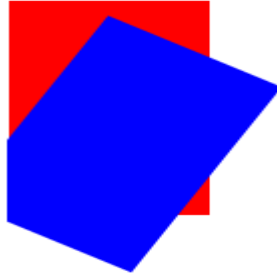


LAB NO.: 10

Date:

ADDITIONAL HTML, JAVASCRIPT QUESTIONS

1. Write the HTML5 program to display the following using canvas tag. Give different colour for the rectangles.



2. Write a JavaScript program to compute the sum and product of an array of integers. Take the array elements from user. (Use alert command)
3. Write a JavaScript conditional statement to sort any three numbers. Display an alert box to show the result.
4. Write a java script program to take some input from the user [string, integer and Boolean]. Based on the input, display the type of the input in the 'text box'.
5. Write the HTML5 program to draw a "smiling face" picture using the canvas tag.
6. Write the JavaScript program to add items into the array and delete items from array using two methods.
7. Write the java script program to display the grade [A, B,C,D] based on the marks entered by student(take the input into text boxes). Enter the marks of 4 subjects and calculate the average(using button). If the avg>90 then A, avg>80 then B, if avg>70 then C, if avg>60 then D.
8. Write the JavaScript program to show the below output.

☒ Red
☐ Green
☐ Blue



9. Write the HTML5 program to display following with validation to each field.

Name [take only capital letters]:
Password[show stars]:
Phone Number[10 digits only]:

10. Write a JavaScript program to perform an online quiz.

INTRODUCTION TO ANGULARJS

Objectives:

In this lab, student will be able to:

- 1. Learn about AngularJS**
- 2. Understand Model-View-Controller Architecture, module, controller**
- 3. Understand basic AngularJS directives**

11.1 INTRODUCTION

AngularJS is a Javascript framework that helps build web applications. Google is the company that developed AngularJS. AngularJS is an open source project which means it can be freely used and shared with others. There are many websites developed with AngularJS. Visit a website called <http://madewithangular.com> which lists some of the web sites developed using AngularJS.

To develop AngularJS applications there is only one script file required named angular.js

To download the file visit <http://angularjs.org>. Click on Download and give the Branch option as 1.7.x, Build as Uncompressed, and rest of the options as default. Click on Download. You will get a file called angular.js on the web page. Save the file as angular.js in your working directory.

Also on the same web site you will find several resources to learn AngularJS.

To develop AngularJS applications, there are two steps.

1. Include reference to the script angular.js
2. Include ng-app attribute somewhere within the HTML

11.1.1 First AngularJS Web Page:

Save the following as a file ending with .htm or .html and open it in a browser.

```
<html>
  <head>
    <script src="angular.js">
    </script>
  </head>

  <body ng-app>
    10+20={{ 10+20 }}
  </body>
</html>
```

In the above, expression within two flower bracket is called bounding expression.

11.1.2 Another example involving JavaScript object

```
<html>
  <head>
    <script src="angular.js">
    </script>
  </head>

  <body>
    <div ng-app>
      10+20={{ 10+20 }}
    </div>
    <div>
      40+50={{ 40+50 }}
    </div>
    <div ng-app>
      {{ 1==2 }}
    </div>
    <div ng-app>
      {{ {name:'MIT', age:61}.name }}
    </div>
  </body>
</html>
```

11.1.3 AngularJS example using JavaScript array

```
<html ng-app>

  <head>
    <script src="angular.js">
    </script>
  </head>

  <body ng-app>
    <div>
      {{ ['MIT','VIT','IIT'][2] }}
    </div>
  </body>
</html>
```

11.2 Module in AngularJS

A module is a container for different parts of an application namely controllers, services, directives, filters etc. A module is like a main() method in other types of applications. To create a module use the angular object's module() method. The angular object is provided in angular.js script.

```
var myApp = angular.module("myModule",[]);
```

The module method has two parameters. The first parameter specifies the name for the module and the second parameter specifies the dependencies for this module. The module myModule is not dependent on any other module, hence we are passing empty array for the second parameter.

11.3 Controller in AngularJS

A controller is a JavaScript function. The job of the controller is to build a model for the view to display. A Controller in AngularJs takes the data from the View, processes the data, and then sends that data across to the view which is displayed to the end user. The Controller will have the core business logic.

```
var myController=function($scope,$routeParams) {  
    $scope.message="AngularJS";  
}
```

Next step is to register the controller with the module.

```
myApp.controller("myController", ['$scope',myController]);
```

First is the name of the controller and second is the controller object.

Create a file called script1.js with the following contents.

```
var myApp = angular.module("myModule",[]);  
  
var myController=function($scope,$routeParams) {  
    $scope.message="AngularJS";  
}  
  
myApp.controller("myController", ['$scope',myController]);
```

Reference the script file script1.js in HTML. Mention the name of the module in ng-app directive. Also, mention the name of the controller in ng-controller directive.

```
<!doctype html>  
<html ng-app="myModule">  
<head>
```

```

        <script src="angular.js"></script>
        <script src="script1.js"></script>
</head>
<body>
    <div ng-controller="myController">
        First message is {{ message }}
    </div>
    <div>
        Second message is {{ message }}
    </div>
</body>
</html>

```

In the above, the second message is not displayed. If we move ng-controller directive to body element from the first div element, then both the message are displayed.

Open the above HTML in a browser. You will find Angular JS displayed in the first div element and no message in the second DIV element. Move the ng-controller attribute to body element. Then open the file. Now, you will find the message in each div element.

