1)When first version of angular released ?Note :MiškoHevery

Gopi | Kavya | Dipankar |

Ng-app refers to the application data

Ng-model binds the html input control to the application data.

Ng-bind binds the application data with the html view.

2)What is the way of extending the HTML with angularjs?Define some directives ?

ng-app,ng-model,ng-bind

q1)Write a simple skelton for the angular js.

3)Write a script to show the relation between the ng-model and the ng-bind.

Above is example of **two way binding**

Q1)Writea angular script to input first\_name,last\_name,employee\_id,dept, salary from the user and then bind it to the output in following manner

Q2)Take input of radius and then show area.

Input

Input

First Name :

Last Name :

EmpId :

Dept :

Salary :

4)Define ng-init with an example ?ng-init + ng-bind + data-ng-bind

Inside the ng-bind , we can do calculation

5)What are **angular expressions** ? Explain with some math example

Q1)Initialize the color variable and style the background-color of textfield

Q2)Initialize the length and width of rectangle and find the area.

Q3)Initialize the object through the name, code and address then display it through the dot operator in expression and ng-bind.

Q4)Initialize an employees object array and then display it manually like {{employees[0].first\_name}}

Problem : Need to iterate through the whole array

So, use the **ng-repeat** for it

Q5)Initialize the Patient object array with attributes :

a)patient\_first\_name

b)patient\_ward\_no

c)patient\_disease

d)patient\_doctor

e)patient\_age

After then display the record in the table using ng-repeat

Q5) 9)Write a script to initialize an array of cities and loop it through ng-repeat. Initialize the array and loop it

6)Write an script to bind the ng-model with the Angular expression.

The **ng-app** directive defines the application, the **ng-controller** directive defines the controller.

Q1)Initialize the first\_name and the last\_name through the angular controller and display the names on ng-model textfield, ng-bind span tag and on the expression tag

Q2)Write a script to input the temperature in centigrade through text field and display the temp in farenheit on button click.The calculation should be done inside the controller

Q3)Initialize the Patient object array with attributes through the controller :

a)patient\_first\_name

b)patient\_ward\_no

c)patient\_disease

d)patient\_doctor

e)patient\_age

After then display the record in the table using ng-repeat

8)Createa angular module with help of controller.

**Directive :**

Different properties of directive :

**a1)**restrict**a2)**scope **a3)**template a4)templateUrl a5)controller a6)link

9)Create a new custom directive with company details .

employee\_id, employeefirstname ,employeelastname, salary

Use it with attribute | ‘E’, ‘A’, class | restrict : ‘C’, comment | ‘M’ | replace : ‘true’

restrict  |template  | replace

Ref : https://www.sitepoint.com/practical-guide-angularjs-directives/

* Note :replace – This specifies if the generated template will replace the HTML element on which the directive is attached. In our case we have used the directive as <hello-world></hello-world>, and replace is set to true. So, after the directive is compiled, the produced output template replaces <hello-world></hello-world>. The final output is <h3>Hello World!!</h3>. If you set replace to false, the default, the output template will be inserted into the element on which the directive is invoked.

9a1)Create a simple directive with link events as mouseover,mouseout.Change the color of paragraph tag. Inside the mouseover doscope.$apply to change the scope value. Change last\_name by Modi. Also use scope.$watch for chacking any change in the scope variable last\_name. If yes then display some message with console.log

elem.bind('click', function() {

elem.css('background-color', 'white');

scope.$apply(function() {

scope.color = "white";

});

9a2)Add the scope.$apply for background-color,color on mouseover and out.

Also give pinter (hand) on mouseover.

element.css(‘cursor’,’pointer’)

Note :The link function is mainly used for attaching event listeners to DOM elements, watching model properties for changes, and updating the DOM.

9a3)Write a script for making adirective such that initialize the $scope.carsPrint every cars in li tag with strong tag through directive. Take restrict as ‘A’,’C’

Css : font-size , color

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

<my-directive name='{{x}}'></my-directive>

</li>

</ul>

All are isolate scope :

* @ - pass by value (attribute binding)
* & - pass by method (expression binding)
* = - pass by reference (two-way data binding)

*Ref : https://thinkster.io/egghead/isolate-scope-am*

First : use the ‘@’ scope tag

9a4)Create a directive superman so that it will alert ‘I am superman’. There should be no parameter to the link function. Again make two directive with **superman** and **stronger** .<span superman stronger></span>

**Ref :**<https://egghead.io/lessons/angularjs-isolate-scope-two-way-binding>

9a5)Write a script by making use of $watch . Create a form for the employee containing emp\_first\_name, emp\_last\_name, emp\_sal . Add button so that it can be pushed to the array.When the size os array increases greater then 5 the make the table background as red.

**Splitting the scope** :

**Isolate Scope @**

Q1)Create a directive for first assigning the flavor through the tag itself like <myDirective flavor=”Vanila”>, later on do it with the help of **Scope @**.

