**Practical-02**

**Aim:** Advanced Angular Js.

1. Sorted Arrays:

Code:

<!DOCTYPE html>

<html>

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

    <body>

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

            <p>Type a letter in the input field:</p>

            <p><input type="text" ng-model="test"></p>

            <ul>

                <li ng-repeat="x in names | filter:test">

                    {{ x }}

                </li>

            </ul>

        </div>

        <script>

            angular.module('myApp', []).controller('namesCtrl', function($scope){

                $scope.names = [

                'Vishal',

                'Rahul',

                'Poonam',

                'Shreya',

                'Sandhya',

                'Virat',

                'Varun',

                'Aditya',

                'Harsh',

                'Aadharsh',

                'Vansh',

                'Devansh',

                'Ved'];

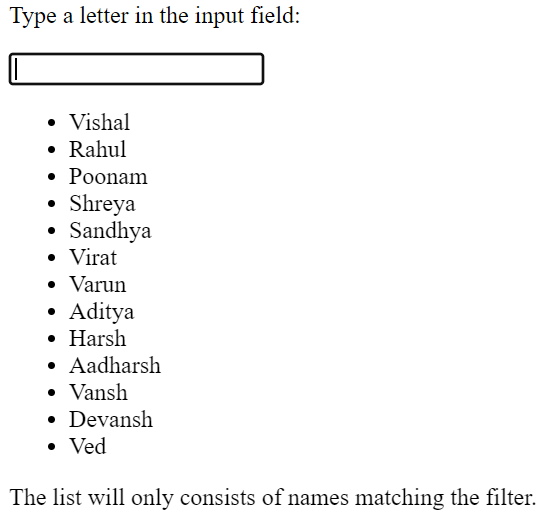
            });

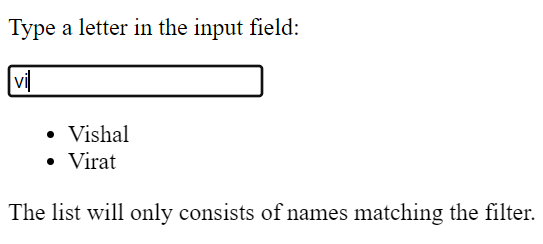
        </script>

        <p>The list will only consists of names matching the filter.</p>

    </body>

</html>

Output:



1. Sorting in tables.

Code:

<!DOCTYPE html>

<html>

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

    <body>

        <p>Click the table headers to change the sorting order:</p>

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

            <table border="1" width="100%">

                <tr>

                    <th ng-click="orderByMe('name')">Name</th>

                    <th ng-click="orderByMe('country')">Country</th>

                </tr>

                <tr ng-repeat="x in names | orderBy:myOrderBy">

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

                    <td>{{ x.country }}</td>

                </tr>

            </table>

        </div>

        <script>

            angular.module('myApp', []).controller('namesCtrl', function($scope){

                $scope.names = [

                {name: 'Jani', country: 'Norway'},

                {name: 'Carl', country: 'Sweden'},

                {name: 'Margareth', country: 'England'},

                {name: 'Hege', country: 'Norway'},

                {name: 'Joe', country: 'Denmark'},

                {name: 'Gustav', country: 'Sweden'},

                {name: 'Birgit', country: 'Denmark'},

                {name: 'Mary', country: 'England'},

                {name: 'Kai', country: 'Norway'}

                ];

            $scope.orderByMe = function(x){

                $scope.myOrderBy = x;

