1.Write an AngularJS program to create a controller to access to variables one from scope and another from root scope.

```
<html>
    <head>
        Controlling Variables
    </head>
    <script src="angular.js"></script>
    <body ng-app="myapp">
    <div  ng-controller="myctrl">
    {{obj1}}
    </div>
    <div ng-controller="myctrl2">
        {{obj2}}
    </div>
    </body>
    <script>
        var app=angular.module("myapp",[]);
        app.controller("myctrl",function($scope,$rootScope){
            $scope.obj1="Controller1_object";
        });
        app.controller("myctrl2",function($scope,$rootScope){
            $rootScope.obj2="Controller2_object";
        });
    </script>
</html>
```

Output:

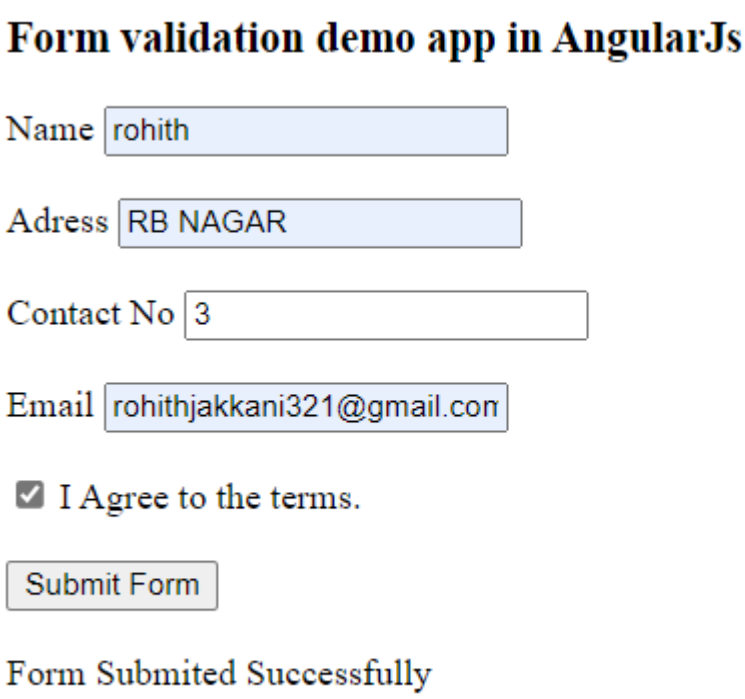
Controlling Variables
Controller1_object
Controller2_object

2. AngularJS application to demonstrate form validation.

```
<!DOCTYPE html>
<head>
<title>AngularJs Form Input Fields Validation Example</title>
<script
src="http://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js">
</script>
<script>
var app = angular.module('formApp', []);
app.controller('formCtrl', function ($scope) {
$scope.sendForm = function () {
$scope.msg='Form Submited Successfully';
};
});
```

```
</script>
<style>
    .valid.false {
    background: red;  }
    .valid.true {
    background: green; }
    .error {
    color: red; }
    </style>
</head>
<body ng-app="formApp" ng-controller="formCtrl">
<h3>Form validation demo app in AngularJs</h3>
<form name="personForm" ng-submit="sendForm()">
    <label for="name">Name</label>
    <input id="name" name="name" type="text" ng-model="person.name" required
/>
<span class="error" ng-show="personForm.name.$error.required"> Required!
</span>
<br /><br />
<label for="adress">Adress</label>
<input id="address" name="address" type="text" ng-model="person.address"
required />
<span class="error" ng-show="personForm.address.$error.required"> Required!
</span>
<br /><br />
<label for="contact">Contact No</label>
<input id="mobile" name="mobile" type="number" ng-model="person.mobile"
required />
<span class="error" ng-show="personForm.mobile.$error.required">Required
number!</span>
<span class="error" ng-show="personForm.mobile.$error.mobile">Invalid
mobile!</span>
<br /><br />
<label for="email">Email</label>
<input id="email" name="email" type="email" ng-model="person.email" required
/>
<span class="error" ng-
show="personForm.email.$error.required">Required!</span>
<span class="error" ng-show="personForm.email.$error.email">Invalid
Email!</span>
<br /><br />
<input type="checkbox" ng-model="terms" name="terms" id="terms" required />
<label for="terms">I Agree to the terms.</label>
<span class="error" ng-show="personForm.terms.$error.required">You must agree
to the terms</span>
<br /><br />
<button type="submit">Submit Form</button>
<br /><br />
```

```
<span>{{msg}}</span>
</form>
</body>
</html>
```



