Introduction

Angular JS

Why AngularJS?

- Building dynamic views with other JS frameworks are harder.
- It allows us to extend HTML vocabulary on your application to attain the same.
- The resulting environment is extraordinarily expressive, readable, and quick to develop.

Features

It is full-featured SPA framework

- Directives
- Data Binding
- Templates
- MVC, MVVM, MVW
- Routing
- Testing
- JqLite

What are Directives?

- They teach HTML new tricks
- We can use pre-defined / custom / third party directives.
- Eg: ng-app -> used to flag an element which Angular should consider to be the root element of our application.

Bootstrap

- The injector that will be used for dependency injection within this app is created.
- The injector will then create the root scope that will become the context for the model of our application.
- Scopes provide mechanism for watching the model and provide event processing life-cycle.
- Angular will then "compile" the DOM starting at the ngApp root element, processing any directives and bindings found along the way.

View, Controllers and Scope

- Scope(ViewModel) is the glue between View and Controller.
- Dynamic Injection
- Two-way bindings

Pros

- Very little boilerplate.
- It's a complete framework, it has routing, controls, templating, two-way binding, it handles REST and AJAX, currency formatting, date formatting, you name it.
- Development is a lot more rapid. It works extremely well with JQuery plugins.
- Directives and modules and code re-usability is very slick.
- It forces your developers into a certain structure, so you know where to look for things.

Cons

- It forces your developers into a certain structure.
- The learning curve is a little steeper.
- Memory management can become an issue if you try to go "too big" with your interfaces.

References

http://angularjs.org/

Questions

?

Thank

U