Angular 2

S1ngS1ng

Front End

Presentation Layer

- UI (User Interface)
  - User <==> Data access layer

Architecture

- Conventional VS SPA

Angular 1

Framework!
Framework!
Framework!

- MVVM (MV*)
- Modulization
- Two-way data binding
- Sementic tag
- Dependency injection
- Test friendly

Angular 1 VS Angular 2

Angular 1

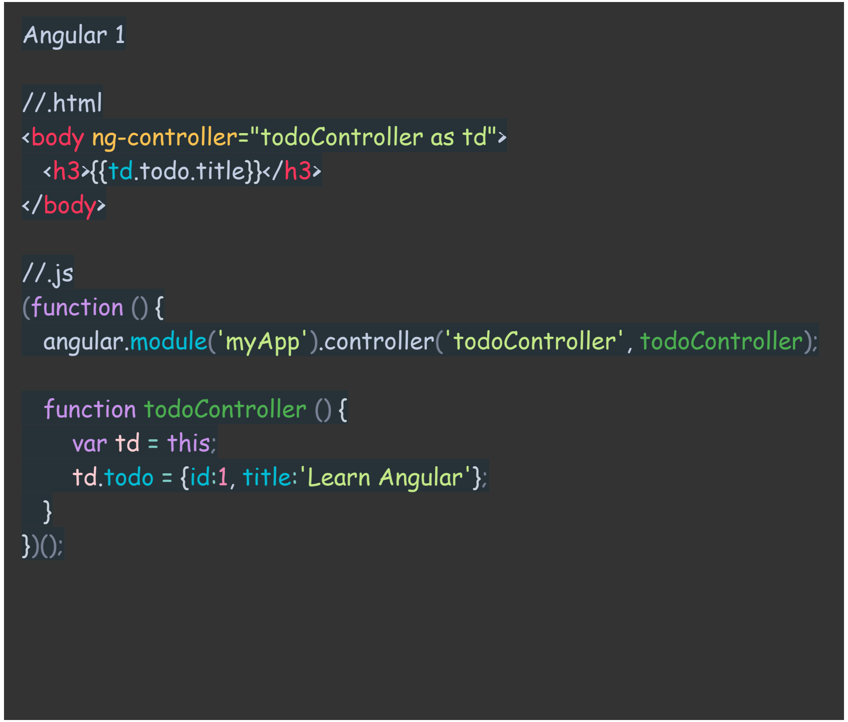
- Directives
- Controllers
- Routing
- $http
- $filter
- .factory, .service
- JavaScript

Angular 2

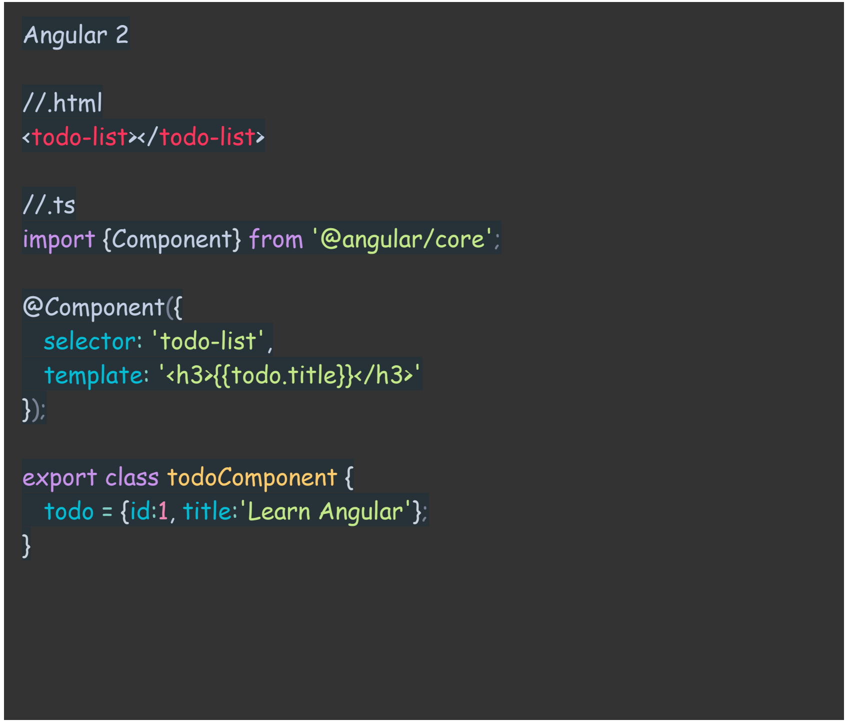
- Less Directives
- Components
- Routing
- HTTP
- Pipes
- Services
- TypeScript

