

Ex. No. 8	Web Application Development using Angular JS Framework
Date of Exercise	11/10/22
Web Host Link	
Video Link	https://drive.google.com/file/d/1kRKtwPB4S8zIXBcQDbTiXjn96p4mjSTD/view?usp=share_link

Aim:

To create a web application to develop the following Budget Calculator using AngularJS. Here, the room rent, accessories, emergency and saving get as percentage (%), then divide the given amount based on the percentage and display each value in the place provided when the “Evaluate My Money” button is clicked (use ng-click directive).

Description:**AngularJS Architecture**

AngularJS is a Single Page Application (SPA) meta-framework. Angular JS follows the MVC architecture. MVC stands for Model-View-Controller evolved as a way to separate data, logical units and presentation in web application development.

- The model is the driving force of the application. This is generally the data behind the application, usually fetched from the server.
- The view is the UI that the user sees and interacts with. It is dynamic, and generated based on the current model of the application.
- The controller is the business logic and presentation layer, which performs actions such as fetching data and makes decisions such as how to present the model, which parts of it to display, etc.

AngularJS Extends HTML

- AngularJS extends HTML with ng-directives.
- The ng-app directive defines an AngularJS application.
- The ng-init directive used to create initial value(model) for the angular JS application.
- The ng-model directive binds the value of HTML controls (input, select, textarea) to application data.
- The ng-bind directive binds application data to the HTML view.

Question Set A – Working with Angular JS Directives

Program :

```
<!DOCTYPE html>
<html lang="en">

<head>
  <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.9/angular.min.js"></scri
pt >

    <script>

        var app=angular.module("myApp",[[]]);//module
        app.controller("control",function($scope){
            $scope.evaluate = function () {
                var tot = parseInt($scope.num);
                var num1 = parseInt($scope.n1);
                var num2 = parseInt($scope.n2);
                var num3 = parseInt($scope.n3);
                var num4 = parseInt($scope.n4);
                var r1 = tot * num1 / 100;
                $scope.r1 = "Room Rent " + (r1);
```



```
style="border:1px solid rgb(40, 40,86);background-color:rgb(40, 40,
86);font-size:20px;color:white;text-align:center;height:50px;margin-left:41%">

    Evaluate My Money</button>
<br><br><br>
<div id="box" style="margin-left:5%">
    <div style="margin-left: 15%;float:
left;;width:100px;height:70px;border:1px solid blue; background-
color:blue;color:white;text-align:center">{{r1}}</div>

    <div style="margin-left: 10%;float:
left;width:100px;height:70px;border:1px solid green; background-
color:green;color:white;text-align:center">{{r2}}</div>

    <div style="margin-left: 10%;float:
left;width:100px;height:70px;border:1px solid yellow; background-
color:yellow;color:black;text-align:center">{{r3}}
    </div>
    <div style="margin-left: 10%;float:
left;width:100px;height:70px;border:1px
solid pink; background-color:pink;color:black;text-align:center">{{r4}}</div>
    </div>
</body>
</script>
</html>
```

Output :

The screenshot shows a web browser window with the address bar displaying "C:/web%20technology%20lab/lab/ex%208a.html". The page title is "Budget Calculator App". The form contains the following elements:

- A heading: "Enter Amount That You Want To Manage"
- A text input field with the placeholder text "e.g.3000".
- A heading: "How Would You Like To Divide Your Money?"
- Four input fields with labels: "Room Rent(%)", "Accessories(%)", "Emergency(%)", and "Saving(%)". Each field has the placeholder text "e.g.3000".
- A dark blue button labeled "Evaluate My Money".
- Four colored squares (blue, green, yellow, pink) representing the output categories.

If we enter the values ,then the output will be:

The screenshot shows the same web browser window, but now the form is populated with values and the results are displayed:

- The "Enter Amount That You Want To Manage" input field contains the value "70000".
- The "How Would You Like To Divide Your Money?" section has four input fields with the following values: "Room Rent(%)": "2000", "Accessories(%)": "400", "Emergency(%)": "5000", and "Saving(%)": "60".
- The "Evaluate My Money" button is still present.
- Below the button, four colored squares display the calculated results:
 - Blue square: "Room Rent 1400000"
 - Green square: "Accessories 280000"
 - Yellow square: "Emergencies 3500000"
 - Pink square: "Savings 42000"

Question Set B – Working with Arrays in Angular JS

Aim:

Create a HTML single page web application to develop product inventory system as per your choice using Angular JS Framework.

Program :

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.9/angular.min.js"></scri
pt>
  <script>
    var app = angular.module("App", []);
    app.controller("control", function ($scope) {
      $scope.items = [];
      $scope.addItem = function () {
        $scope.items.push({ "N": $scope.itemname, "P": $scope.itemprice
});
      };
      $scope.remove = function ($index) {
        $scope.items.splice($index, 1);
      };
    });
  </script>
</head>

<body ng-app="App" ng-controller="control">
  <p style="color:blue;font-size: large;"><b>Product Inventory
System</b></p>

  <p style="color:blue">Enter the Product Name</p><input type="text" ng-
model="itemname"
  placeholder="enter product name"><br>
```

```
<p style="color:blue">Enter the Product Price</p><input type="text" ng-  
model="itemprice"  
  placeholder="enter product price"><br><br>  
  
<button ng-click="addItem()">Add Product</button>  
<hr>  
<br><br>  
<br><br>  
<table border="1" cellpadding="0" width="50%">  
  <tr style="background-color:green;color:white;">  
    <th>S.No</th>  
    <th>Name</th>  
    <th>Price</th>  
    <th>Remove</th>  
  </tr>  
  
  <tr ng-repeat="x in items" class="table w-100 mx-auto mt-5 table-  
bordered table-hover text-center">  
  
    <td>{{$index + 1}}</td>  
    <td>{{x.N}}</td>  
    <td>{{x.P}}</td>  
    <td><button ng-click="remove($index)"  
style="color:red;">[delete]</button></td>  
  </tr>  
</table>  
</body>  
  
</html>
```

Output:

Product Inventory System

Enter the Product Name

Enter the Product Price

S.No	Name	Price	Remove
------	------	-------	--------

Product Inventory System

Enter the Product Name

Enter the Product Price

S.No	Name	Price	Remove
1			<input type="button" value="[delete]"/>
2			<input type="button" value="[delete]"/>
3			<input type="button" value="[delete]"/>
4			<input type="button" value="[delete]"/>
5			<input type="button" value="[delete]"/>
6			<input type="button" value="[delete]"/>

Result:

Thus, The web page has been created successfully using HTML, CSS and Angular JS.