7. Write an AngularJS program to create a simple CRUD

application (Create, Read, Update, and Delete) for managing users.

```
//Code
<!DOCTYPE html>
<html>
<title>USER MANAGEMENT APPLICATION</title>
<head>
<script type="text/javascript"
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min
.js">
</script>
<script>
var app=angular.module("userMgmtApp",[]);
app.controller("userMgmtAppCntrl",function($scope){
$scope.users=[
{'name':"Mahesh Kumar",
'email':'maheshkumar@gmail.com','editing':false},
{'name':'ABC','email':'abc@gmail.com','editing':false},
{'name':'XYZ','email':'xyz@gmail.com','editing':false}
]
$scope.createUser=function()
{
if($scope.newUserName && $scope.newUserEmail)
{
var u={
'name':$scope.newUserName,
'email':$scope.newUserEmail,
'editing':false
}
$scope.users.push(u)
$scope.newUserName=''
$scope.newUserEmail=''
```

```
}
else{
alert("Please provide the user name and email id")
}
}
$scope.readUser=function(user)
{
user.editing=true
}
$scope.updateUser=function(user){
user.editing=false
}
$scope.deleteUser=function(user)
{
var yes=confirm("Are you sure you want to delete")
if(yes==true)
{
var index=$scope.users.indexOf(user)
$scope.users.splice(index,1)
}
}
});
</script>
</head>
<body ng-app="userMgmtApp">
<h1>USER MANAGEMENT APPLICATION</h1>
<div ng-controller="userMgmtAppCntrl">
Enter the User Name:<input type="text" ng-model="newUserName">
Enter the User Email:<input type="text" ng-model="newUserEmail">
<button ng-click="createUser()">Create</button>
<br/>
<br/>
```

```html
<table border="1">
<tr>
<th>SLNO</th>
<th>NAME</th>
<th>EMAIL</th>
<th>READ</th>
<th>UPDATE</th>
<th>DELETE</th>
</tr>
<tr ng-repeat="user in users">
<td>{{$index+1}}</td>
<td>
<span
ng-show="!user.editing">{{user.name}}</span>
<input type="text" ng-show="user.editing" ng-model="user.name">
</td>
<td>
<span ng-show="!user.editing">{{user.email}}</span>
<input type="text" ng-show="user.editing" ng-model="user.email">
</td>
<td>
<button ng-click="readUser(user)">Read</button>
</td>
<td>
<button ng-click="updateUser(user)">Update</button>
</td>
<td>
<button ng-click="deleteUser(user)">Delete</button>
</td>
</tr>
</table>
</div>
```

```
</body>

</html>

8. Develop AngularJS program to create a login form, with

validation for the username and password fields.

//Code
<!DOCTYPE html>

<html>

<title>Angular JS Login Form</title>

<head>

<script type="text/javascript"

src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min

.js">

</script>

<script>

var app=angular.module("loginApp",[]);

app.controller('loginAppCntrl',function($scope){

$scope.userName=''

$scope.password=''

$scope.noAttempts=0

$scope.login=function(){

// console.log("Inside login function")

if($scope.userName=="harsha" && $scope.password=="123456")

{

alert("Login Successfull")

}

else{

$scope.noAttempts++

if($scope.noAttempts<=3)

{

alert("Incorrect user name/password! Attempt No. "+$scope.noAttempts)

}

else{
```

```
document.getElementById("loginButton").disabled=true

}

}

}

});
```

```html
</script>

<style>

.error-message{

color:red;

font-size: 20px;

}

</style>

</head>

<body ng-app="loginApp" ng-controller="loginAppCntrl">

<h1>Angular JS Login Form</h1>

<form name="loginForm" ng-submit="submitForm()">

Enter the User Name:<input type="text" name="userName"

ng-model="userName" ng-minlength="5" ng-maxlength="8" required

placeholder="Enter

User Name">

<span class="error-message"

ng-show="loginForm.userName.$error.required &&

loginForm.userName.$dirty">User

Name is Required</span>

<span class="error-message"

ng-show="loginForm.userName.$error.minlength">Minimum Length Must be

5</span>

<span class="error-message"

ng-show="loginForm.userName.$error.maxlength">Maximum user name length

is limitted

to 8</span>

<br/>
```

```
<br/>

Enter the Password: <input type="password" name="password"

ng-model="password" ng-minlength="5" ng-maxlength="8" required

placeholder="Enter

your password">

<span class="error-message" ngshow="loginForm.password.$error.required

&& loginForm.password.$dirty">Password is required</span>

<span class="error-message"

ng-show="loginForm.password.$error.minlength">Minimum Password length

is 5</span>

<span class="error-message"

ng-show="loginForm.password.$error.maxlength">Maximum password length

is limitted

to 8</span>

<br/>

<br/>

<button type="submit" ng-disabled="loginForm.$invalid"

ng-click="login()" id="loginButton">Login</button>

</form>

</body>

</html>
```

9. Create an Angular JS application that displays a list of employees and their

salaries. Allow users to search for employees by name and salary. Note:

Employee details may be included in the program.

