

# **Introduction**

## **6) AngularJS:**

**AngularJS** is an open-source web application framework. It was originally developed in 2009 by Misko Hevery and Adam Abrons. It is now maintained by Google. Its latest version is 1.2.21.

Definition of AngularJS as put by its official documentation is as follows–

AngularJS is a structural framework for dynamic web applications. It lets you use HTML as your template language and lets you extend HTML's syntax to express your application component \s clearly and succinctly. Its data binding and dependency injection eliminate much of the code you currently have to write. And it all happens within the browser, making it an ideal partner with any server technology.

### **General Features**

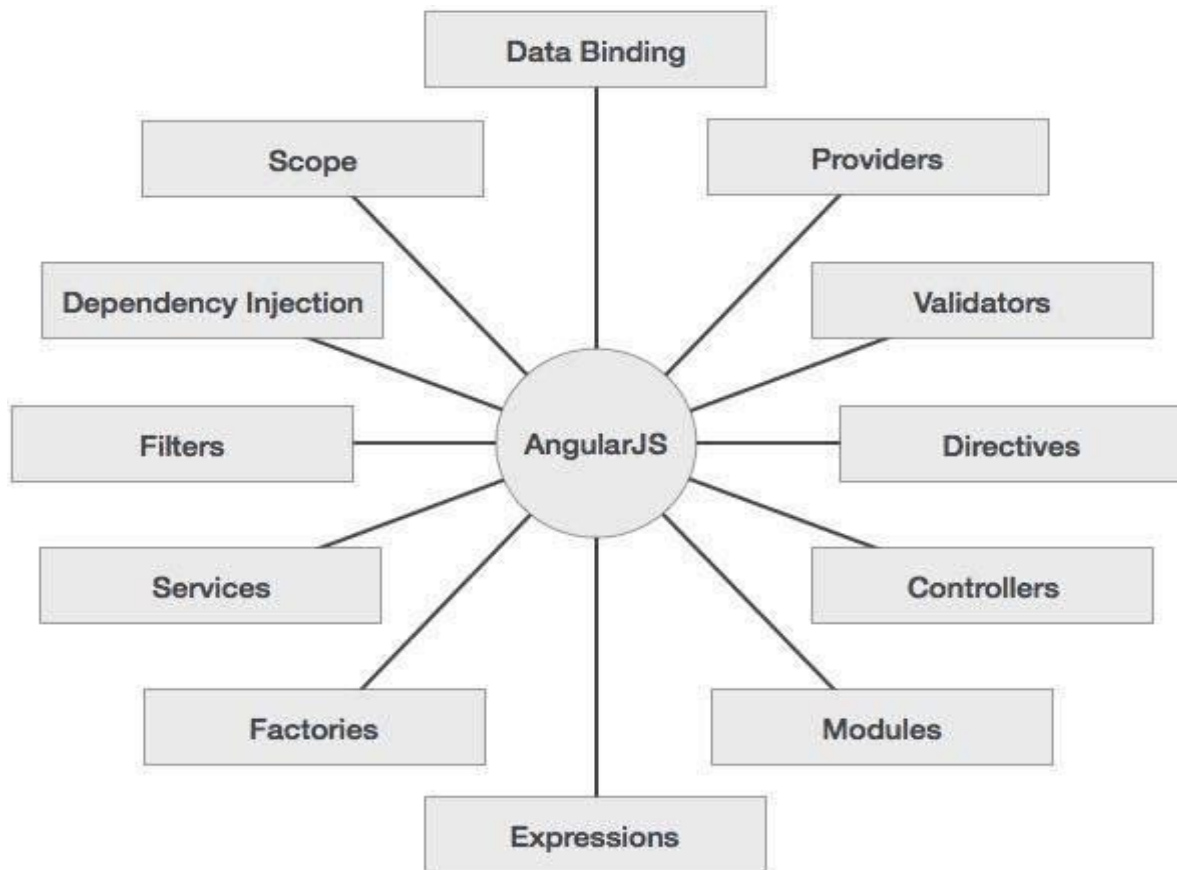
The general features of AngularJS areas follows–

- AngularJS is a efficient framework that can create RichInternetApplications(RIA).
- AngularJS provides developers an options to write client side applications using JavaScript in a clean Model View Controller (MVC) way.
- ApplicationswritteninAngularJSarecross-browsercompliant.AngularJSautomatically handles JavaScript code suitable for each browser.
- AngularJS is open source, completely free, and used by thousands of developers around the world. It is licensed under the Apache license version 2.0.
- Overall, AngularJS isa frame work to build large scale, high-performance, and easy to-maintain web applications.