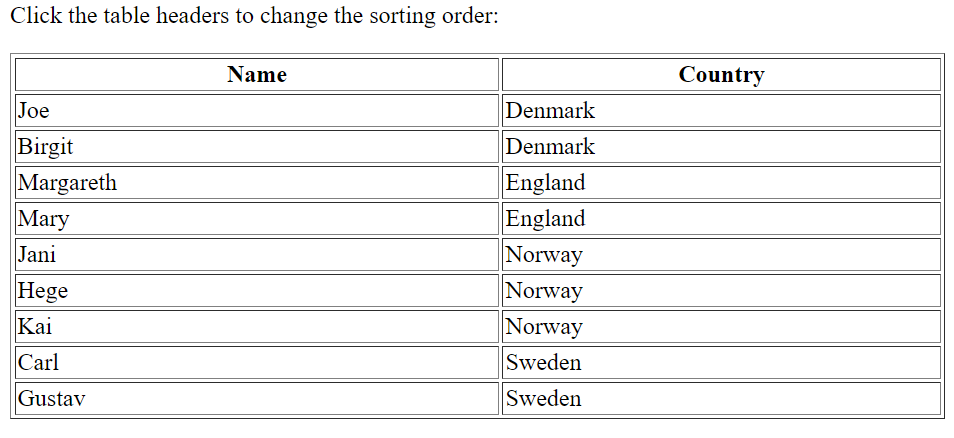
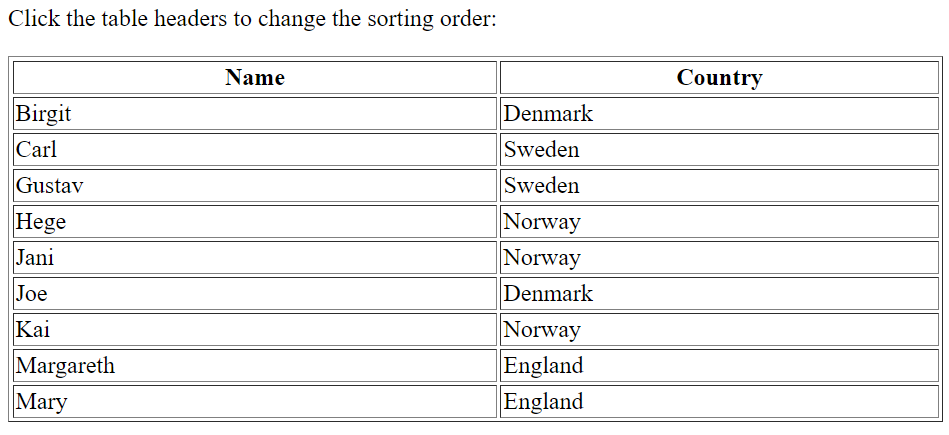
            }

            });

        </script>

    </body>

</html>

Output:

1. Show data on click of button.

Code:

<!DOCTYPE html>

<html>

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

    <body>

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

            <button ng-click="myFunc()">Click Me!</button>

            <div ng-show="showMe">

                <h1>Menu:</h1>

                <div>Pizza</div>

                <div>Pasta</div>

                <div>Pesce</div>

            </div>

        </div>

        <script>

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

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

                $scope.showMe = false;

                $scope.myFunc = function(){

                    $scope.showMe = !$scope.showMe;

                }

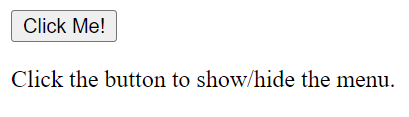
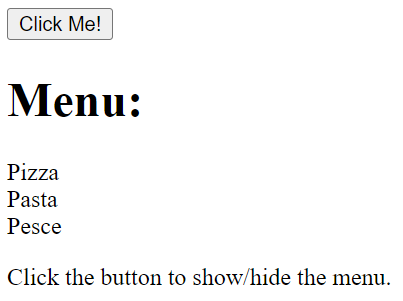
            });

        </script>

        <p>Click the button to show/hide the menu.</p>

    </body>

</html>

Output:

1. Form input field validation.

Code:

<!DOCTYPE html>

<html>

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

    <style>

        input.ng-invalid{

            background-color: pink;

        }

        input.ng-valid{

            background-color: lightgreen;

        }

    </style>

    <body ng-app="">

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

        <form name="myForm">

            <input name="myName" ng-model="myName" required>

            <span ng-show="myForm.myName.$touched && myForm.myName.$invalid">The name is required.</span>

            <p>The input field requires content, and will therefore become green when you write in it.</p>

            <br><br>

            <input name="myInput" ng-model="myInput" required>

        </form>

        <p>The input's valid state is:</p>

        <h1>{{ myForm.myInput.$valid }}</h1>

    </body>

</html>

Output:

