SATURDAY 24 JANUARY 2015 13:13

Angular JS Tutorial

Introduction

- It was released in 2012
- It is backed by Google development team
- It is a JavaScript framework

Overview

AngularJS is distributed as a **JavaScript file**, it can be added to a web page like this:

```
<script
src="http://ajax.googleapis.com/ajax/libs/angula
rjs/1.2.26/ angular.min.js"></script>
```

It extends HTML attributes with **Directives**

We can write **Expressions** which are much like JavaScript expressions

Applications are controlled by **Controllers**

Modules make the application more readable, and keep the global namespace clean

AngularJS provides client-side **validation**

AJS Expressions

AngularJS expressions are written inside double braces:

```
<div ng-app="">
    <p>My first expression: {{ 5 + 5 }}</p>
</div>
```

AngularJS expressions are much like JavaScript expressions: They can contain literals, operators, and variables

If you remove the *ng-app* directive, HTML will display the expression as it is, without solving it

AJS Directives

AngularJS extends HTML attributes with Directives

- ***ng-app***
    - defines an AngularJS application
    - it will automatically initialize the application when a web page is loaded
    - it can connect code modules
- ***ng-model***
    - binds the value of HTML controls to application data
    - provides type validation for application data
    - provides status for application data
- ***ng-bind*** binds application data to HTML view

  ng-repeat repeats an HTML element

  ```
  <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>
  ```

  ng-init defines initial values for an AngularJS application

    - normally, you don't use *ng-init*, you use a controller or module i instead (see below)

AJS Controllers

- Controllers control the data of applications
- The *ng-controller* directive defines the application controller

  A Controller is a JavaScript object

  ```
  <div ng-app=“myApp” ng-
  controller="personController">
      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>
      function personController($scope) {
          $scope.firstName="John",
              $scope.lastName="Doe"
      }
  </script>
  ```

  AngularJS will invoke personController with a *$scope* object

  - *$scope* is the application object

  All application Controllers should belong to a module (see below)

  It is common to store Controllers in external files

  ```
  <script src="personController.js"></script>
  ```

  - *personController.js* is the name of the file which contains our controller

AJS Modules

- Modules define applications
- Modules make your application more readable
- Modules keep the global namespace clean

  All controllers should belong to a module

  ```
  <!DOCTYPE html>
  <html>
  <head>
      <script
  src="http://ajax.googleapis.com/ajax/libs/angula
  rjs/         1.2.26/angular.min.js"></script>
  </head>
  <body>
  <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>
  </body>
  </html>
  ```

  - *MyCtrl* controller is a property of the *MyApp* module

AJS Validation

- AngularJS controls and forms (collection of input controls) can validate input data
- Client-side validation cannot alone secure user input. Server side validation is also necessary