

Angular JS

Table of Content

Angular JS	1
1. About Angular	2
1.1 What is AngularJS?	2
1.2 Features	2
1.3 Core Features	2
1.4 Concepts	3
1.5 Advantages of AngularJS	4
1.6 Disadvantages of AngularJS	4
1.7 The AngularJS Components	4
Angular JS Practise Exercise	4
ng-model	4
ng-bind	5
ng-init	5
expression	6
Controller	6
ng-repeat	6
filters:	6
\$http	7
ng-disabled	7
ng-show	8
ng-click	8
form	8
validation	9
ng-include	10

1. About Angular

1.1 What is AngularJS?

AngularJS is an open source web application framework. It was originally developed in 2009 by Misko Hevery and Adam Abrons. It is now maintained by Google. Its latest version is 1.3.14.

AngularJS is a structural framework for dynamic web apps. It lets you use HTML as your template language and lets you extend HTML's syntax to express your application's components clearly and succinctly. Angular's data binding and dependency injection eliminate much of the code you currently have to write. And it all happens within the browser, making it an ideal partner with any server technology.

1.2 Features

- AngularJS is a powerful JavaScript based development framework to create RICH Internet Application(RIA).
- AngularJS provides developers options to write client side application (using JavaScript) in a clean MVC(Model View Controller) way.
- Application written in AngularJS is cross-browser compliant. AngularJS automatically handles javascript code suitable for each browser.
- AngularJS is open source, completely free, and used by thousands of developers around the world. It is licensed under the Apache License version 2.0.

Overall, AngularJS is a framework to build large scale and high performance web application while keeping them as easy-to-maintain.