### **Core Features:**

The core features of AngularJS are as follows–

- Data-binding–Itistheautomaticsynchronizationofdatabetweenmodelandview components.
- Scope – These are objects that refer to the model. They act as a glue between controller and view.
- Controller–These are JavaScript functions bound to a particular scope.
- Services–AngularJScomeswithseveralbuilt-inservicessuchas\$httptomakea XML Http Requests. These are singleton objects which are instantiated only once in app.
- Filters–These select a subse of items from an array and returns a new array.
- Directives – Directives are markers on DOM elements such as elements, attributes, css, and more. These can be used to create custom HTML tags that serve as new, custom widgets. AngularJS has built-in directives such as ngBind, ngModel, etc.
- Templates – These are the rendered view with information from the controller and model. These can be a single file (such as index.html) or multiple views in one page using partials.
- Routing–It is concept of switching views.
- Deep Linking – Deep linking allows to encode the state of application in the URL so that it can be bookmarked. The application can then be restored from the URL to the same state.
- Dependency Injection – AngularJS has a built-in dependency injection subsystem that helps the developer to create, understand, and test the applications easily.



### **Advantages of Angular JS:**

The advantages of AngularJS are–

- It provides the capability to create Single Page Application in a very clean and maintainable way.
- It provides data binding capability to HTML. Thus, it gives user a rich and responsive experience.
- AngularJS code is unit testable.
- AngularJS uses dependency injection and makes use of separation of concerns.
- AngularJS provides reusable components.
- With AngularJS, the developers can achieve more functionality with short code.
- In AngularJS, views are pure HTML pages, and controllers written in JavaScript do the business processing.
- On the top of everything, AngularJS applications can run on all major browsers and smart phones, including Android and iOS based phones/tablets.

## Building Angular JS

```
<html>
<head>
<scriptsrc="angular.min.js"></script>
<script>
var app=angular.module("myapp",[ ])
app.controller("myctrl",function($scope){
$scope.s=[{ rollno:"1",
name:"syed",
dept:"it", address:"hyd"    },
{
    rollno:"2",
    name:"mohd",
    dept:"cse",
    address:"chennai"
    },
{
    rollno:"3",
    name:"uddin",
    dept:"mech",
    address:"bangalore"
    },
{
    rollno:"4",
    name:"ahmed",
    dept:"ece",
    address:"mumbai"
    },
{
    rollno:"5",
    name:"ali",
    dept:"civil",
    address:"delhi"
    }
    ]]
    })
    </script>
    </head>
    <bodyng-app="myapp">
    <divng-controller="myctrl">
    <tableborder="4">
    <tr>
    <td>rollno</td>
    <td>name</td>
    <td>dept</td>
    <td>address</td>
    </tr>
    <trng-repeat="studentin s">
    <td>{{ student.rollno }}</td>
    <td>{{ student.name }}</td>
    <td>{{ student.dept }}</td>
    <td>{{ student.address }}</td>
    </tr>
    </table>
    </body></html>
```

## Input Validation using AngularJS:

---

```
<!DOCTYPE html>
<html>
<scriptsrc="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.9/angular.min.js">
</script>
<body>

<h2>AngularJS Validation Example</h2>

<form ng-app="myApp"ng-controller="validateCtrl" name="myForm" novalidate>

<p>Username:<br>
<input type="text"name="user"ng-model="user"required>
<spanstyle="color:red"ng-show="myForm.user.$dirty&&myForm.user.$invalid">
<spanng-show="myForm.user.$error.required">Usernameisrequired.</span>
</span>
</p>

<p>Email:<br>
<input type="email" name="email" ng-model="email"required>
<spanstyle="color:red"ng-show="myForm.email.$dirty&&myForm.email.$invalid">
<spanng-show="myForm.email.$error.required">Emailisrequired.</span>
<spanng-show="myForm.email.$error.email">Invalidemailaddress.</span>
</span>
</p>

<p>
<input type="submit"
ng-disabled="myForm.user.$dirty&&myForm.user.$invalid|| myForm.email.$dirty &&
myForm.email.$invalid">
</p>

</form>

<script>
var app = angular.module('myApp', []); app.controller('validateCtrl',function($scope){
$scope.user='AzharFarhan';
$scope.email='syedazharfarhan@gmail.com';
});
</script>

</body>
</html>
```