You can add objects to scope in the controller. You can refer the object in HTML. Edit the script1.js in the following way.

```

var myApp = angular.module("myModule",[]);

var myController = function($scope,$routeParams) {
    var employee = {
        firstName: "Rakesh",
        lastName : "Pawar",
        gender: "Male"};

    $scope.employee = employee;
}

myApp.controller("myController", ['$scope',myController]);

```

Create an HTML file named angobj.html with the following content.

```

<!doctype html>
<html ng-app="myModule">

    <head>
        <script src="angular.js"></script>
        <script src="script1.js"></script>

    </head>

```

```

<body ng-controller="myController">
  <div>
    First Name : {{ employee.firstName }}
  </div>

  <div>
    Last Name : {{ employee.lastName }}
  </div>
  <div>
    Gender : {{ employee.gender }}
  </div>
</body>
</html>

```

The Angularjs framework is built on Model-View-Controller pattern. This pattern is based on splitting the business logic layer, the data layer and the presentation layer into separate sections. The advantage of Angularjs is that you need not have to write lot of code to bind data to HTML controls. By writing small snippets of code it is possible to bind data. The Controller represents the layer that has the business logic. User events trigger the functions which are stored inside your controller. The user events are part of the controller. Views are used to represent the presentation layer which is provided to the end users. Models are used to represent your data. The data in your model can be as simple as just having primitive declarations.

11.4 \$scope object in Angularjs

The scope is a JavaScript object which basically binds the "controller" and the "view". One can define member variables in the scope within the controller which can then be accessed by the view.

Edit the following in script1.js.

```

var app = angular.module("DemoApp", []);
  app.controller("DemoController", ['$scope', function($scope,$routeParams) {

    $scope.fullName=function(firstName,lastname){
      return firstName + lastname;
    }

  }]);

```

Create a HTML file called scope.html with the following content.

```

<!doctype html>
<html ng-app="DemoApp">

    <head>
        <script src="angular.js"></script>
        <script src="script1.js"></script>
    </head>

    <body ng-controller="DemoController">
        <div ng-controller="DemoController">

            {{fullName("Kiran,"16")}}

        </div>

    </body>
</html>

```

11.5 ng-model Directive

ng-model is a directive in Angular.JS that represents models and its primary purpose is to bind the "view" to the "model".

Type the following in script1.js

```

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

app.controller('DemoCtrl', ['$scope',function($scope,$routeParams){
    $scope.pDescription="This topic looks at how Angular JS works \nModels in
Angular JS"}]);

```

Create an HTML file called ngmodel.html

```

<!doctype html>
<html>

    <head>
        <script src="angular.js"></script>
        <script src="script1.js"></script>

    </head>

```


[illegible]

11.5.2 Use of ng-model in select element

Type the following in script1.js

```
var app = angular.module('DemoApp',[]);
    app.controller('DemoCtrl', ['$scope',function($scope,$routeParams){
        $scope.pName="Edible Items";

        $scope.fruits =
        {
            option1 : "Apple",
            option2 : "Banana",
            option3 : "Chicku"
            option4 : "Mango"
        };
    }]);
```

Create a file called ngmodel3.html

[illegible]

11.6.3 number filter

```
<div ng-app="DemoApp" ng-controller="DemoController">
```

```
    This amount is {{ amount | number:2 }}
```

```
</div>
```

```
var app = angular.module('DemoApp',[]);
app.controller('DemoController',['$scope',function($scope,$routeParams){

    $scope.amount =3.565656;
}]);
```

In the above, 2 indicates fraction size.

11.6.4 currency filter

This filter formats a currency filter to a number.

Suppose, if you wanted to display a number with a currency such as \$, then this filter can be used.

```
<div ng-app="DemoApp" ng-controller="DemoController">
```

```
    The amount is {{ amount | currency:"USD$":0 }}
    The amount is {{ amount | currency:'Rs.':2 }}
```

```
</div>
```

```
var app = angular.module('DemoApp',[]);
app.controller('DemoController',['$scope',function($scope,$routeParams){

    $scope.amount =20.5614;
}]);
```

In the above the second expression after currency filter indicates the currency symbol(default is \$) and the third indicates number of fractional digits.

11.6.5 json filter

This filter formats a JSON(JavaScript Object Notation) like input and applies the JSON filter to give the output in JSON.

In the below example we will use a controller to send a JSON type object to a view via the scope object. We will then use a filter in the view to apply the JSON filter.

```
<div ng-app="DemoApp" ng-controller="DemoController">

    This tutorial is {{tutorial | json}}

</div>
<script type="text/javascript">
    var app = angular.module('DemoApp',[]);
    app.controller('DemoController',['$scope',function($scope,$routeParams){

        $scope.tutorial={ TutorialID:12,tutorialName:"Angular" };
    }]);
</script>
```

Lab Exercises :

1. Develop an AngularJS Application to display the text CSE branch, MIT Manipal with H3 tag.
2. Develop an AngularJS Application which has two textboxes for First Name and Last Name. A hint should be displayed on the two textboxes containing "Enter First Name" and "Enter Last Name". As soon as some content is entered in the two textboxes it should be displayed below the two textboxes.
3. Develop an AngularJS Application which has two textboxes. The first textbox is for price and the second one for quantity. Write an AngularJS function in the scope object which takes two numbers and returns the product of the two numbers. As soon as the textboxes are altered, display the product of the two numbers below the second textbox.
4. Develop an AngularJS Application which has an object named Months. Store all the months in the Months object. Display the months in a listbox.
5. Display a string "QwErTy" in an AngularJS application in all lowercase and uppercase using filters.
6. Display the value of PI(assume 22/7) to 2 and 4 decimal places. Use filters.
7. Display a JSON expression in JSON format. Use filters.

Additional Exercises :

1. Create an Array named leapYears in an AngularJS application and store all the leap years starting from 2000 to 2030. Display the leap years in a list box.
2. Develop an AngularJS Application which has a variable in \$scope named count. Initialize the count to 5. Display count number of textboxes in the web page.