3. Create a module for multiplication in java script and by including that module in angularJS find the multiplication of two numbers?

```
<html>
<head>
    <title>Tables</title>
    <script src="angular.js"></script>
</head>
<body ng-app="tablesApp">
    <div ng-controller="appController">
        Count: <input id="Text1" type="number" min="0" ng-model="count"/>
        Number: <input id="Text2" type="number" min="1" ng-model="num"/><br/>
        Value Of Count entered<div>{{count}}</div><br />
        Value Of Number entered<div>{{num}}</div><br />
        <div ng-repeat="n in eval(count,num) track by $index">
            {{num}}*{{Counter1(n,num)}}={{n}}-----{{$index+1}}
        </div>
    </div>
</body>
<script type="text/javascript">
    var app = angular.module("tablesApp", []);
```

```
    app.controller("appController", function ($scope) {
        $scope.count = 12;
        $scope.num = 9;
        $scope.eval = function (count, num) {
            $scope.arr = [];
            for (var i = 1; i<= count; i++) {
                $scope.arr.push(num * i);
            }
            return $scope.arr;
        };
        $scope.Counter1 = function (i, number) {
            return i/number;
        };
    }
    );
</script>
</html>
```

Count: 12 Number: 9

Value Of Count entered
12

Value Of Number entered
9

9*1=9-----1
9*2=18-----2
9*3=27-----3
9*4=36-----4
9*5=45-----5
9*6=54-----6
9*7=63-----7
9*8=72-----8
9*9=81-----9
9*10=90-----10
9*11=99-----11
9*12=108-----12

4. Write an AngularJS program to validate a form with input fields username, contact number and email. Validate the form using built in validation directives.

5. AngularJS application to demonstrate custom form validation.

```
<!DOCTYPE html>
<script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.9/angular.min.js">
</script>
<body ng-app="developapp">
<form name="form1">
Username: <input name="username" required><br><br>
Age: <input name="userage" ng-model="userage" required app-directive>
</form>
<p>The input's valid state is:</p>
<h1>{{form1.userage.$valid}}</h1>
<script>
var app = angular.module('developapp', []);
app.directive('appDirective', function() {
	return {
		require: 'ngModel',
		link: function(scope, element, attr, mCtrl) {
			function myValidation(value) {
				if (value >=18) {
					mCtrl.$setValidity('charE', true);
				} else {
					mCtrl.$setValidity('charE', false);
				}
				return value;
			}
			mCtrl.$parsers.push(myValidation);
		}
	};
});  </script>
<p>The input field must have user age greater than 18 to be considered valid for
voting.</p>
</body>
</html>
```

6. Create a module for division in java script and by including that module in angularJS find the division of two numbers?

```
<html>
```

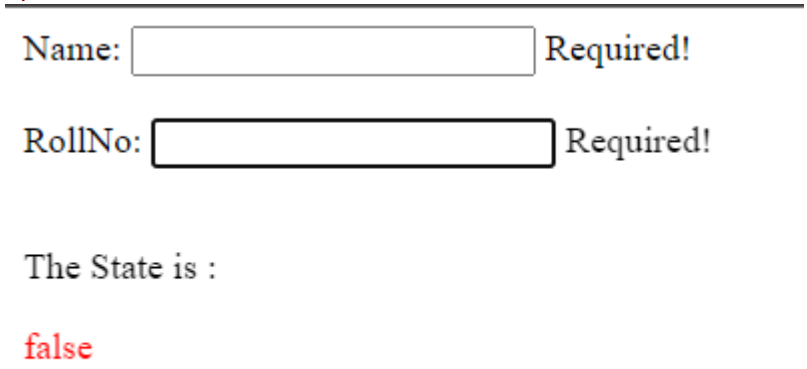
```
    <script src="angular.js"></script>
    <head>DIVISION</head>
    <body ng-app="myapp">
        <div ng-controller="myctrl">
            {{res(50,10)}}
        </div>
    </body>
    <script>
        var app=angular.module("myapp",[]);
        app.controller("myctrl",function($scope){
            $scope.dividend=100;
            $scope.divisor=10;
            $scope.res=function(dividend,divisor){
                return dividend/divisor;
            }
        });
    </script>
</html>
```

DIVISION
5

7. Write an AngularJS program to validate an input field for student roll no, the filed must contain the string “19881A”. Perform this custom validation using directives provided.