## Program 1

**Develop Angular JS program that allows user to input their first name and last name and display their full name.**

**Note: The default values for first name and last name may be included in the program.**

```
<!DOCTYPE html>
<html>
<script src="angular.min.js"></script>
<body>

<p>Try to change the names.</p>

<div ng-app="myApp" ng-controller="myCtrl">

First Name: <input type="text" ng-model="firstName"><br>
Last Name: <input type="text" ng-model="lastName"><br>
<br>
Full Name: {{ firstName + " " + lastName }}

</div>

<script>
var app = angular.module('myApp', []);
app.controller('myCtrl', function($scope) {
    $scope.firstName= "John";
    $scope.lastName= "Doe";
});
</script>

</body>
</html>
```

### **Out put:**

Try to change the names.

First Name:   
Last Name:

Full Name: John Doe

## Program 2

**Develop an Angular JS application that displays a list of shopping items. Allow users to add and remove items from the list using directives and controllers.**

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

```
<!DOCTYPE html>
<html>
<script src="angular.min.js"></script>
<body>

<script>
var app = angular.module("myShoppingList", []);
app.controller("myCtrl", function($scope) {
    $scope.products = ["Milk", "Bread", "Cheese"];
    $scope.addItem = function () {
        $scope.products.push($scope.addMe);
        $scope.addMe="";
    }
    $scope.removeItem = function (X) {
        $scope.products.splice(X, 1);
    }
});
</script>

<div ng-app="myShoppingList" ng-controller="myCtrl">
    <ul>
        <li ng-repeat="X in products">
            {{X}}
            <span ng-click="removeItem($index)">×</span>
        </li>
    </ul>
    <input ng-model="addMe">
    <button ng-click="addItem()">Add</button>
</div>

<p style="color:red">Click the little X to remove an item from the shopping list.</p>

</body>
</html>
```

**Out put:**

- Milk ×
- Bread ×
- Cheese ×
- Ghee ×
- Vegetables ×

Click the little X to remove an item from the shopping list.

- Bread ×
- Ghee ×

Click the little X to remove an item from the shopping list.

### Program 3

**Develop a simple Angular JS calculator application that can perform basic mathematical operations (addition, subtraction, multiplication, division) based on user input.**

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

<script>
var app = angular.module('CalculatorApp', []);
  app.controller('CalculatorController', function($scope) {
    $scope.result = function() {
      if ($scope.operator == '+') {
        return $scope.a + $scope.b;
      }
      if ($scope.operator == '-') {
        return $scope.a - $scope.b;
      }
      if ($scope.operator == '*') {
        return $scope.a * $scope.b;
      }
      if ($scope.operator == '/') {
        return $scope.a / $scope.b;
      }
    };
  });
</script>

<div ng-app="CalculatorApp" ng-controller="CalculatorController">
  <p><input type="number" ng-model="a"> Choose First Number</p>
  <p><input type="number" ng-model="b"> Choose Second Number</p>
  <p><select ng-model="operator">
    <option>+</option>
    <option>*</option>
    <option>-</option>
    <option>/</option>
  </select> Choose Operator</p>
  <p>{{ result() }}</p>
</div>

</body>
</html>
```



**Out put:**

Choose First Number

Choose Second Number

Choose Operator

43344

Choose First Number

Choose Second Number

Choose Operator

-107140705

Choose First Number

Choose Second Number

Choose Operator

-48034

Choose First Number

Choose Second Number

Choose Operator

-0.05132526428680864

## Program 4

**Write an Angular JS application that can calculate factorial and compute square based on given user input.**

```
<html>
<script src= "angular.min.js"></script>
<script>
var myApp = angular.module('myModule', []);
  myApp.controller('findFactorial', function($scope) {

    $scope.calculateFactorial = function() {
      var num=parseInt($scope.a);
      var i, fact=1, factw;

      if(num<0 || isNaN(num))
      {

        $scope.result="NOT defined factorial";

