AngularJS History

* AngularJS version 1.0 was released in 2012.
* AngularJS is a JavaScript framework. It is a library written in JavaScript.
* Global link to Add Angular js-<script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>

AngularJS Extends HTML

AngularJS extends HTML with **ng-directives**.

* The **ng-app** directive defines an AngularJS application.
* The **ng-model** directive binds the value of HTML controls (input, select, textarea) to application data.
* The **ng-bind** directive binds application data to the HTML view.

<div ng-app="">  
  <p>Name: <input type="text" ng-model="name"></p>  
  <p ng-bind="name"></p>  
</div>

Example explained:

AngularJS starts automatically when the web page has loaded.

The **ng-app** directive tells AngularJS that the <div> element is the **root element** of an AngularJS application.

The **ng-model** directive binds the value of the input field to the application variable **name**.

The **ng-bind** directive binds the **innerHTML** of the <p> element to the application variable **name**.

The ng-init directive defines **initial values** for an AngularJS application.

<div ng-app="" ng-init="firstName='John'">  
  
<p>The name is <span ng-bind="firstName"></span></p>  
  
</div>

AngularJS Expressions

AngularJS expressions can be written inside double braces: {{ *expression* }}.

<div ng-app="" ng-init="myCol='lightblue'">  
<input style="background-color:{{myCol}}" ng-model="myCol" value="{{myCol}}">  
</div>

AngularJS Modules

* The module is a container for the application controllers.
* Controllers always belong to a module.
* A module is created by using the AngularJS function angular.module

var app = angular.module("myApp", []);

* Add a controller to your application, and refer to the controller with the ng-controller directive.
* <div ng-app="**myApp**" ng-controller=**"myCtrl"**>  
  {{ firstName + " " + lastName }}  
  </div>  
    
  <script>  
    
  var app = angular.module(**"myApp"**, []);  
    
  app.controller(**"myCtrl"**, function($scope) {  
      $scope.firstName = "John";  
      $scope.lastName = "Doe";  
  });  
    
  </script>
* AngularJS has a set of built-in directives which you can use to add functionality to your application.

<div ng-app="myApp" w3-test-directive></div>  
  
<script>   
var app = angular.module("myApp", []);  
  
app.directive("w3TestDirective", function() {  
    return {  
        template : "I was made in a directive constructor!"  
    };  
});  
</script>

Create New Directives

When naming a directive, you must use a camel case name , w3TestDirective, but when invoking it, you must use -separated name, w3-test-directive:

<body ng-app="myApp">  
  
<w3-test-directive></w3-test-directive>  
  
<script>  
var app = angular.module("myApp", []);  
app.directive("w3TestDirective", function() {  
    return {  
        template : "<h1>Made by a directive!</h1>"  
    };  
});  
</script>  
</body>

->Repeating HTML Elements

The ng-repeat directive repeats an HTML element:

Example

<div ng-app="" ng-init="names=['Jani','Hege','Kai']">

<ul>

<li ng-repeat="x in names">

{{ x }}

</li>

</ul>

</div>

You can invoke a directive by using:

* Element name
* Attribute
* Class
* Comment