

Technology Series

AngularJS

03

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

Client Side

Agenda Client Side

04

Events
Services

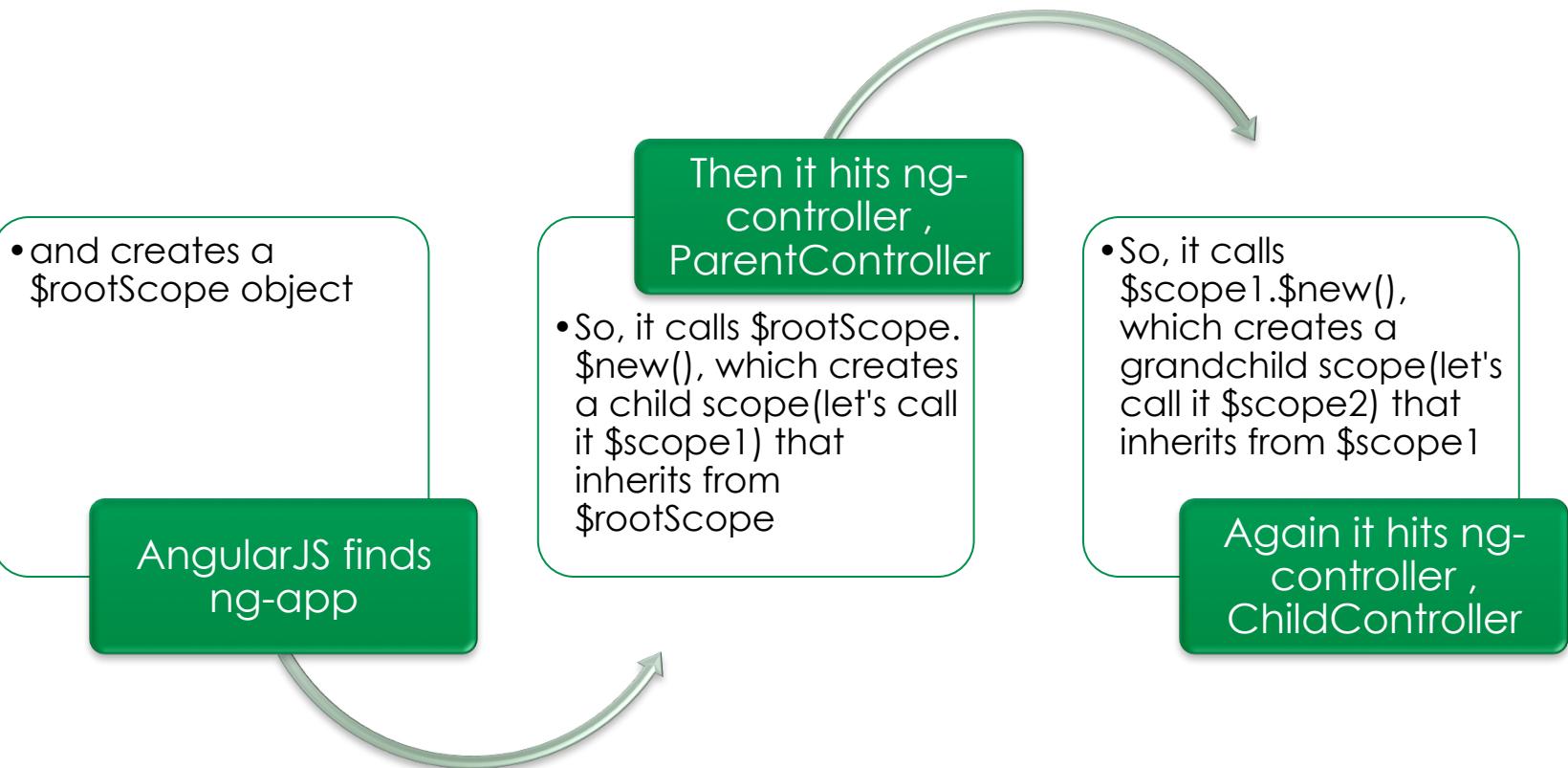
Angular Events

- **ng-click**
 - defines an AngularJS click event
- **ng-hide**
 - `ng-hide="true"` makes an HTML element **invisible**.
 - `ng-hide="false"` makes an HTML element **visible**.
- **ng-show**
 - `ng-show="true"` makes an HTML element **visible**.
 - `ng-show="false"` makes an HTML element **invisible**.