Again the flavor from the controller.like : <myDirective flavor=”{{mtTest}}”>. myTest is initializing from the controller.

**Isolate scope =**

Application :

Q1)Makea application to add any item to the list and remove it using splice(x,1);

**if ($scope.products.indexOf($scope.addMe) == -1) {**            $scope.products.push($scope.addMe);  
**} else {  
            $scope.errortext = "The item is already in your shopping list.";  
        }**

9a3)angular directive compile phase : <https://www.bennadel.com/blog/2794-when-do-you-need-to-compile-a-directive-in-angularjs.htm>

Note :link and compile do not work together.

In the directive definition object, if you **only** define link, that's like shorthand for having an empty compile function with an empty preLink function with your code in the postLink function. As soon as you define compile, link is ignored by angular, because compile should return the linking functions.

If you only return one function from compile, then it'll be executed **post** link.

Or, put differently, link is just a shortcut to the postLink function that gets called after the scope has been linked by compile.

10)Write an angular script for the email validation.Make a form and then an email textfield.validate email with

**Application status : directive can provide the status of the application data.(invalid,dirty,touched,error)**

11)Write an script to show different types of status for the email.Like : invalid,dirty,touched,error

* ng-empty
* ng-not-empty
* ng-touched
* ng-untouched
* ng-valid
* ng-invalid
* ng-dirty
* ng-pending
* ng-pristine

Atleast red for ng-invalid and green for ng-valid.

**$rootScope : global scope also outside the controller**

Q1)Initialize the color with $rootScope and with the $scope inside the controller.Check it on the different div 1st)Inside ng-controller 2nd)outside the ng-controller

Q2)Initialize the 3 controller’s .Also assign the first\_name , last\_name too all controllers but the dept through the $rootScope. Display all with first\_name,last\_name,dept.

**Filter’s**

* currency Format a number to a currency format.
* date Format a date to a specified format.
* filter Select a subset of items from an array. -
* json Format an object to a JSON string.
* limitTo Limits an array/string, into a specified number of elements/characters.–

q1)Create a textfield.Take it’s ng-model as key then filter the size odul array by key.

* lowercase Format a string to lower case. -
* number Format a number to a string. –

q1)Write a digit 100000 and then apply number filter

* orderBy Orders an array by an expression.
* uppercase Format a string to upper case. -

Q1)Write a scipt to filter the first name with **uppercase and lowercase**

Q2)Add filter to the directive.Add filter to ng-repeat. Create a2-dimensional array with name and country. Then use filter **orderby** by country

Q3)Write a script for the currency filter.

Q4a)Write a script to initialize the phone directory in controller and filter the ordered list using the textfield for search.

Hint : | filter : key

Q4b)Write a script to get the record from [h**ttps://jsonplaceholder.typicode.com/posts**](https://jsonplaceholder.typicode.com/posts). Now perform following action :

☺Display the title as uppercase. Add a text to search by ng-model=search’

☺Sort by **orderBy** filter to **‘title’.**Make ng-click on the **th** tag and call for sortMe() function

☺

Ref :http://tutlane.com/tutorial/angularjs/angularjs-forms-data-binding-tutorial-with-example

Filter remaining :

Q5)Sort array by clicking on name.

**Services :**

1. $location.absUrl(); | $location.path()
2. $http service
3. $timeout
4. $interval : console something after 8000 msec

$interval(function(){

},8000);

Q1)Write a script to create an service for square and subtract number and return the value.

app.service(‘myService’,function(){

this.sum = function(a,b){

return (a+b);

}

}

Q2)Calculate the salary by passing the basic salary as an argument to the custom service.

Gross\_sal = bs + da+hra+ta-ptx

Create following services :

1)Salary – pass the other services as an dependency like Da,Ta,Hra

getSalary(bs){}

2)Da : 7% of basic salary

getDa(){}

3)Ta : 8% of basic salary

getTa(){}

4)Hra : 10% of the basic salary

getHra(){}

Q3)Write a script to make following services :

**1)Calculator**

a)sum(a,b){}

b)subtract(a,b){}

c)multiply(a,b){}

d)divide(a,b){}

e)modulas(a,b){}

**2)Reactangle**

a)area(a,b){}

b)perimeter(a,b){}

**3)Circle**

a)area(r)

b)circumference(r)

Q4)Dependency between the services.

For this demo let us create a small calculator app that does two things: squares and cubes. We will create following entities in AngularJS:

1. **MathService** – A simple custom angular service that has 4 methods: add, subtract, multiply and divide. We will only use multiply in our example.
2. **CalculatorService** – A simple custom angular service that has 2 methods: square and cube. This service has dependency on MathService and it uses MathService.multiply method to do its work.
3. **CalculatorController** – This is a simple controller that handler user interactions. For UI we have one textbox to take a number from user and two buttons; one to square another to multiply.

