route:
  - ▶ `<a [routerLink] = "[route path]">Description</a>`
- ▶ Use router-outlet element to specify location of the route in the template
- ▶ Use \* to route all undefined paths

- Routes allow for navigation through the website.
- Routes are setup in the module file and specify the path for a specific router link.
- \* can be used to route all undefined paths.
- Example: [https://www.tutorialspoint.com/angular2/angular2\\_routing.htm](https://www.tutorialspoint.com/angular2/angular2_routing.htm)

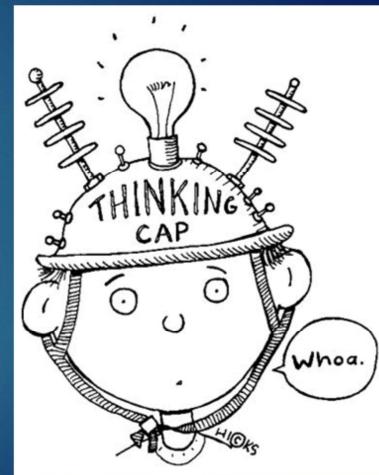


## EXAMPLE

ANGULAR ROUTING

# ASSESSMENT

ANGULAR ROUTING



- Write on the board a route definition for a route called home.
- Write on the board a link that uses a route called home.

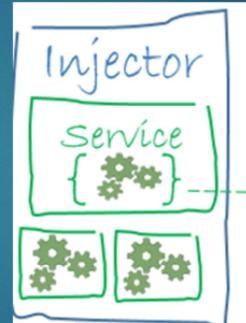
## Assignment

- ▶ Create a new component and html template.
- ▶ Add html for the form into it that you created previously.
- ▶ Setup up routing and a link so that the add link goes to this new component.



# Service

- ▶ Separate Class
- ▶ Used to interact with the data
- ▶ Read and Persist
- ▶ Called from the component



- A service class is usually used to interact with the data.
- It provides the reading and persisting the data usually calling services from the server.
- These are hooked up through the component.

# CORS

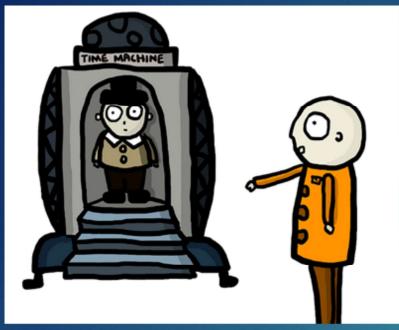
- ▶ Cross-Origin Resource Sharing
- ▶ By default Web API does not allow cross domain requests.
- ▶ Modify Startup.cs

- ▶ ConfigureServices Method

```
services.AddCors(o =>
o.AddPolicy("AllowSpecificOrigin", builder =>
{
    builder.AllowAnyOrigin()
        .AllowAnyMethod()
        .AllowAnyHeader();
}));
```

- ▶ Configure Method

```
app.UseCors("AllowSpecificOrigin");
```



## EXAMPLE

ANGULAR SERVICE

## Assignment

- ▶ Create a service that calls into your previously created web api.
- ▶ Return data and loop through it using the ng-for directive for the table instead of using the hard coded array.



# QUICK REVIEW

ANGULAR



- Write on the board the command to create a new angular application.
- Which piece does html go in?
- Write on the board a div that uses ngif based on a property named headerVisible.
- Write on the board a component definition.
- What are the four types of binding?
- Write on the board interpolation for a property called name.
- Write on the board a text field with a two way binding to a property called name.
- Write on the board a button with an event binding to an event called onButtonClick.
- Write on the board a route definition for a route called home.
- Write on the board a link that uses a route called home.
- Which Angular piece would be used to read data from an API?

## Additional Resources

- ▶ Cheat Sheet
  - ▶ <https://angular.io/guide/cheatsheet>
- ▶ Angular Tutorial
  - ▶ <https://angular.io/tutorial>
- ▶ Udemy
  - ▶ <https://www.udemy.com/learn-angular-from-scratch/>
- ▶ JSFiddle Template
  - ▶ <https://jsfiddle.net/yurzui/1hk7knwq/>