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>Studen Roll No</title>
    <script src="angular.js"></script>
</head>
<body>
<div ng-app='myapp' >
    <form name="form">
        Name: <input type="text" name="name" ng-model='name' required>
        <span ng-show='form.name.$error.required'>Required!</span><br/><br/>

        RollNo: <input type="text" name="roll" ng-model='roll' required my-
validator>
        <span ng-show='form.roll.$error.required'>Required!</span><br/><br/>



    </form>
        <p>The State is : </p>
```

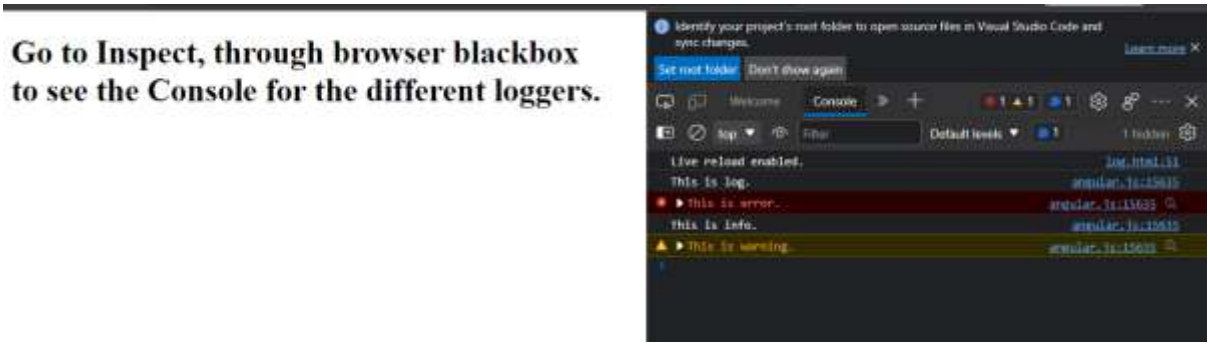
```
        <span style="color:red">{{form.roll.$valid}}</span>

    <script type="text/javascript">
        var app = angular.module('myapp',[])
        app.directive('myValidator',function(){
            return{
                require:'ngModel',
                link:function(scope,element,attr,mCtrl){
                    function myValidation(value){
                        if(value.indexOf("19881A")>-1){
                            mCtrl.$setValidity("charE",true)
                        }else{
                            mCtrl.$setValidity("charE",false)
                        }
                        return value
                    }
                    mCtrl.$parsers.push(myValidation)
                }
            }
        })
    </script>
</div>
</body>
</html>
```



8. AngularJS application to access data from server using $log service.

```
<!DOCTYPE html>
<html>
<head>
<title>Logging Example in AngularJS</title>
<script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.0/angular.min.js"></s
cript>
```

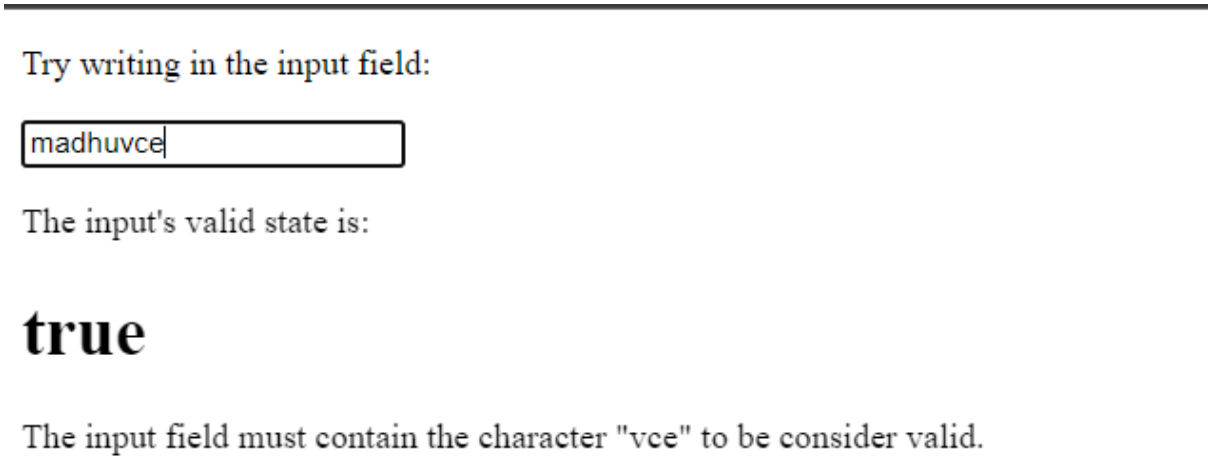
```
<script>
var app = angular.module('myApp', [])
app.controller("myController", function ($log) {
            $log.log('This is log.');
            $log.error('This is error.');
            $log.info('This is info.');
            $log.warn('This is warning.');
            $log.debug('This is debugging.');
        });
</script>
</head>
<body ng-app="myApp">
<form id="form1">
<div ng-controller="myController">
<p> <h1> Go to Inspect, through browser blackbox <br/>
to see the Console for the different loggers.
</form>
</body>
</html>
```



9. Write an AngularJS program to validate an input field for Employee ID, the filed must contain the string “VCE”. Perform this custom validation using directives provided.