//Code

```
<!DOCTYPE html>

<html>

<title>Angular JS Filter Employee Search Application</title>

<head>

<script type="text/javascript"

src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.mi

n.js">
```

```
</script>

<script>

var app=angular.module("empSearchApp",[]);

app.controller("empSearchAppCntrl",function($scope){

$scope.empList=[

{'name':'Aditya Kumar','salary':500000},

{'name':'Chetan','salary':400000},

{'name':'Manju','salary':300000},

{'name':'Prashanth','salary':400000},

{'name':'Thanuja','salary':500000},

{'name':'Manasa','salary':600000}

]

$scope.clearFilters=function()

{

$scope.searchName=''

$scope.searchSalary=''

}

});

</script>

</head>

<body ng-app="empSearchApp">

<h1>Employee Search Application</h1>

<div ng-controller="empSearchAppCntrl">

Search by Employee Name:<input type="text" ng-model="searchName">

Search by Employee salary:<input type="number"

ng-model="searchSalary">

<button ng-click="clearFilters()">Clear Filters</button>

<br/>

<h3>List of Employees</h3>

<table border="1">

<tr>

<th>SLNO</th>
```

```html
<th>EMP NAME</th>

<th>SALARY</th>

</tr>

<tr ng-repeat="emp in empList |

filter:{name:searchName,salary:searchSalary}">

<td>{{$index+1}}</td>

<td>{{emp.name}}</td>

<td>{{emp.salary}}</td>

</tr>

</table>

</div>

</body>

</html>
```

10.Create Angular JS application that allows users to maintain a collection of items. The application should display the current total number of items, and this count should automatically update as items are added or removed. Users should be able to add items to the collection and remove them as needed.

Note: The default values for items may be included in the program.

```html
//Code
<!DOCTYPE html>

<html>

<title>Item Management Application</title>

<head>

<script type="text/javascript"

src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js">

</script>

<script>

var app=angular.module("itemMgmtApp",[]);

app.controller("itemMgmtAppCntrl",function($scope){

$scope.itemList=['Pen','Pencil','Eraser','Book']

$scope.addItem=function()

{
```

```
if($scope.newItem)

{

if($scope.itemList.indexOf($scope.newItem)==-1)

{

$scope.itemList.push($scope.newItem)

}

else{

alert('This item is already there in the item collection')

}

}

else{

alert('Please Enter the item to add')

}

}

$scope.removeItem=function(item)

{

var yes=confirm("Are you sure you want to delete "+item)

if(yes==true)

{

var index=$scope.itemList.indexOf(item)

$scope.itemList.splice(index,1)

}

}

});

</script>

</head>

<body ng-app="itemMgmtApp">

<h1>Item Management Application</h1>

<div ng-controller="itemMgmtAppCntrl">

Enter an item to add: <input type="text" ng-model="newItem">

<button ng-click="addItem()">ADD</button>

<br/><br/>
```

```html
<b>List of Items</b>

<table border="1">

<tr>

<th>SLNO</th>

<th>Item</th>

<th>Remove</th>

</tr>

<tr ng-repeat="item in itemList">

<td>{{$index+1}}</td>

<td>{{item}}</td>

<td><button ng-click=removeItem(item)>Remove</button></td>

</tr>

</table>

<br/>

Total Number of Items=<b>{{itemList.length}}</b>

</div>

</body>

</html>
```

11.Create AngularJS application to convert student details to Uppercase using angular filters. Note: The default details of students may be included in the program.

```html
//Code

<!DOCTYPE html>

<html>

<title>Student Details in Uppercase</title>

<head>

<script type="text/javascript"

src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js">

</script>

<script>

var app=angular.module("studDetailsUpperApp",[]);
```

```
app.controller("studDetailsUpperAppCntrl",function($scope){

$scope.studDetails=['harish','kumar','chetan','prashanth','thanuja']

$scope.upper=true

$scope.lower=false

$scope.Lower=function()

{

//console.log('called')

$scope.upper=false

$scope.lower=true

}

$scope.Upper=function()

{

$scope.upper=true

$scope.lower=false

}

});
</script>
</head>
<body ng-app="studDetailsUpperApp">
<h1>Student Details in Uppercase</h1>
<div ng-controller="studDetailsUpperAppCntrl">
<button ng-click="Upper()">Upper</button>
<button ng-click="Lower()">Lower</button>
<table border="1">
<tr>
<th>SLNO</th>
<th>NAME</th>
</tr>
<tr ng-repeat="student in studDetails">
<td>{{$index+1}}</td>
<td ng-show="upper">{{student|uppercase}}</td>
<td ng-show="lower">{{student|lowercase}}</td>
```

```
</tr>
</table>
</div>
</body>
</html>
```

12.Create an AngularJS application that displays the date by using date filter parameters.

```
//Code
<!DOCTYPE html>
<html>
<title>Date Application</title>
<head>
<script type="text/javascript"
src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js">
</script>
<script>
var app=angular.module("dateApp",[]);
app.controller("dateAppCntrl",function($scope){
$scope.currentDate=new Date();
});
</script>
</head>
<body ng-app="dateApp">
<h1>Date in different formats</h1>
<div ng-controller="dateAppCntrl">
Current Date and Time: {{currentDate}}<br/>
Short Date: {{currentDate|date: 'short'}}<br/>
Long Date: {{currentDate |date: 'fullDate'}}<br/>
Medium Date:{{currentDate| date: 'medium'}}
</div>
</body>
</html>
```