



Angular - Overview



- Angular 2 Overview
- Components
- Directives
- Services
- Pipe
- HttpClientModule
- Data Binding
- Events
- Observables & Promises
- UI Design

- What is AngularJS
- Comparing Angular 1 and Angular 2
- Environment Setup
- Introduction to QuickStart package

- Structural client side framework for dynamic web application
- What HTML would be if it was created for to create a complete Dynamic Web Apps
- Often used to build SPAs (Single Page Apps)
- Grouping of Code into “Components”

- Server Side Framework
- JavaScript library (JQ, ReactJS, MooToos)
- Architecture of Design Pattern (MVC, MVVM)
- Platform (unlike .Net, Java)
- Plugin, Extension

- Web Apps that load a single HTML page and dynamically update the page as the user interacts with the app.
- It gives the user more fluid experience
- Certain parts of the application are updated but full page is not reloaded

- Components divide the responsibilities of the app
- Re-Use of code
- Easy to read and share with other developers
- Angular2 makes better use of Component

- Speed & Performance
- Simple & Expressive
- Cross platform
- Flexible
- Comprehensive Routing
- 2 Way Data Binding
- Dynamic Templating

- Registering Call backs
- Manipulating DOM programmatically
- Marshalling Data to and from the UI
- Initialization Code

- Template
- Directive
- Model
- Scope
- Component
- Filter
- View
- Data Binding
- Controller
- Dependency Injection
- Injector
- Module
- Service

- In 2014, The Angular team decided to drastically change the Angular framework.
- If you already know Angular1, there is still more to learn as many things have been changed, added or removed
- We will not be looking at angular 1 more in depth

- Re-Usable and more readable code with components
- Reduce Learning Curve
- Performance enhancement
- ES6 – JavaScript 6
- Better Mobile Support
- Embrace FLUX (Design Architecture) & RxJS (Reactive Extension to Java Script)
- Controller and \$scope are gone
- Properties are bound to components instead of \$scope

- Data flow changed
- One way data binding from a component to the UI
- One way data binding from UI to component using event
- Two data binding still available, but not needed much.
 - ng-model is available, but used less & that too in different way

- Similar to Angular 1
- Some Syntactical changes to make things more clear
- New build in Directives

- Now Angular2 is bundled with lots of dependencies. Application setup is done by Angular quickstart
- For running the angular 2 apps, below dependencies need to be first available on your system
 - NodeJS
 - ATOM / Bracket/ Visual Studio

- Below are the instruction of setting up the first angular app named : **HelloWorldAngularApp**
- Create a folder named **HelloWorldAngularApp**
- Unzip the provided quick start folder.
- Go inside the quick start folder and select all the files and cut, and paste in HelloWorldAngularApp folder.
- Remove unnecessary folders.

Remove the not required files with below command.

```
for /f %i in (non-essential-files.txt) do del %i /F /S /Q  
rd e2e /s /q
```

Run below commands to run the application

```
npm install  
npm start
```



- `npm install`: to install the npm module in project folder
- `npm start`: runs compiler and server at the same time, both in “watch- mode”
- `npm run tsc` – runs the typescript compiler once
- `npm run tsc:w` – runs the typescript compiler in watch mode.
- `npm run lite`: -runs the lite-server, a light-weight, static file server
- `npm run typings`: runs the typing tool
- `npm run postinstall`: -called by npm automatically after, it successfully completes package installation.

Thank You!

Email: info@yash.com

Web: www.yash.com

© YASH Technologies, 1996-2013. All rights reserved.

The information in this document is based on certain assumptions and as such is subject to change. No part of this document may be reproduced, stored or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of YASH Technologies Inc. This document makes reference to trademarks that may be owned by others. The use of such trademarks herein is not as assertion of ownership of such trademarks by YASH and is not intended to represent or imply the existence of an association between YASH and the lawful owners of such trademarks.