Assessment – AngularJS

1. How do you create an AngularJS module 'myApp' that is dependent on the modules "myApp.c", "myApp.s","myApp.f" and "myApp.d"?
   1. var myApp = angular.createModule("myApp")

.inject("myApp.c", "myApp.s","myApp.f" ,"myApp.d");

* 1. var myApp = angular.module("myApp")

.inject("myApp.c", "myApp.s","myApp.f" ,"myApp.d");

* 1. var myApp = angular.module("myApp",

["myApp.c", "myApp.s","myApp.f" ,"myApp.d"]);

* 1. var myApp = angular.module("myApp",

"myApp.c", "myApp.s","myApp.f" ,"myApp.d");

1. What are the 5 recipe types apart from the special recipes to create components in AngularJS module?
   1. Value, Factory, Service, Provider, Constant
   2. Singleton, Factory, Service, Provider, Literal
   3. Singleton, Module, Service, Provider, Constant
   4. Value, Builder, Prototype, Provider, Constant
2. What is the name the directive that is used to initialize angularJS application?
   1. ngView
   2. ngModule
   3. ngRoute
   4. ngApp
3. In which AngularJS special component do you place the logic to manipulate DOM?
   1. controller
   2. directive
   3. filter
   4. service
4. Which function should I invoke to refresh the view, if I modify an attribute in the angular scope outside angularJS context?
   1. angular.refresh();
   2. scope.refresh();
   3. controller.apply();
   4. scope.apply();
5. What are the phases in AngularJS’ HTML compilation and what is done in each phase?
   1. **Bootstrap:** Loads angular application module, with all its dependent modules along with ngCore modules

**Execute:** Executes HTML view created after compilation

* 1. **Compile:** Compiles HTML template to create AngularJS view with binding

**Execute:** Executes AngularJS view listening to DOM events for triggers

* 1. **Compile:** Identifies / matches directives in HTML DOM

**Sort:** Sorts directives based on priority and a combined link function is created

**Link:** Links specific instance of scope to template, registers listeners on DOM elements, sets up required $watch with the scope

* 1. **Bootstrap:** Loads angular application module, with all its dependent modules along with ngCore modules

**Link:** Links specific instance of scope to template, registers listeners on DOM elements, sets up required $watch with the scope

1. Which of the following ways is /are INCORRECT to create an AngularJS controller in the AngularJS module – “appModule” – dependent on components registered with following names – svc, $scope, $http, myValue, myConstant?
   1. var myController = appModule.controller(“myController”, [“svc”, “$scope”, “$http”, “myValue”, “myConstant”,

function(svc, $scope, $http, myValue, myConstant) {

//TODO define controller’s constructor body here

}]);

* 1. var MyController = function(svc, $scope, $http, myValue, myConstant) {

//TODO define controller’s constructor body here

};

MyController.$inject([“svc”, “$scope”, “$http”, “myValue”, “myConstant”]);

appModule.controller(“myController”,MyController);

* 1. var myController = appModule.controller(“myController”, function(svc, $scope, $http, myValue, myConstant) {

//TODO define controller’s constructor body here

};

* 1. var myController = appModule.controller(“myController”, function() {

var svc = appModule.lookup(“svc”);

var $scope = appModule.lookup(“$scope”);

var $http = appModule.lookup(“$http”);

var myValue = appModule.lookup(“myValue”);

var myConstant = appModule.lookup(“myConstant”);

//TODO define controller’s constructor body here

};

1. Which of the following statements are CORRECT about AngularJS scope?
   1. AngularJS creates a $rootScope when it encounters ngApp directive in the loaded HTML template.
   2. AngularJS scopes are hierarchical where in child scope inherit from its parent scope through prototype chain
   3. All AngularJS scopes including “isolated” scope inherits prototypically from its parent scope
   4. Scope provides **$apply** to watch model change and **$watch** to propagate model changes from outside “Angular realm”
2. Where all can we place AngularJS directive?
   1. Only elements
   2. Only attributes
   3. Only CSS styles
   4. Only Comments
   5. All of the above
3. How do you initialize / configure AngularJS $route module? Choose the wrong option from the following list
   1. //no need to download angular-route separately as it is part of ngCore

<script>

var appModule = angular.module("myapp",['ngRoute']);

appModule.map($route, “myURL1”,

“partials/myPartial1.html”, partial1Controller);

//TODO map remaining URLs

</script>

* 1. <script src="bower\_components/angular-route/angular-route.js"></script>

<script>

var appModule = angular.module("myapp",['ngRoute']);

appModule.config(['$routeProvider',

function($routeProvider) {

//TODO configure route with URL mapping

}]);

</script>

* 1. <script src="bower\_components/angular-route/angular-route.js"></script>

<script>

var appModule = angular.module("myapp",['ngRoute']);

$route.$inject(['$routeProvider',

function($routeProvider) {

//TODO configure route with URL mapping

}]);

</script>

* 1. <script src="bower\_components/angular-route/angular-route.js"></script>

<script>

var appModule = angular.module("myapp",['ngRoute']);

$route.$inject(['urlMappings', {“url1” : {“partial1.html” : “url1template.html”, “controller”: url1Controller},

“url2” : {“partial2.html” : “url2template.html”, “controller”: url2Controller}

}

]);

</script>

1. When do you use AngularJS filters?
   1. Sorting a list of javascript objects based on a property in the object
   2. Displaying a text in the selected language
   3. Convert <div> element to <span> element
   4. Filter DOM elements based on CSS selector
2. When is it recommended to use AngularJS provider?
   1. Reusable component that needs to be configured before it can be injected into other angular JS components
   2. Component to hold view state
   3. Component to interact with HTTP server
   4. All of the above0
3. Which AngularJS components provides APIs for AJAX and JSONP?
   1. $ajax
   2. $resource
   3. $jsonp
   4. $http
4. Which of the following options best describe Karma framework?
   1. Karma records user interactions on the browser and can be replayed
   2. Karma is a command-line javascript test runner in which any of the testing framework can be plugged in to execute test cases
   3. Karma is a testing framework that provides APIs to test Angular JS components
   4. Karma is a browser plugin to debug and test AngularJS application
5. How do you inject a custom Angular scope object into an Angular JS controller “MyController” using ngMock and Jasmine API inside beforeEach ?

var scope;

beforeEach(function(){

module('appModule');

inject(function($controller, $rootScope){

scope = $rootScope.$new();

$controller('MyController',{$scope:scope});

});

//TODO set state in scope

});

var scope;

beforeEach(function(){

module('appModule');

var myController =

angular.controller(“MyController”,

[“scope”, function($scope){

//TODO set state in scope

}]);

});

var scope;

beforeEach(function(){

module('appModule');

angular.controller(‘MyController’).

inject(function($rootScope){

scope = $rootScope.$new();

//TODO set state in scope

});

});

var scope;

beforeEach(function(){

module('appModule');

var myController =

angular.controller(“MyController”);

myController.$inject([“scope”]);

});