      }
      else
      {

        for(i=num;i>=1;i--)
        {
          fact=fact*i;
        }
        $scope.result="factorial Of " + num + " is " + fact;

      }
    }
    $scope.calculateSquare=function()
    {
      var n1=parseInt($scope.a)
      var square=n1*n1;
      $scope.result="Square Of " + n1 + " is " + square;
    }
  });
</script>

<body ng-app="myModule" ng-controller="findFactorial">
<div>Enter a Number: <input type="text" ng-model="a"></p><br>
<input type="button" ng-click="calculateFactorial()" value="calculateFactorial">
<input type="button" ng-click="calculateSquare()" value="calculateSquare"><br><br>
  {{result}}
</div>
</body>
</html>
```

**Out put:**

Enter a Number:

factorial Of 5 is 120

Enter a Number:

Square Of 5 is 25

## Program 5

**Develop AngularJS application that displays a details of students and their CGPA. Allow users to read the number of students and display the count.**

**Note: Student details may be included in the program.**

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

<style>
body {
margin: 1%;
font-size: 120%;
background-color: pink;
}
</style>

<body ng-app="myApp" ng-controller="ListController">
<h1 align='center'>Student Details</h1>
<table border=1 align='center'>

<tr>
<th>Roll.No</th>
<th>Name</th>
<th>City</th>
<th>Mobile No.</th>
<th>CGPA</th>
</tr>

<tr ng-repeat="item in itemsDetails">
<td> {{ item.Rollno }} </td>
<td> {{ item.name }} </td>
<td> {{ item.city }} </td>
<td> {{ item.mobilenos }} </td>
<td> {{ item.CGPA }} </td>
</tr>
</table><br><br>

<p align='center'> The Number Of Student Details In The Table :  {{ hrr }}</p>
</body>

<script>
var app = angular.module('myApp', []);
app.controller('ListController', function($scope) {

$scope.itemsDetails = [
{
Rollno: 01,
name: 'Ragu ',
```

city: 'Karanatak ',  
mobilen: 9978689865 ,  
CGPA:9.9 ,  
},

{  
Rollno: 02,  
name: 'Adhir',  
city: 'Pune',  
mobilen: 9978657865,  
CGPA:7.9,  
},

{  
Rollno: 03,  
name: 'Adhira',  
city: 'tamilanadu',  
mobilen: 8778657865,  
CGPA:8.23,  
},

{  
Rollno: 04,  
name: 'Sanket',  
city: 'Mumbai',  
mobilen: 8978657865,  
CGPA:7.9,  
},

{  
Rollno: 05,  
name: 'Tanveer',  
city: 'keral',  
mobilen: 9967657865,  
CGPA:6.34,  
},

{  
Rollno: 06,  
name: 'Sameer',  
city: 'Tamilanadu',  
mobilen: 9945657865,  
CGPA:8.889,  
},

{  
Rollno: 07,  
name: 'Sudhira',  
city: 'Gujarata',  
mobilen: 8878657865,  
CGPA:6.9,  
}

];

```
$scope.hrr=$scope.itemsDetails.length;
```

```
});
```

```
</script>
```

```
</html>
```

**Out put:**

## Student Details

Roll.No	Name	City	Mobile No.	CGPA
1	Ragu	Karanatak	9978689865	9.9
2	Adhir	Pune	9978657865	7.9
3	Adhira	tamilanadu	8778657865	8.23
4	Sanket	Mumbai	8978657865	7.9
5	Tanveer	keral	9967657865	6.34
6	Sameer	Tamilanadu	9945657865	8.889
7	Sudhira	Gujarata	8878657865	6.9

