

Technology Series

AngularJS

02

Presented by
Subhash EP



Subhash



ANGULARJS
by Google®

Course Agenda – Introduction

- 01
 - AngularJS core concepts
 - Two-way binding
- 02
 - Routing basics
 - Templating basics
 - Validations introduction
 - Directives concepts
 - What is dependency injection
- 03
 - Development environment
 - Chrome
 - Chrome plugins for AngularJS – Batarang, ng-Inspector
 - Sublime Text
 - Angular seed project
 - Anatomy of Angular Project
 - MVW, MVC, MVVM

Course Agenda – Structuring AngularJS

- 01
 - Modules
 - Built-in
 - Own modules
 - Best practices
- 02
 - Controllers
 - Scope
- 03
 - From a simple HTML page to a complex AngularJS app
 - Ng-prefix
 - Expressions
 - Adding Logic
 - Properties
 - Functions
 - Nesting
 - Dependency Injection and controllers

Course Agenda – Client Side

- 04
 - Events
 - Services
- 05
 - Form element
 - Controls
 - Form Validations

Course Agenda – Routing

- 06
 - Views
 - Routing
 - Factories
 - Providers

Course Agenda – Let us go deeper

- 07
 - AJAX
 - \$http
 - Data binding
 - Two-way Data Binding
 - Simple Page Applications
- 08
 - Directives
 - More on DI

