AngularJs is a Javscript open source front-end **framework** that is mainly used to develop single page web applications(SPAs). **AngularJs is basics of HTML,CSS and Javascript**

Library file:

<script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.9/angular.min.js"></script>

 It is recommended that you load the AngularJS library either in the <head> or at the start of the <body>.This is because calls to angular.module can only be compiled after the library has been loaded.

**Model:** It is responsible for managing application data.

**View:** It is responsible for displaying all data or only a portion of data to the users.

**Controller:** It is responsible to control the relation between models and views.

**Module (not Model)**:

1. The module is a **container for the different parts of an application**.

2. The module is a container for the application controllers.

3. Controllers always belong to a module.

A module is created by using the AngularJS function angular.module

<div ng-app="myApp">...</div>//view part  
  
<script>  
  
var app = angular.module("myApp", []);//script for module only  
  
</script>

The ng-app directive also tells AngularJS that the <div> element is the "owner" of the AngularJS application.

It is common in AngularJS applications to put the **module** and the **controllers** in JavaScript files like **myApp**.**js** and **myCtrl**.**js. we can include in HTML file using** <script src=”myapp.js”> and <script src=”myctrl.js”>

**Controller**

AngularJS controllers **control the data** of AngularJS applications

AngularJS applications are controlled by controllers.

<body>

<div ng-app="myapp" ng-controller="myctrl">

<input type="text" ng-model="firstname">

You Entered:{{firstname}}

<script>

angular.module("myapp",[]).controller("myctrl",function($scope){

$scope.firstname='Akash';

});

**Two-Way Binding:**

<div ng-app="myapp" ng-controller="myctrl">

//any change in this(View) reflected in view

<input type="text" ng-model="firstname">

You Entered:{{firstname}}

<script>

angular.module("myapp",[]).controller("myctrl",function($scope){

$scope.firstname='Akash';//any change in this(Module) reflected in view

});</script>

**Scope**:

1. The scope is the binding part between the HTML (view) and the JavaScript

(controller).

2. The scope is available for both the view and the controller.

3. When you make a controller in AngularJS, you pass the $scope object as an argument:

Example: The scope is available for both the view and the controller

<div ng-app="myapp" ng-controller="myctrl">

<h1>{{name}}</h1>

<script>

angular.module("myapp",[]).controller("myctrl",function($scope)

{

$scope.name="Akash";

});

</script>

**Directives:**

The ng-app directive initializes an AngularJS application.(root element)

The ng-init directive initializes application data.

The ng-model directive binds the value of HTML controls (input, select, textarea) to application data.

The ng-repeat directive repeats an HTML element:

**Ng-repeat**

<div ng-app="" ng-init="names=[  
{name:'Jani',country:'Norway'},  
{name:'Hege',country:'Sweden'},  
{name:'Kai',country:'Denmark'}]">

<ul>  
  <li ng-repeat="x in names">  
    {{ x.name + ', ' + x.country }}  
  </li>  
</ul>  
  
</div>

**$$** prefix in AngularJS is used as a private variable.

**$** prefix is used to define angular core functionalities such as variable, parameter, property or method, etc.

**ngRoute**

The **ngRoute** module helps in accessing different pages of an application without reloading the entire application.

 AngularJS application can have only **one root scope** but can have multiple child scopes.

# **AngularJS Data Binding**

Data binding is a very useful and powerful feature used in software development technologies. It acts as a bridge between the view and business logic of the application.

AngularJS follows Two-Way data binding model.

**filter**

A filter is used to format the value of the expression to display the formatted output.

**<p>**The name is {{ firstName | uppercase }}**</p>**

There are some built-in filters provided by AngularJS such as **Currency, Date, Filter, JSON, Limit, Lowercase, Number, Orderby, and Uppercase**.

**Form Validation**

AngularJS provides client-side form validation. It checks the state of the form and input fields (input, textarea, select), and lets you notify the user about the current state.

* **$dirty**- states that value has been changed.
* **$invalid**- states that value entered is invalid.
* **$error** - states the exact error.

**Directives:**

|  |  |
| --- | --- |
| **DIRECTIVES** | **DESCRIPTION** |
| ng-app | Start of AngularJS application. |
| ng-init | Used to initialise a variable |
| ng-model | ng-model is used to bind to the HTML input controls |
| ng-controller | Attaches a controller to the view |
| ng-bind | Binds the value with HTML tag/element |
| ng-repeat | Repeats HTML template once per each item in the specified collection. |
| ng-show | Shows or hides the associated HTML element |
| ng-readonly | Makes HTML element read-only |
| ng-disabled | Use to disable or enable a button dynamically |
| ng-if | Removes or recreates HTML element |
| ng-click | Custom step on click |

Ng-module directive **binds the values of application data to HTML input controls** in angular JS.

#### Ng-bind directive binds Application data to HTML tags in angular JS.

 ng-bind or data-ng-bind or x-ng-bind I OR ng:binds same.

### **ii. ng-bind**

By using ng-bind directive the content of an **HTML element** replace with the value of a given variable as well as an expression.

1. <!DOCTYPE html>
2. <html >
3. <head>
4. <script src="~/Scripts/angular.js"></script>
5. </head>
6. <body ng-app="">
7. <div>5 + 5 = <span ng-bind="5 + 5"></span> <br /></div>
8. </body>
9. </html>

**Output:**

5+5=10

### **ng-model**

The ng-model directive uses to bind the elements such as <select>,<textarea>,<input> to a particular property on $scope object to assign the value of a property will be the property of element vice versa.

1. <head>
2. <script src="~/Scripts/angular.js"></script>
3. </head>
4. <body ng-app>
5. Enter Text:<input type="text" ng-model="name" />
6. <div>
7. This is the tutorial of {{name}}
8. </div>
9. </body>

**Output:**

**Enter Text:**

This is the tutorial of

**Enter Text: Angular JS**

This is the tutorial of Angular Js.

Two way data binding Exampe:

**app.component.html file:**

1. **<h2>**Two-way Binding Example**</h2>**
2. **<input** [(ngModel)]="fullName" **/>** **<br/><br/>**
3. **<p>** {{fullName}} **</p>**

If we consider an AngularJS application to consist of:

* View, which is the HTML.
* Model, which is the data available for the current view.
* Controller, which is the JavaScript function that makes/changes/removes/controls the data.

Then the scope is the Model.

The scope is a JavaScript object with properties and methods, which are available for both the view and the controller.

Filter: Imagine you can sort list o employee by salary

<ul>  
  <li ng-repeat="x in names | orderBy:'country'">  
    {{ x.name + ', ' + x.country }}  
  </li>  
</ul>