The Number Of Student Details In The Table : 7

## Program 6

**Develop an AngularJS program to create a simple to-do list application. Allow users to add, edit, and delete tasks.**

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

```
<!DOCTYPE html>
<html ng-app="ToDoApp">
<head>
<script src="angular.min.js"></script>
</head>
<body ng-controller="ToDoController">
<h2>To-Do List</h2>
<div>
<input type="text" ng-model="newTask" placeholder="Add a new task">
<button ng-click="addTask()">Add</button>
</div>
<ul>
<li ng-repeat="task in tasks">
{{ task.name }}
<span ng-show="task.editing">
<input type="text" ng-model="task.name" ng-blur="saveTask(task)">
</span>
<span ng-show="!task.editing">
<button ng-click="editTask(task)">Edit</button>
<button ng-click="removeTask(task)">Delete</button>
</span>
</li>
</ul>
<script>
var app = angular.module('ToDoApp', []);
app.controller('ToDoController', function($scope) {
//Default Tasks
$scope.tasks = [{ name: 'Attend Class'},

{ name: 'Complete Assignment'},
{ name: 'Study for CIE'}];
$scope.addTask = function()
{
if ($scope.newTask)
{
$scope.tasks.push({ name: $scope.newTask, editing: false });
$scope.newTask = "";
}
};
$scope.editTask = function(task)
{
task.editing = true;
};
$scope.saveTask = function(task)
{
task.editing = false;
};
};
```

```
$scope.removeTask = function(task)
{
const index = $scope.tasks.indexOf(task);
if (index !== -1) {
$scope.tasks.splice(index, 1);
}
};
});
</script>
</body>
</html>
```



**Out put:**

## To-Do List

- Attend Class
- Complete Assignment
- Study for CIE
- Workshop By 3rd sem students

## To-Do List

- Attend Class
- Complete Assignment
- Study for CIE from 5th sem students
- Workshop By 3rd sem students

## To-Do List

- Complete Assignment
- Study for CIE from 5th sem students

## Program 7

**Write an AngularJS program to create a simple CRUD application (Create, Read, Update, and Delete) for managing users.**

```
<!DOCTYPE html>
<html >
<head>
  <title></title>
  <script src="angular.min.js"></script>
  <script src="jquery.min.js"></script>
</head>
<body>

  <div ng-app="mainApp" data-ng-controller="CRUDController">

    <table>
      <tr>
        <td>EmpId: </td>
        <td>
          <span>{{ EmpModel.Id }}</span></td>
      </tr>
      <tr>
        <td>Name:</td>
        <td>
          <input type="text" data-ng-model="EmpModel.Name" /></td>
      </tr>
      <tr>
        <td>Salary:</td>
        <td>
          <input type="number" data-ng-model="EmpModel.Salary" /></td>
      </tr>
      <tr>
        <td>
          <input type="button" value="Save Data" data-ng-click="AddData()" /></td>
        <td>
          <input type="button" value="Update Data" data-ng-click="UpdateData()" /></td>
      </tr>
    </table>

    <table border="1" style="width: 300px">
      <thead>
        <th>Id</th>
        <th>Name</th>
        <th>Salary</th>
      </thead>
      <tr data-ng-repeat="Emp in EmpList" data-ng-click="BindSelectedData(Emp)">
        <td>{{ Emp.Id }}</td>
        <td>{{ Emp.Name }}</td>
        <td>{{ Emp.Salary }}</td>
        <td>
          <input type="button" value="Delete" data-ng-click="DeleteData(Emp)" /></td>
      </tr>
    </table>
  </div>
</body>
</html>
```

```
</tr>
</table>
</div>
```

```
<script type="text/javascript">
  var app = angular.module("mainApp", []);
  app.controller('CRUDController', function ($scope) {
```

```
    $scope.EmpModel = {
      Id: 0,
      Salary: 0,
      Name: "",
    };
```

```
    $scope.EmpList = [];
    $scope.AddData = function () {
      var _emp = {
        Id: $scope.EmpList.length + 1,
        Name: $scope.EmpModel.Name,
        Salary: $scope.EmpModel.Salary
      };
      $scope.EmpList.push(_emp);
      ClearModel();
    }
  }
```

```
    $scope.DeleteData = function (emp) {
      var _index = $scope.EmpList.indexOf(emp);
      $scope.EmpList.splice(_index, 1);
    }
  }
```

```
    $scope.BindSelectedData = function (emp) {
      $scope.EmpModel.Id = emp.Id;
      $scope.EmpModel.Name = emp.Name;
      $scope.EmpModel.Salary = emp.Salary;
    }
  }
```

```
    $scope.UpdateData = function () {
      $.grep($scope.EmpList, function (e) {
        if (e.Id == $scope.EmpModel.Id) {
          e.Name = $scope.EmpModel.Name;
          e.Salary = $scope.EmpModel.Salary;
        }
      });
      ClearModel();
    }
  }
```

```
    function ClearModel() {
      $scope.EmpModel.Id = 0;
      $scope.EmpModel.Name = "";
      $scope.EmpModel.Salary = 0;
    }
  }
```

```
});
```

```
</script>
</body>
</html>
```