```
<!DOCTYPE html>
<html>
<script
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.9/angular.min.js"></s
cript>
<body ng-app="myApp">

<p>Try writing in the input field:</p>

<form name="myForm">
<input name="myInput" ng-model="myInput" required my-directive>
</form>

<p>The input's valid state is:</p>
<h1>{{myForm.myInput.$valid}}</h1>
```

```
<script>
var app = angular.module('myApp', []);
app.directive('myDirective', function() {
    return {
        require: 'ngModel',
        link: function(scope, element, attr, mCtrl) {
            function myValidation(value) {
                if (value.indexOf("vce") > -1) {
                    mCtrl.$setValidity('charE', true);
                } else {
                    mCtrl.$setValidity('charE', false);
                }
                return value;
            }
            mCtrl.$parsers.push(myValidation);
        }
    };
});
</script>

<p>The input field must contain the character "vce" to be consider valid.</p>

</body>
</html>
```



10. Create an AngularJS program to display the table with different styles for even and odd rows.

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

```
        <tr><thead><td>S.No</td><td>NAME</td><td>AGE</td><td>SALARY</td></thea
d></tr>
        <tr><td>1</td><td>J.Rohith</td><td>20</td><td>100000</td></tr>
        <tr><td>2</td><td>J.Ankith</td><td>18</td><td>200000</td></tr>
        <tr><td>3</td><td>J.Sathwika</td><td>16</td><td>300000</td></tr>
        <tr><td>4</td><td>J.Satyanarayana</td><td>50</td><td>400000</td></tr>
    </table>
    <style>
        table,th,td{
        border: 2cm;
        border-collapse: collapse;
        padding: 5mm;
        }
        table tr:nth-child(odd){
            background-color: antiquewhite;
        }
        table tr:nth-child(even){
            background-color: bisque;
        }
    </style>
</html>
```

S.No	NAME	AGE	SALARY
1	J.Rohith	20	100000
2	J.Ankith	18	200000
3	J.Sathwika	16	300000
4	J.Satyanarayana	50	400000

11. AngularJS application to demonstrate $scope Life Cycle

```
<html>
<title>
life cycle of angular js
</title>
<script src="angular.js">
</script>
```

```
<body ng-app="myapp">
<div ng-controller="myctrl">
{{username}}
</div>
<script>
var app=angular.module("myapp",[]);
app.controller("myctrl",function($scope,$timeout){
$scope.username="Rohith";
setTimeout(function(){
$scope.username="Raju";
alert("updated variable"+$scope.username);
$scope.$apply();
$scope.$digest();
},2000)
});
</script>
</body>
</html>
```

12. AngularJS application to demonstrate the limitto filter

```
<!DOCTYPE html>
<html>
    <script src="angular.js"></script>
    <body ng-app="myapp">
        <div ng-controller="myctrl">
            <p>limit upto first 3 numbers</p>
            <ul>
            <li ng-repeat="x in names|limitTo:3" >
                {{x}}</li>
            </ul>
            <p>limit upto last 3 numbers</p>
            <ul>
                <li ng-repeat="x in names|limitTo:-3" >
                {{x}}</li>
                </ul>
    <script>
        var app=angular.module("myapp",[]);
        app.controller("myctrl",function($scope){
            $scope.names=["rohith","mani","himavanth","raju","rani","rama"]
        });
    </script>
    </body>
</html>
```

13. Write an AngularJS program to handle mouse and key events.