**Factory method** :

<http://www.dotnettricks.com/learn/angularjs/understanding-angularjs-factory-service-and-provider>

**Hint..**

app.factory('Myfactory',function(){

varcetObj = {};

cetObj.minus = function(a,b){

return (a-b);

}

returncetObj;

});

q1)Create a factory method calculate for area of triangle.

**Dependency injection :**

What is dependency injection :

Q1)Dependency Injection is a software design pattern in which components are given their dependencies instead of hard coding them within the component.Reusability is there

Q2)Which components can be injected

Hint..value,factory,service,provider,constant

**Ref** :<http://anandmanisankar.com/posts/angularjs-dependency-injection-demystified/>

**Table tag in anguler**

Angular select :

**Angular $http :**

Methods of $http service

* .delete()
* .get()
* .head()
* .jsonp()
* .patch()
* .post()
* .put()

Reference api :<https://jsonplaceholder.typicode.com/>

<https://jsonplaceholder.typicode.com/posts> - Display all the posts

<https://jsonplaceholder.typicode.com/posts/1> - Get the post of id 1

<https://jsonplaceholder.typicode.com/posts/1/comments> - All posts of comment id 1

<https://jsonplaceholder.typicode.com/posts?userId=1> – All posts of user id 1

Link :[http://www.w3schools.com/angular/customers.php](http://www.w3schools.com/angular/customers.php" \t "_blank).

Table Tag :

Q1)Write a script to access the api through the $http service and then display them in the tabular form.

A1)Do**orderBy** filter on the ‘Country’ column.

A2)Douppercase filter on <td></td>

A3)Apply ng-if=”$odd|$even” on the <td></td>

Q2)Fire the get all users , get user by id , post a user from angular js.

Note for the preflight issues :

app.config(function ($httpProvider) {

$httpProvider.defaults.headers.common = {};

$httpProvider.defaults.headers.post = {};

$httpProvider.defaults.headers.put = {};

$httpProvider.defaults.headers.patch = {};

});

**Ref :<https://reqres.in/>**

**Select Box**

1)Using ng-repeat directive

2)Using the ng-options directive

**Promises in angular js**

**Forms : | Formvalidation | ref :** [**http://websystique.com/angularjs/angularjs-form-validation-example/**](http://websystique.com/angularjs/angularjs-form-validation-example/)

Different states are : $valid,$invalid,$pristine,$dirty,$error

   .username.ng-valid {

          background-color: lightgreen;

      }

      .username.ng-dirty.ng-invalid-required {

          background-color: red;

      }

      .username.ng-dirty.ng-invalid-minlength {

          background-color: yellow;

      }

      .email.ng-valid {

          background-color: lightgreen;

      }

      .email.ng-dirty.ng-invalid-required {

          background-color: red;

      }

      .email.ng-dirty.ng-invalid-email {

          background-color: yellow;

      }

**Ref : http://tutlane.com/tutorial/angularjs/angularjs-ng-blur-event-function-with-example**

**novalidate**

**frm.$invalid**

**from.unm.$dirty**

**ng-minlength  
 ng-maxlength**

**ng-messages**

**ng-cloak**

**For validation install the : bower install angular-messages**

**App.controller(‘MyController’,[‘ngMessages’]){}**

Q1)Create a checkbox and then use ng-show to show/hide the h1 value.

Q2)Create 4 radio buttons and on select display their values

ng-switch, ng-switch-when

q3)Do the same for checkbox

q4)angular.copy()

Difference between = and angular.copy()

Completely new object gets created in copy() and only reference in =

Q5)Create 3 fields Username , password and email then validate them such that once after the validation the submit button gets activated either disabled.

**Events :**

**Ref :http://tutlane.com/tutorial/angularjs/angularjs-ng-blur-event-function-with-example**

**Q1)ng-click** directive to increment the counter each time the button is pressed

ng-dblclick

**q2)ng-blur :** Write a script to make a text field.Apply ng0blur in it. When the mouse is out the there is red message bellow the textfield that ‘You are out now’

**q3)ng-change :** Create a select box and display the employee details when it is selected

**q4)ng-focus :** When focused then put some text in the textfield

**q5)ng-keydown :** When key down then the asci value gets displayed in the textfield.callgetKey($event) | event.keyCode

**q6)ng-keyup :**

**q7)ng-mousedown :** Increment counter with each mousedown

**q8)ng-mouseenter : ‘’**

**q9)ng-mouseleave : ‘’**

**Routing in angularjs:**

<scriptsrc="https://ajax.googleapis.com/ajax/libs/angularjs/1.3.14/angular.min.js"></script>

<scriptsrc="https://ajax.googleapis.com/ajax/libs/angularjs/1.3.14/angular-route.min.js"></script>

ngRoute |

1.3.4

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

app.config(['$routeProvider',function($routeProvider){

$routeProvider.when('/home',{

templateUrl : 'home.htm',

controller : 'HomeController'

})….

<script type="text/ng-template" id="home.htm">

<h1>Home is here</h1>

Angular-slider

https://jsfiddle.net/ValentinH/954eve2L/