### Out put:

EmpId: 0

Name:

Salary:

Id	Name	Salary	
1	Amir K	300000	<input type="button" value="Delete"/>
2	Sameer Kalpan	440000	<input type="button" value="Delete"/>
3	Sandeep Kyattannavar	870000	<input type="button" value="Delete"/>
4	Prakash Guggari	250000	<input type="button" value="Delete"/>
5	Vishwanath K	750000	<input type="button" value="Delete"/>
6	Sunil Vernker	340000	<input type="button" value="Delete"/>

EmpId: 2

Name:

Salary:

Id	Name	Salary	
1	Amir K	300000	<input type="button" value="Delete"/>
2	Samer Kalpan	450000	<input type="button" value="Delete"/>
3	Sandeep Kyattannavar	870000	<input type="button" value="Delete"/>
4	Prakash Guggari	250000	<input type="button" value="Delete"/>
5	Vishwanath K	750000	<input type="button" value="Delete"/>
6	Sunil Vernker	340000	<input type="button" value="Delete"/>

EmpId: 0

Name:

Salary:

Id	Name	Salary	
1	Amir K	300000	<input type="button" value="Delete"/>
3	Sandeep Kyattannavar	870000	<input type="button" value="Delete"/>
5	Vishwanath K	750000	<input type="button" value="Delete"/>
6	Sunil Vernker	340000	<input type="button" value="Delete"/>

## Program 8

**Develop AngularJS program to create a login form, with validation for the username and password fields.**

```
<!DOCTYPE html>
<html>
<head>
<title>Login Form with Validation</title>
<script src="angular.min.js">
</script>
</head>
<body ng-app="myApp">
<div ng-controller="LoginController">
<h2>Login Form with Validation</h2>
<form name="loginForm" novalidate>
<label>Username:</label><br><br>
<input type="text" ng-model="username" name="username" required>
<span ng-show="loginForm.username.$error.required && loginForm.username.$dirty"> Username is
required.</span>
<br><br>
<label>Password :</label><br><br>
<input type="password" ng-model="password" name="password" ng-pattern="/^(?=.*[A-Za-
z])(?=.*\d)[A-Za-z\d]{8,}$/" required>
<span ng-show = "loginForm.password.$error.required && loginForm.password.$dirty"> Password is
required.</span>
<span ng-show="loginForm.password.$error.pattern && loginForm.password.$dirty">
Password must be alphanumeric and at least 8 characters long.
</span>
<br><br>
<button ng-click="login()" ng-disabled="loginForm.$invalid">Login</button>
</form>
<div ng-show="isLoggedIn">
<p>Login successful! Welcome, {{ username }}!</p>
</div>
</div>
<script>
var app = angular.module('myApp', []);
app.controller('LoginController', function ($scope) {
$scope.user = { username: '', password: '' };
$scope.isLoggedIn = false;
$scope.login = function () {
$scope.isLoggedIn = true;
};
});
</script>
</body>
</html>
```

**Out put:**

## Login Form with Validation

Username:

Password :

Password must be alphanumeric and at least 8 characters long.

## Login Form with Validation

Username:

Password :

Login successful! Welcome, Raghunath!

## Program 9

**Create an AngularJS 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.**

```
<!DOCTYPE html>
<html ng-app="employeeApp">
<head>
<title>Employee List</title>
<script src="angular.min.js"></script>
</head>
<body ng-controller="EmployeeController">
<h2>Employee List</h2>
<label>Search by Name:</label>
<input type="text" ng-model="searchName" />
<label>Search by Salary:</label>
<input type="number" ng-model="searchSalary" />
<button ng-click="searchEmployees()">Search</button>
<ul>
<li ng-repeat="employee in filteredEmployees">
{{ employee.name }} - {{ employee.salary }}
</li>
</ul>
<script>
angular.module('employeeApp', [])
.controller('EmployeeController', function ($scope) {
$scope.employees = [
{ name: 'Keerthana SS', salary: 53000 },
{ name: 'Rakshith B', salary: 60000 },
{ name: 'Santhosh', salary: 70000 },
{ name: 'Radhika Pai', salary: 55000 },
{ name: 'Skandha', salary: 80000 }
];

$scope.filteredEmployees = $scope.employees;
$scope.searchEmployees = function () {
$scope.filteredEmployees = $scope.employees.filter(function (employee) {
return (
(employee.name.toLowerCase().includes($scope.searchName.toLowerCase()) || !$scope.searchName)
&&
(employee.salary == $scope.searchSalary || !$scope.searchSalary)
);
});
});
});
</script>
</body>
</html>
```