Agenda

Structuring AngularJS

Agenda

Structuring AngularJS

01

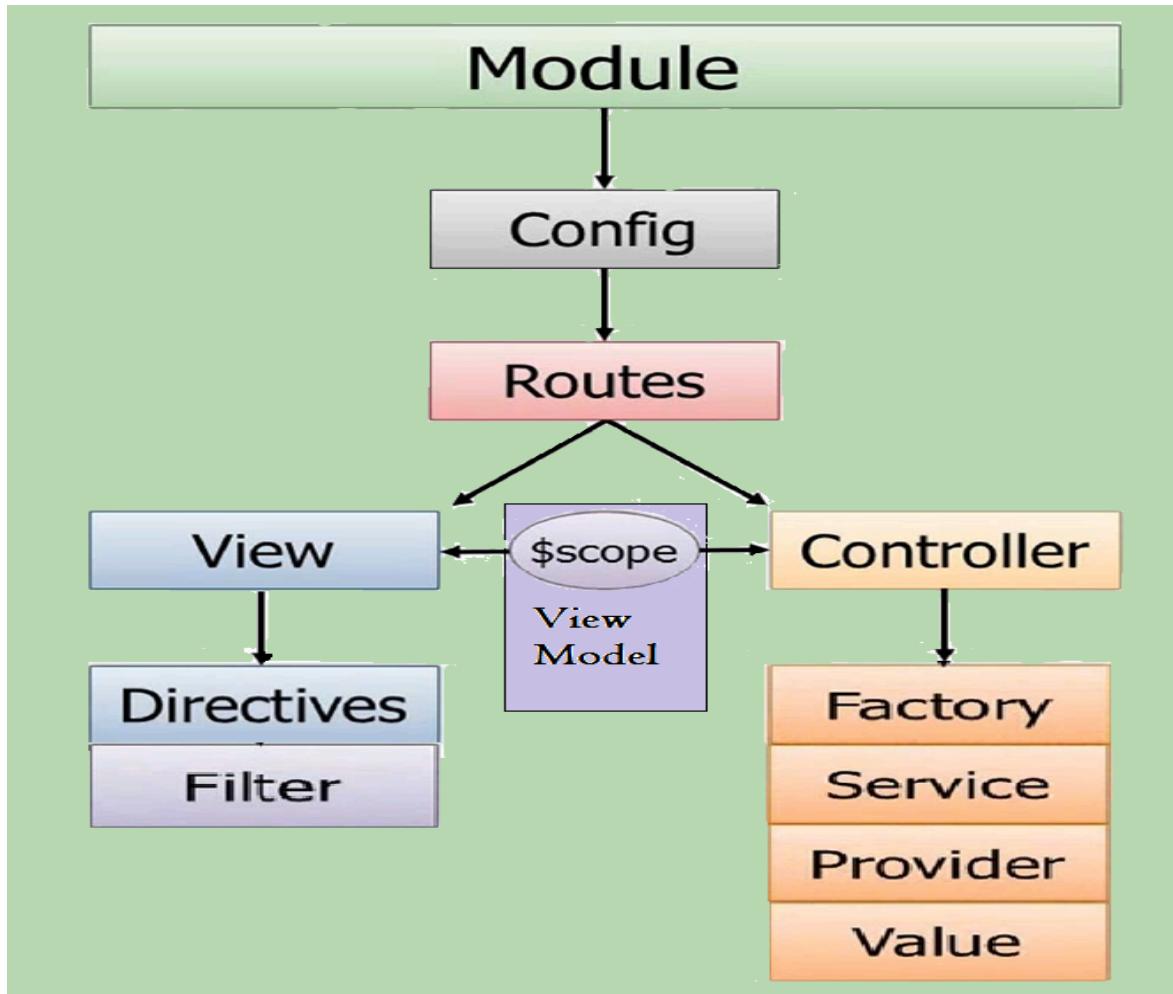
Modules

Built-in

Own modules

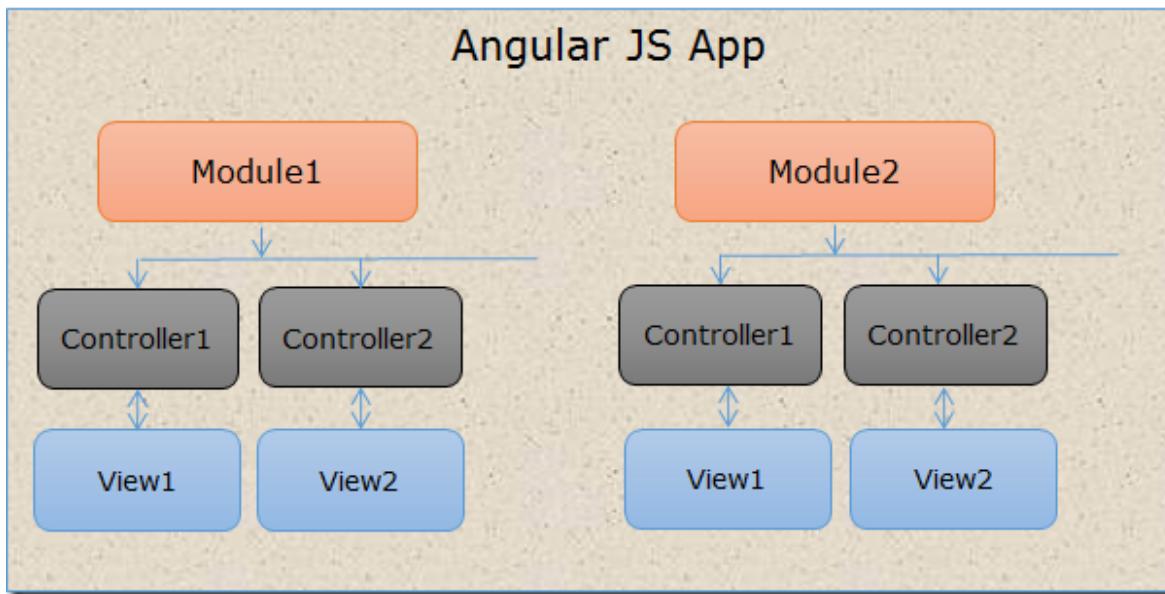
Best practices

AngularJS Architecture



Modules

- Reusable Components
- Container that groups app components under a common name



Seed Project & Modules

- there are different modules defined in the files
 - controllers.js
 - directives.js
 - filters.js
 - services.js, etc
- In app.js module myApp depends on the modules in those JS files

Built-in Modules

- AngularJS has several built-in modules.
- The `ng` module defines
 - many directives, filters, services for our use.
- We have seen
 - `ng-model`, `ng-controller`, etc

ng module

- Core Module
- loaded by default
- When AngularJS application started.

Best Practices

- /app
 - /img -- application level images
 - /css -- application level css
 - /js
 - app.js -- the main app module
 - /modules
 - /login
 - /js
 - controllers.js --controllers for login module
 - directives.js --directives for login module
 - /views -- views for login module
 - /css
 - /img
 - loginModule.js -- Main login module definition
 - /comment
 - /js
 - controllers.js --controllers for comment module
 - directives.js --directives for comment module
 - /views -- views for comment module
 - /css
 - /img
 - commentModule.js -- Main comment module definition
- index.html

Demo Lab

Own Modules

02.01 – 02.02

Agenda

Structuring AngularJS

02

Controllers
Scope

Role of A Controller

- Augment scope
 - by attaching models and functions
 - To be accessed in the view.
- Nothing but a constructor instantiated by AngularJS
- Scope glue between the controller and the view
 - controller can add properties to the scope and the view can access those.

AngularJS versions

- There is a major difference
 - between 1.2.x and 1.3.x
 - As far as Modules and Controllers
- Experience them in our next Demo Labs

Demo Lab

Controllers & Scopes

Agenda

Structuring AngularJS

03 A

From a simple HTML page to a complex AngularJS app

Ng-prefix

Expressions

Adding Logic

Properties, Functions

Simple HTML to Complex App

- Simple HTML templates
- Add Modules
- Add Controllers
- Add Properties and Functions to Scope

ng- prefix

- The ng- prefixed attributes carry special meaning in AngularJS apps.
- Technically we call them directives.

Agenda

Structuring AngularJS

03 B

Nesting
Dependency Injection and
controllers

Nesting

- Nesting Controllers
- Controllers can be nested
 - in such cases
 - the inner controller scope
 - inherits the outer controller scope
 - (prototypal inheritance in effect)

That's all



End of Session

©

subhash.ep@gmail.com
linkedin.com/in/subhashep