Directives

Diff with Angular1 - Basic

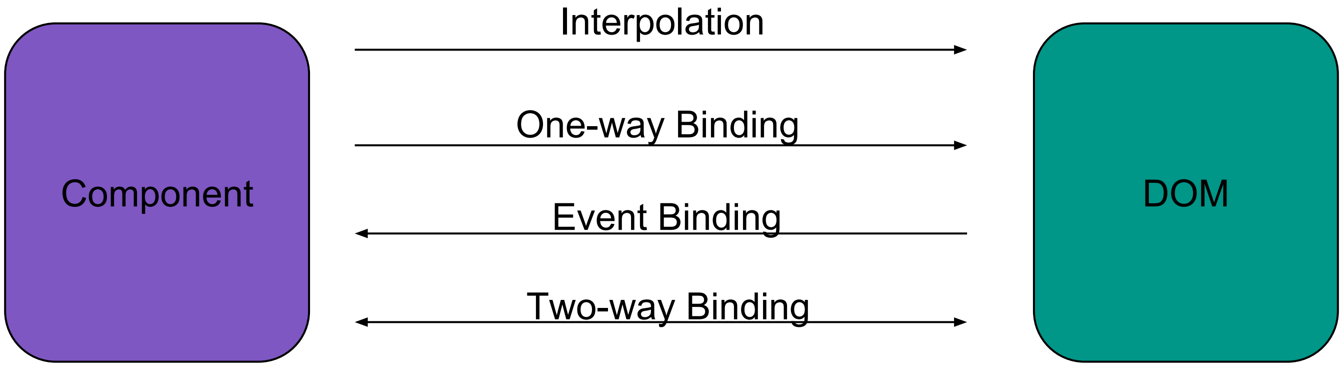


```
Angular 1

//.html
<body ng-controller="todoController as td">
  <h3>{{td.todo.title}}</h3>
</body>

//.js
(function () {
  angular.module('myApp').controller('todoController', todoController);

  function todoController () {
     var td = this;
     td.todo = {id:1, title:'Learn Angular'};
  }
})();
```



```
Angular 2

//.html
<todo-list></todo-list>

//.ts
import {Component} from '@angular/core';

@Component({
  selector: 'todo-list',
  template: '<h3>{{todo.title}}</h3>'
});

export class todoComponent {
  todo = {id:1, title:'Learn Angular'};
}
```

Understand Data-binding



Component

Interpolation

One-way Binding

Event Binding

Two-way Binding

DOM

Diff with Angular 1 - Directives

```
Angular 1

<ul>
  <li ng-repeat="todo in td.todos | limitTo:2:0">
     {{todo.title}}
  </li>
</ul>


<div ng-if="td.todos.length">
  You have {{td.todos.length}} todos :)
</div>
```

```
Angualr 2

<ul>
  <li *ngFor="let todo of todos | splice:0:2">
     {{todo.title}}
  </li>
</ul>


<div *ngIf="todos.length">
  You have {{todos.length}} todos :)
</div>
```

Diff with Angular 1 - Built-in directives

Angular 1	Angular 2
ng-app, ng-controller	bootstrap(), @Component()
ng-class, ng-href, ng-switch	[ng-class], [href],[ng-switch]
ng-click, ng-focus, ng-blur, ng-keyup	(click), (focus), (blur), (keyup)
ng-repeat, ng-if, ng-switch-when	*ngFor, *ngIf, *ngSwitchWhen
ng-model = “name”	[(name)]

Languages: Angular 1 vs Angualr 2

Angular 1

- JavaScript
- (Dart, CoffeeScript)

Angular 2

- TypeScript
- (JavaScript, Dart)

TypeScript

TS Code

```
import { cat } from 'animal';

export class Cat {

    constructor() {}

    catMeow() {

        return '=^o^=';

    }

}
```

What is TS?

- Superset of JS
  - JS(ES5) < ES6 < TypeScript
- Static Language
  - Compile to JS
- Strongly Typed
  - Optional type declaration
- Features ES6
  - Arrow Function

JS: the Good and Bad

Good

- Compatibility
- Libraries +++
- EZ to start
- Debug tool

Bad

- Dynamic typing
- No Modularity
- Verbose, IIFE
- Debug

EZ TS

- ES6 + Extra Features (optional)
- Free Tools!