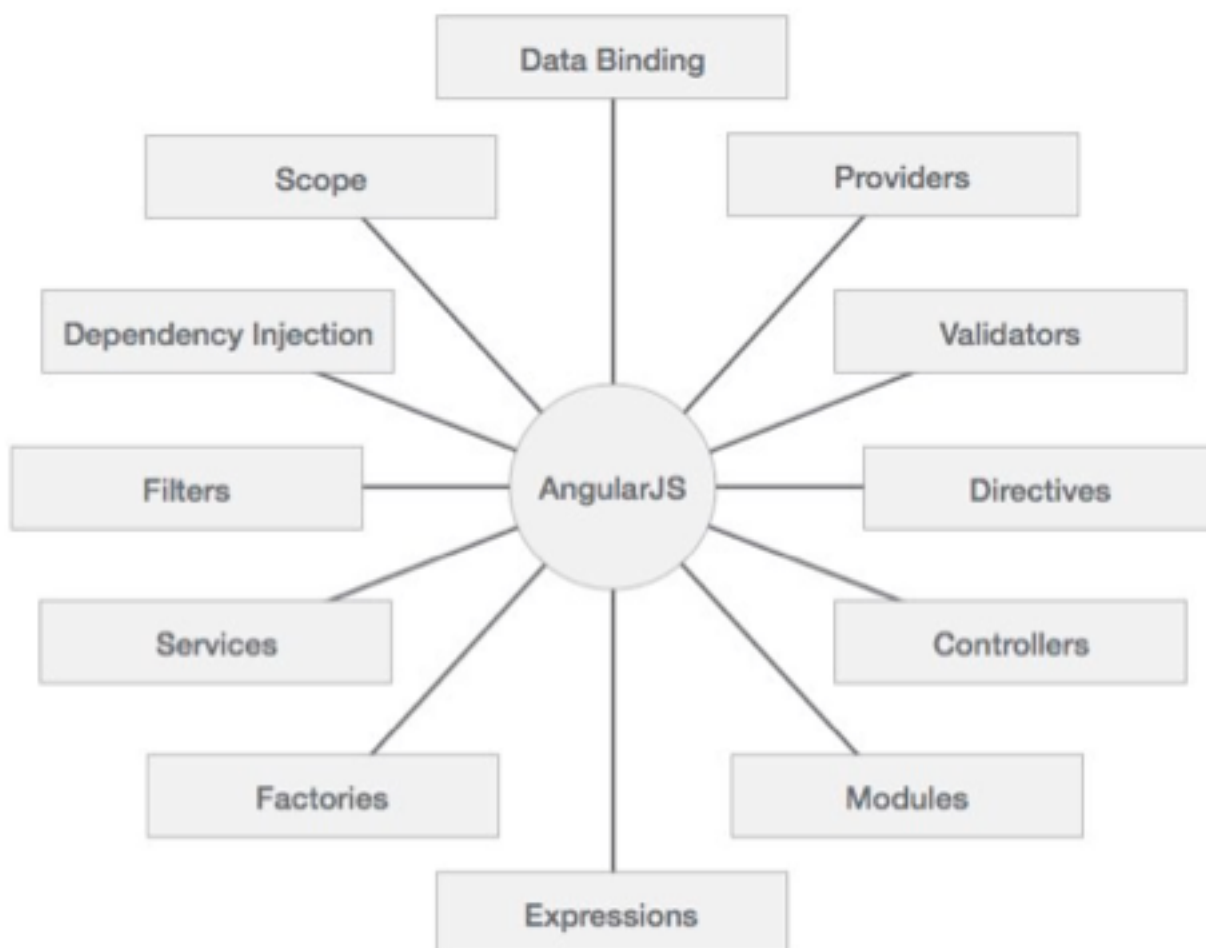
1.3 Core Features

- Data-binding: It is the automatic synchronization of data between model and view components.
- Scope: These are objects that refer to the model. They act as a glue between controller and view.
- Controller: These are Javascript functions that are bound to a particular scope.
- Services: AngularJS come with several built-in services for example \$http to make a XMLHttpRequests. These are singleton objects which are instantiated only once in app.
- Filters: These select a subset of items from an array and returns a new array.
- Directives: Directives are markers on DOM elements (such as elements, attributes, css, and more). These can be used to create custom HTML tags that serve as new, custom widgets. AngularJS has built-in directives (ngBind, ngModel...)
- Templates: These are the rendered view with information from the controller and model. These can be a single file (like index.html) or multiple views in one page using "partials".
- Routing: It is concept of switching views.

- **Model View Whatever:** MVC is a design pattern for dividing an application into different parts (called Model, View and Controller), each with distinct responsibilities. AngularJS does not implement MVC in the traditional sense, but rather something closer to MVVM (Model-View-ViewModel). The Angular JS team refers to it humorously as Model View Whatever.
- **Deep Linking:** Deep linking allows you to encode the state of application in the URL so that it can be bookmarked. The application can then be restored from the URL to the same state.
- **Dependency Injection:** AngularJS has a built-in dependency injection subsystem that helps the developer by making the application easier to develop, understand, and test.

1.4 Concepts

Following diagram depicts some important parts of AngularJS which we will discuss in detail in the subsequent chapters.



1.5 Advantages of AngularJS

- AngularJS provides capability to create Single Page Application in a very clean and maintainable way.
- AngularJS provides data binding capability to HTML thus giving user a rich and responsive experience.
- AngularJS code is unit testable.
- AngularJS uses dependency injection and make use of separation of concerns.
- AngularJS provides reusable components.
- With AngularJS, developer write less code and get more functionality.
- In AngularJS, views are pure html pages, and controllers written in javascript do the business processing.

On top of everything, AngularJS applications can run on all major browsers and smart phones including Android and iOS based phones/tablets.

1.6 Disadvantages of AngularJS

Though AngularJS comes with lots of plus points but same time we should consider the following points:

- Not Secure : Being JavaScript only framework, application written in AngularJS are not safe. Server side authentication and authorization is must to keep an application secure.
- Not degradable: If your application user disables Javascript then user will just see the basic page and nothing more.

1.7 The AngularJS Components

The AngularJS framework can be divided into following three major parts:

- ng-app : This directive defines and links an AngularJS application to HTML.
- ng-model : This directive binds the values of AngularJS application data to HTML input controls.
- ng-bind : This directive binds the AngularJS Application data to HTML tags.

Angular JS Practise Exercise

ng-model

```
<!DOCTYPE html>
<html lang="en-US">
<script src="http://ajax.googleapis.com/ajax/libs/angularjs/1.3.14/
angular.min.js"></script>
<body>

<div ng-app="">
```

```
        <p>Name : <input type="text" ng-model="name"></p>
        <h1>Hello {{name}}</h1>
    </div>

</body>
</html>
```

ng-bind

```
<!DOCTYPE html>
<html>
<script src="http://ajax.googleapis.com/ajax/libs/angularjs/1.3.14/
angular.min.js"></script>
<body>

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

</body>
</html>
```

ng-init

Sample 1:

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

Sample 2:

```
<div ng-app="" ng-init="quantity=1;cost=5">
    <p>Total in dollar: {{ quantity * cost }}</p>
</div>
```

Sample 3:

```
<div ng-app="" ng-init="person={firstName:'John',lastName:'Doe'}">
    <p>The name is {{ person.lastName }}</p>
</div>
```

Sample 4:

```
<div ng-app="" ng-init="points=[1,15,19,2,40]">
    <p>The third result is {{ points[2] }}</p>
</div>
```

expression

```
<!DOCTYPE html>
<html>
<script src="http://ajax.googleapis.com/ajax/libs/angularjs/1.3.14/
angular.min.js"></script>
<body>
<div ng-app="">
  <p>My first expression: {{ 5 + 5 }}</p>
</div>
</body>
</html>
```

Controller

```
<div ng-app="myApp" ng-controller="myCtrl">
  First Name: <input type="text" ng-model="firstName"><br>
  Last Name: <input type="text" ng-model="lastName"><br>
  <br>
  Full Name: {{firstName + " " + lastName}}
</div>

<script>
var app = angular.module('myApp', []);
app.controller('myCtrl', function($scope) {
  $scope.firstName= "John";
  $scope.lastName= "Doe";
});
</script>
```