Demo Lab

ng Events

Revisit Scope Hierarchy



Watcher

- A watcher monitors model changes and takes action in response.
- The \$scope object has a function \$watch() that's used to register a watcher.

Demo Lab Watcher

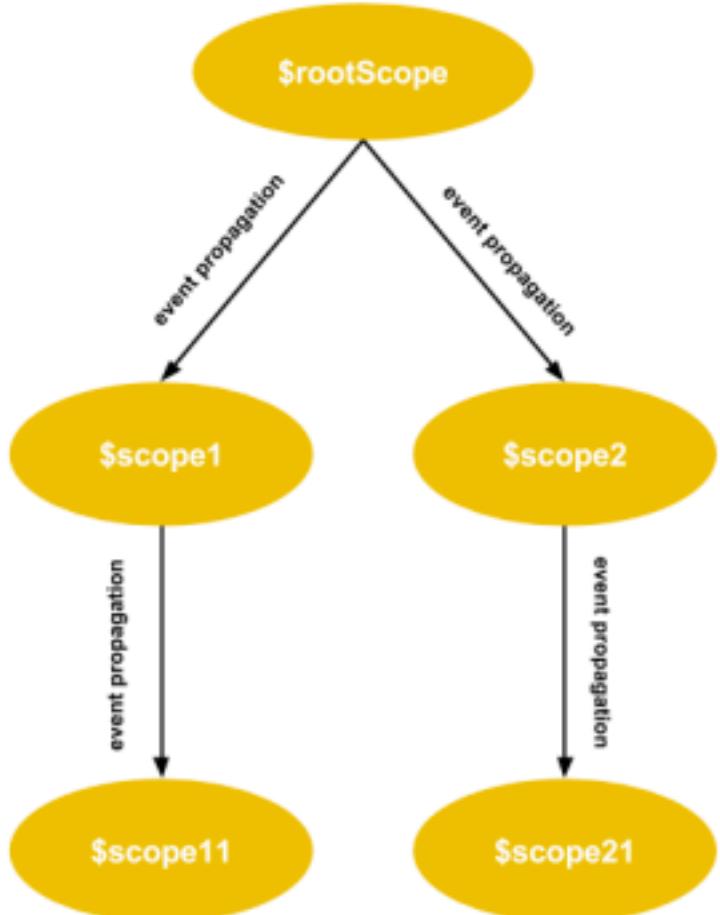
Event Propagation in AngularJS

- AngularJS supports two types of event propagation:
- **Emitting** the event upwards in the scope hierarchy.
- **Broadcasting** the event downwards in the scope hierarchy.

\$broadcast

\$emit

`$rootScope.$broadcast()`



`$rootScope`

`$scope1`

`$scope11`

`$scope11.$emit()`

event propagation

event propagation

event propagation

`$scope11`

Sub

\$scope.\$on Listener

- \$on function
 - registers event listeners
 - that should be called when the event occurs

Demo Lab

\$emit and \$on

Services

- An AngularJS service
 - encapsulates some specific business logic and
 - exposes an API,
 - to be used by other components,
 - to invoke this logic.

Services

- service is an injectable type,
- and this means it can be inserted into different AngularJS components
- It is always singleton!!

Agenda Client Side

05

Form element
Controls
Form Validations

Demo Lab

Form element
Controls
Form Validations

That's all



End of Session

©

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