**Out put:**

## Employee List

Search by Name:  Search by Salary:

Search

- Keerthana SS - 53000
- Rakshith B - 60000
- Santhosh - 70000
- Radhika Pai - 55000
- Skandha - 80000

## Employee List

Search by Name:  Search by Salary:

Search

- Radhika Pai - 55000

## Employee List

Search by Name:  Search by Salary:

Search

- Santhosh - 70000

## Employee List

Search by Name:  Search by Salary:

Search

## Employee List

Search by Name:  Search by Salary:

Search



## Program 10

**Create AngularJS 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.**

```
<!DOCTYPE html>
<html>
<script src="angular.min.js"></script>
<body>

<script>
var app = angular.module("myShoppingList", []);
app.controller("myCtrl", function($scope) {
    $scope.products = ["Milk ", "Bread ", "Cheese          "];
    $scope.arrr=$scope.products.length;
    $scope.addItem = function () {
        $scope.errortext = "";
        if (!$scope.addMe) {return;}
        if ($scope.products.indexOf($scope.addMe) == -1) {

            $scope.arrr=$scope.products.length+1;
            $scope.products.push($scope.addMe);
            $scope.addMe="";
            $scope.arrr=$scope.arrr++;

        } else {
            $scope.errortext = "The Item Is Already In Your Shopping List.";
        }
    }
    $scope.removeItem = function (x) {
        $scope.errortext = "";

        $scope.products.splice(x,1);
        $scope.arrr=$scope.products.length;
        $scope.arrr=$scope.arrr--;

    }
});
</script>

<div ng-app="myShoppingList" ng-controller="myCtrl">

    <p>The Number Of Items In Current COLLECTION : {{ arrr }}</p>
    <ul>
        <li ng-repeat="x in products">{{ x }}<button ng-click="removeItem($index)"> Remove
    </button></li>
    </ul>
    <input placeholder="Enter An Item" ng-model="addMe" >
    <button ng-click="addItem()">Add</button>
```

<p>{{errortext}}</p>  
</div>

</body>  
</html>

## Out put:

The Number Of Items In Current COLLECTION : 3

- Milk
- Bread
- Cheese

The Number Of Items In Current COLLECTION : 5

- Milk
- Bread
- Cheese
- Butter
- Butter\_milk

The Number Of Items In Current COLLECTION : 3

- Bread
- Butter
- Butter\_milk

## Program 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>
  <head>
    <title>Angular JS Filters</title>
    <script src = "angular.min.js">
    </script>
  </head>

  <body>
    <h2>Student Details In UPPARCASE</h2>

    <div ng-app = "mainApp" ng-controller = "studentController">
      <table border = "1">
        <tr>
          <td>Enter first name:</td>
          <td><input type = "text" ng-model = "student.firstName"></td>
        </tr>
        <tr>
          <td>Enter last name: </td>
          <td><input type = "text" ng-model = "student.lastName"></td>
        </tr>

        <tr>
          <td>Enter usn: </td>
          <td><input type = "text" ng-model = "student.usn"></td>
        </tr>

        <tr>
          <td>Enter branch: </td>
          <td><input type = "text" ng-model = "student.branch"></td>
        </tr>

        <tr>
          <td>Enter subject1: </td>
          <td><input type = "text" ng-model = "student.sub1"></td>
        </tr>
      <tr>
        <td>Enter subject2: </td>
        <td><input type = "text" ng-model = "student.sub2"></td>
      </tr><br>
      <tr>
        <td>Enter subject3: </td>
        <td><input type = "text" ng-model = "student.sub3"></td>
      </tr>
    </div>
  </body>
</html>
```