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>
```

filters:

uppercase:

```
<div ng-app="" ng-controller="personCtrl">
  <p>The name is {{ lastName | uppercase }}</p>
```

</div>

lowercase:

```
<div ng-app="" ng-controller="personCtrl">
  <p>The name is {{ lastName | lowercase }}</p>
</div>
```

currency:

```
<div ng-app="" ng-controller="costCtrl">
  <input type="number" ng-model="quantity">
  <input type="number" ng-model="price">
  <p>Total = {{ (quantity * price) | currency }}</p>
</div>
```

orderby:

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

\$http

```
<div ng-app="myApp" ng-controller="customersCtrl">
  <ul>
    <li ng-repeat="x in names">
      {{ x.Name + ', ' + x.Country }}
    </li>
  </ul>
</div>
<script>
var app = angular.module('myApp', []);
app.controller('customersCtrl', function($scope, $http) {
  $http.get("http://www.w3schools.com/angular/customers.php")
    .success(function(response) {$scope.names = response.records;});
});
</script>
```

ng-disabled

```
<div ng-app="">
  <p>
    <button ng-disabled="mySwitch">Click Me!</button>
  </p>

  <p><input type="checkbox" ng-model="mySwitch">Button</p>
```

```
</div>
```

ng-show

```
<div ng-app="">
  <p ng-show="true">I am visible.</p>
  <p ng-show="false">I am not visible.</p>
</div>
```

ng-click

```
<div ng-app="myApp" ng-controller="personCtrl">
<button ng-click="toggle()">Toggle</button>
<p ng-hide="myVar">
  First Name: <input type="text" ng-model="firstName"><br>
  Last Name: <input type="text" ng-model="lastName"><br>
  <br>
  Full Name: {{firstName + " " + lastName}}
</p>
</div>

<script>
var app = angular.module('myApp', []);
app.controller('personCtrl', function($scope) {
  $scope.firstName = "John",
  $scope.lastName = "Doe"
  $scope.myVar = false;
  $scope.toggle = function() {
    $scope.myVar = !$scope.myVar;
  };
});
</script>
```

form

```
<div ng-app="myApp" ng-controller="formCtrl">
  <form novalidate>
    First Name:<br>
    <input type="text" ng-model="user.firstName"><br>
    Last Name:<br>
    <input type="text" ng-model="user.lastName">
    <br><br>
    <button ng-click="reset()">RESET</button>
  </form>
  <p>form = {{user}}</p>
```



```

    <p>master = {{master}}</p>
</div>

<script>
var app = angular.module('myApp', []);
app.controller('formCtrl', function($scope) {
    $scope.master = {firstName: "John", lastName: "Doe"};
    $scope.reset = function() {
        $scope.user = angular.copy($scope.master);
    };
    $scope.reset();
});
</script>

```

validation

```

<!DOCTYPE html>
<html>
<script src="http://ajax.googleapis.com/ajax/libs/angularjs/1.3.14/
angular.min.js"></script>
<body>

<h2>Validation Example</h2>

<form ng-app="myApp" ng-controller="validateCtrl"
name="myForm" novalidate>

<p>Username:<br>
    <input type="text" name="user" ng-model="user" required>
    <span style="color:red" ng-show="myForm.user.$dirty && myForm.user.
    $invalid">
    <span ng-show="myForm.user.$error.required">Username is required.</
    span>
    </span>
</p>

<p>Email:<br>
    <input type="email" name="email" ng-model="email" required>
    <span style="color:red" ng-show="myForm.email.$dirty &&
myForm.email.$invalid">
    <span ng-show="myForm.email.$error.required">Email is required.</
    span>
    <span ng-show="myForm.email.$error.email">Invalid email address.</
    span>
    </span>
</p>

<p>
    <input type="submit"
    ng-disabled="myForm.user.$dirty && myForm.user.$invalid ||
    myForm.email.$dirty && myForm.email.$invalid">

```

```
</p>

</form>

<script>
var app = angular.module('myApp', []);
app.controller('validateCtrl', function($scope) {
    $scope.user = 'John Doe';
    $scope.email = 'john.doe@gmail.com';
});
</script>

</body>
</html>
```

ng-include

```
<!DOCTYPE html>
<html ng-app="">
<link rel="stylesheet" href = "http://maxcdn.bootstrapcdn.com/
bootstrap/3.2.0/css/bootstrap.min.css">
<script src= "http://ajax.googleapis.com/ajax/libs/angularjs/1.3.14/
angular.min.js"></script>
<body ng-controller="userCtrl">

<div class="container">
<div ng-include="'myUsers_List.htm'"></div>
<div ng-include="'myUsers_Form.htm'"></div>
</div>

<script src= "myUsers.js"></script>

</body>
</html>
```