```
<html>
    <script src="angular.js"></script>
    <div ng-app="myapp" ng-ctrl="myctrl">
        <h1>These are the mouse events </h1>
        <h1 ng-mousemove="count=count+1">Mouse Move</h1>
        <h2>The remaining count is</h2>
        <h3>{{count}}</h3>
        <h1 ng-mouseover="count1=count1+1">Mouse Over</h1>
        <h2>the updated count is</h2>
        <h3>{{count1}}</h3>
        <h1 ng-mouseenter="count2=count2+1">Mouse Enter</h1>
        <h2>the updated count is</h2>
        <h3>{{count2}}</h3>
        <p ng-mouseleave="count3=count3+1">Mouse Leave</p>
        <h2>the update count is </h2>
        <h3>{{count3}}</h3>
        <input ng-keyup="count4=count4+1">
        <p>updated count for keyup is{{count4}}</p>
        <input ng-keypress="count5=count5+1">
        <p>updated count for keypressed is{{count5}}</p>
        <input ng-keydown="count6=count6+1">
        <p>updated count for keydown is{{count6}}</p>
    </div>
    <script>
        var app=angular.module("myapp",[]);
        app.controller("myctrl",function($scope){
            $scope.count=0;
            $scope.count1=0;
            $scope.count2=0;
            $scope.count3=0;
            $scope.count4=0;
            $scope.count5=0;
            $scope.count6=0;
        });
    </script>
</html>
```

These are the mouse events

Mouse Move

The remaining count is

18

Mouse Over

the updated count is

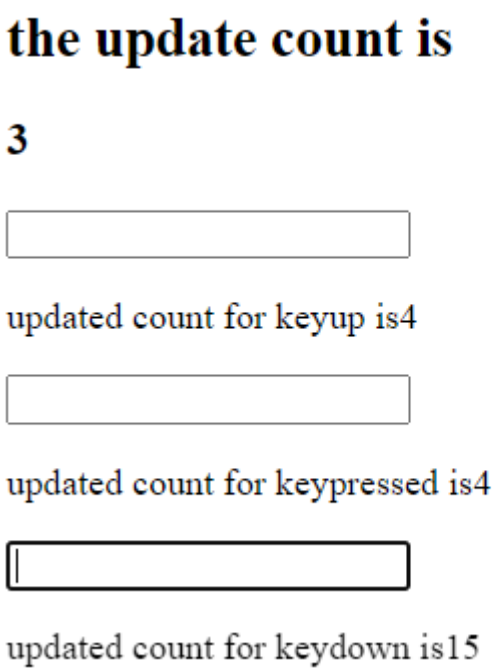
5

Mouse Enter

the updated count is

4

Mouse Leave



14. Write an AngularJS program to access student data from webserver using $http service.

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>Http server Data</title>
    <script
src="http://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></sc
ript>
</head>
<body >
<table ng-app='myapp' ng-controller='myctrl'>
    <tr>
        <th>Name</th><th>Age</th><th>Salary Rating</th>
    </tr>

        <tr ng-repeat='emp in Data'>
            <td>{{emp.name}}</td>
            <td>{{emp.age}}</td>
            <td>{{emp.sal }}</td>
        </tr>
</table>
<script type="text/javascript">
    var app= angular.module("myapp",[])
    app.controller("myctrl",function ($scope,$http){
        $http.get('data.txt').success(function(response){
```

```
                $scope.Data=response
                console.log(response)
            })
        })

    </script>

    </body>
    </html>
```

15. AngularJS application using custom filter to assig grade ‘B’ if salary is less than or equal to 15000 ,otherwise grade ‘A’ in salary table.

```
<html>
  <head>
    <title>AngularJS Custom Filter Application</title>
    <script src = "angular.min.js">
    </script>
  </head>
  <body ng-app = "myApp" ng-controller="myController">
    <h1>Employee Details of VCE before filter</h1>
    <table border=1>
        <tr>
        <th> Name of the Employee</th>
        <th> Gender</th>
        <th> Salary</th>
        </tr>
        <tr ng-repeat="x in employee">
        <td>{{x.name}}</td>
        <td>{{x.gender}}</td>
        <td>{{x.salary}}</td>
        </tr>
   </table>
 <h1>Employee Details of VCE After applying Grade filter</h1>
    <table border=1>
        <tr>
        <th> Name of the Employee</th>
```

```
        <th> Gender</th>
        <th> Salary</th>
        </tr>
        <tr ng-repeat="x in employee">
        <td>{{x.name}}</td>
        <td>{{x.gender}}</td>
        <td>{{x.salary | salary}}</td>
        </tr>
    </table>
<script>
var app=angular.module("myApp",[]);
app.filter("salary",function(){
        return function(salary){
                if(salary>=20000){
                return "A";
                }
                else{
                return "B";
                }
        }
});
app.controller("myController",function($scope){
$scope.employee=[
{name:"ganesh",gender:"male",salary:55000},
{name:"sowmya",gender:"female",salary:15000},
{name:"siva",gender:"male",salary:30000},
{name:"gayathri",gender:"female",salary:15000}
];
});
</script>   </body></html>
```