```
        <tr>
        <td>Enter sem: </td>
        <td><input type = "text" ng-model = "student.sem"></td>
    </tr>
```

```
    <tr>
    <td>Enter collage name: </td>
    <td><input type = "text" ng-model = "student.colg"></td>
    </tr>
```

```
</table>
    <p>NAME In Uppercase: {{ student.fullName() | uppercase }}</p>
    <p>USN In Uppercase: {{ student.usn | uppercase }}</p>
    <p>BRANCH In Uppercase: {{ student.branch | uppercase }}</p>
    <p>SUBJECT1 In Uppercase: {{ student.sub1 | uppercase }}</p>
    <p>SUBJECT2 In Uppercase: {{ student.sub2 | uppercase }}</p>
    <p>SUBJECT3 In Uppercase: {{ student.sub3 | uppercase }}</p>
    <p>SEM In Uppercase: {{ student.sem | uppercase }}</p>
    <p>COLLEGE NAME In Uppercase: {{ student.colg | uppercase }}</p>
```

```
</div>
```

```
<script>
```

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

```
    mainApp.controller('studentController', function($scope) {
```

```
        $scope.student = {
```

```
            firstName: "",
```

```
            lastName: "",
```

```
            usn: "",
```

```
            branch: "",
```

```
            sub1: "",
```

```
            sub2: "",
```

```
            sub3: "",
```

```
            sem: "",
```

```
            colg: "",
```

```
            fullName: function() {
```

```
                var studentObject;
```

```
                studentObject = $scope.student;
```

```
                return studentObject.firstName + " " + studentObject.lastName;
```

```
            }
```

```
        };
```

```
    });
```

```
</script>
```

```
</body>
```

```
</html>
```

## **Out put:**

### **Student Details In UPPARCASE**

Enter first name:	nagaraj b
Enter last name:	baradeli
Enter usn:	2ka05cs026
Enter branch:	computer science and engin
Enter subject1:	java programming
Enter subject2:	andriod applications
Enter subject3:	angular js
Enter sem:	i
Enter collage name:	smt. kamala and sri venkapp

NAME In Uppercase: NAGARAJ B BARADELI

USN In Uppercase: 2KA05CS026

BRANCH In Uppercase: COMPUTER SCIENCE AND ENGINEERING

SUBJECT1 In Uppercase: JAVA PROGRAMMING

SUBJECT2 In Uppercase: ANDRIOD APPLICATIONS

SUBJECT3 In Uppercase: ANGULAR JS

SEM In Uppercase: I

COLEEGE NAME In Uppercase: SMT. KAMALA AND SRI VENKAPPA M. AGADI COLLEGE OF ENGINEERING AND TECHNOLOGY, LAXMESHWAR

## Program 12

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

```
<!DOCTYPE html>

<html>

<head>

<title>Date Filter</title>

<script src="angular.min.js">

</script>

</head>

<body>

<div ng-app="firstApp" ng-controller="firstCntrl">

<h1> DATE FORMATS</h1>

<p>{{ today | date : "dd.MMMM.yyyy" }}</p>
<p>{{ today | date : "d.MMMM.yyyy" }}</p>
<p>{{ today | date : "d.MMM.yyyy" }}</p>
<p>{{ today | date : "d.MM.yyyy" }}</p>
<p>{{ today | date : "d.M.yyyy" }}</p>
<p>{{ today | date : "d.M.yy" }}</p>
<p>{{ today | date : "d.M.y" }}</p>

<p>{{ today | date : "short" }}</p>
<p>{{ today | date : "medium" }}</p>
<p>{{ today | date : "shortDate" }}</p>
<p>{{ today | date : "mediumDate" }}</p>
<p>{{ today | date : "longDate" }}</p>
<p>{{ today | date : "fullDate" }}</p>

</div>

<script>

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

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

$scope.today = new Date();

});

</script>

</body> </html>
```

Out put:

## DATE FORMATS

25.January.2024

25.January.2024

25.Jan.2024

25.01.2024

25.1.2024

25.1.24

25.1.2024

1/25/24 4:02 PM

Jan 25, 2024 4:02:28 PM

1/25/24

Jan 25, 2024

January 25, 2024

Thursday, January 25, 2024