Types

Strongly Typed

- any (object)
- void
- boolean
- number
- string
- type[]

Arrow function

From: ES6

```
(arg) => arg.toUpperCase();


function (arg) {

        return arg.toUpperCase();

}
```

Class

```
export class TodoList {

    constructor (todoService) {

        return todoService.get();

    }

}
```

Decorators

Syntatic Sugar

```
@Component({
    selector: ‘todoApp’,
    templateUrl: ‘./todo.html’,
    style: `
        .todo {color: purple}
        `
})
```

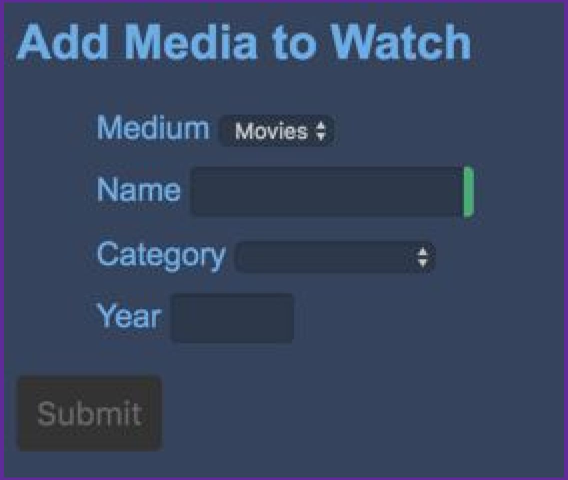
Import

```
import { Component } from
‘@angular/core’;

import { TodoList} from ‘./todoList.
component’;
```

Back to Angular 2

Form

What is form



Add Media to Watch

Medium Movies

Name

Category

Year

Form

- Template-Driven
- Half-half
- Model-Driven

Template-driven

Easy, straight-forward

HTML

- Input Elements
- Data-binding
- Error Message

TS

- Action

Component-driven

Complex, hard to understand

In .html

- Input elemnts
- Control binding

In .ts

- Life cycle
- Validation
- Error messages

Half-half

Intermediate, easy to understand

In .html

- Input elements
- Control binding
- Error messages

In .ts

- Controls
- Form
- Validation

Services

Services

Example

- Export class as service
  - @Injectable()
- Inject service into component
  - constructor() or @Inject
- Provider: container of services
- Recommend single initiation
  - Add to boostrap

Diff with Angular 1

Angular 1

- Factories
- Services
- Providers
- Constants
- Values

Angular 2

- Class

Router, HTTP

Built in

Router

- ROUTER_PROVIDERS, ROUTER_DIRECTIVES, RouterConfig
- <router-outlet>, [routerLink]

HTTP

- HTTP_PROVIDER, Http
- .map(), .json() .subscribe()

Eco-system

Tools

All Free :)

- Scaffold
- Code
- Test
- CSS Framework
- Deploy

Scaffold

Basic structure, boilerplate

- Yeoman
- Angular-seed

Code

Have fun coding :)

- Webstorm
- Visual Studio Code (recommend)
- Sublime Text
- Atom

Package Control

Dependency management

- NPM
- Bower

GRUNT

Gulp

Task Runner

Run ‘em all

npm scripts

- Grunt
- Gulp
- NPM Script

Jasmine

KARMA

Test

Framework and runner

Protractor
end to end testing for AngularJS

- Jasmine
- Karma
- Protractor (Selenium Server, webdriver-manager)

CSS Framework

EZ way to build stunning page

- Bootstrap
  - Except IE8 or Opera Android and iOS
- Angular Material (alpha)
  - n-1
- Materialize
  - Chrome 35+, FF 31+, Safari 7+, IE 10+
- Foundation
  - Except IE7 & IE8

Deploy

Pack, :ship-it:

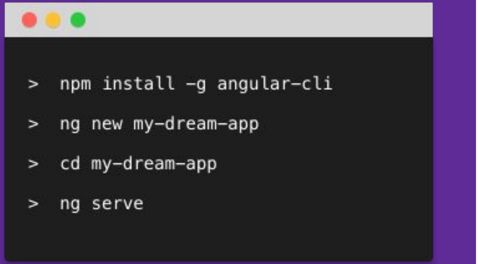
- Concat, Minify
- Tools
  - gulp
  - angular-cli
  - webpack

ANGULAR CLI

Hello CLI

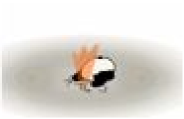
FREE!!! XD

- Angular-CLI



```
>  npm install -g angular-cli
>  ng new my-dream-app
>  cd my-dream-app
>  ng serve
```

Much Obliged



Lecturer: S1ngS1ng

Panelists: Lian Liu, Jerry Yang

Host: Shi H.

Registration: Xiao Liu

Boardcast: Hao, 8Xuan, Karina Rang

Editor: Bohua Tian

Bosses: J.Z., R.Z., J.L., T.S.

Platform: bittiger.io

and .. Thank YOU!

GLHF coding :)

In Seatle?

Add lun to be in our local discussion group

Full-stack?
Front-end?

Add xiao4742 to be in our discussion group

Keep in Touch!

GitHub: S1ngS1ng
LinkedIn: liuxing0514
WeChat: 1025583636