

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>
  <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.min.js">
    </script>

    <script>
      var app=angular.module("empSearchApp",[]);
      app.controller("empSearchAppCntrl",function($scope){
        $scope.empList=[
          {'name':'Harish Kumar B T','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>
            <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>

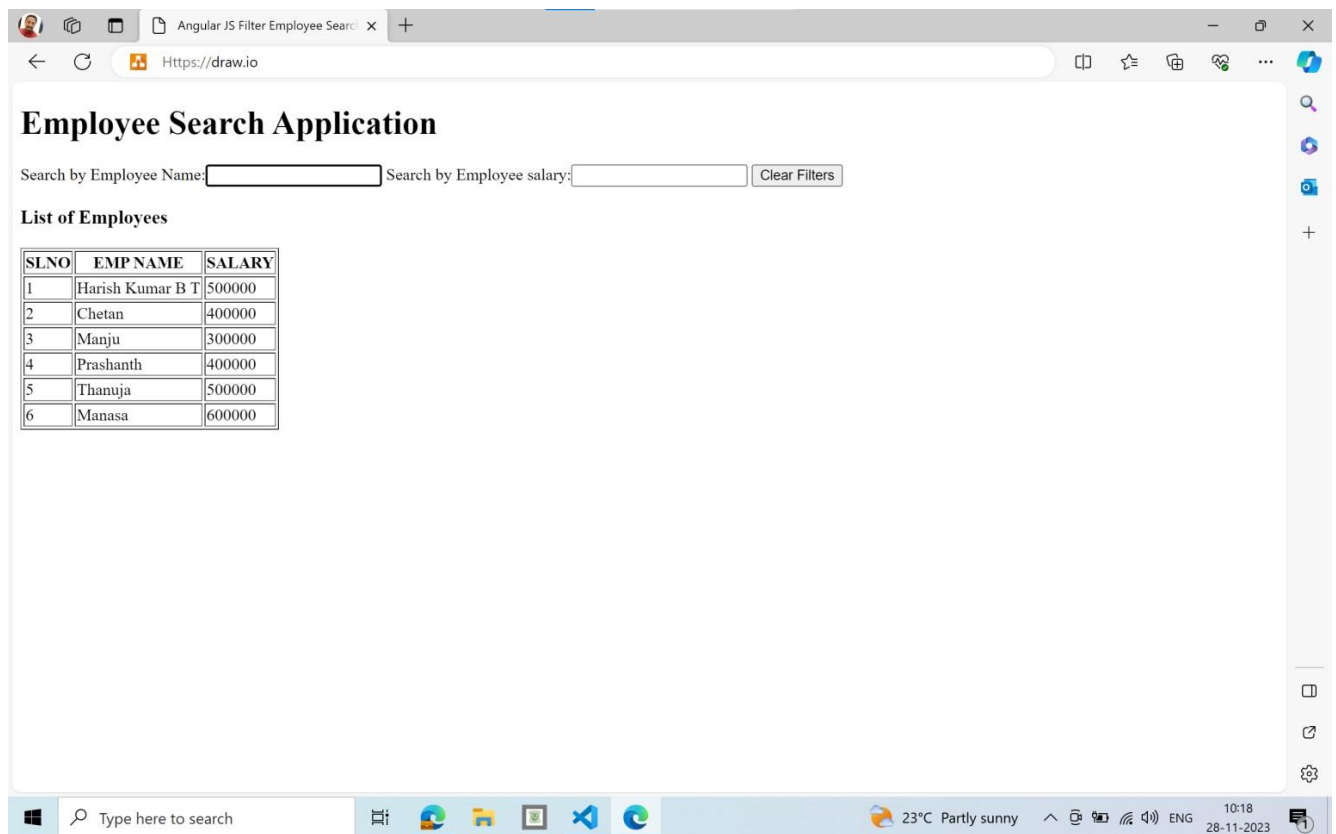
        </tr>

    </table>

</div>
</body>
</html>

```

Output:



The screenshot shows a web browser window with the URL <https://draw.io>. The page displays an "Employee Search Application". At the top, there are two search input fields: "Search by Employee Name:" and "Search by Employee salary:", followed by a "Clear Filters" button. Below the search fields, there is a section titled "List of Employees" which contains a table with the following data:

SLNO	EMP NAME	SALARY
1	Harish Kumar B T	500000
2	Chetan	400000
3	Manju	300000
4	Prashanth	400000
5	Thanuja	500000
6	Manasa	600000

Angular JS Filter Employee Search

+

←

↻

https://draw.io

🔍

🔖

🔗

🔒

⋮

🌐

Employee Search Application

Search by Employee Name: Search by Employee salary: Clear Filters

List of Employees

SLNO	EMP NAME	SALARY
1	Harish Kumar B T	500000
2	Chetan	400000
3	Prashanth	400000
4	Thanuja	500000

🏠

🔍 Type here to search

📅

🌐

📁

📄

🔗

🌐

🌤️ 23°C Partly sunny

⤴️

🔊

🔊

ENG

10:18

28-11-2023

🔔

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>
  <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>
</html>
```

```

        });
    </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/>

    <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>

```

Output:

The screenshot shows a web browser window with the title 'Item Management Application'. The address bar shows the file path 'C:/Users/user/Desktop/AngularFDP/Lab10/lab10.html'. The page content includes a form to add items and a table listing existing items.

Enter an item to add:

List of Items

SLNO	Item	Remove
1	Pen	<input type="button" value="Remove"/>
2	Pencil	<input type="button" value="Remove"/>
3	Eraser	<input type="button" value="Remove"/>
4	Book	<input type="button" value="Remove"/>

Total Number of Items=4

The browser's taskbar at the bottom shows the Windows logo, a search bar with 'Type here to search', and several application icons. The system tray on the right displays the weather as '23°C Partly sunny', the date '28-11-2023', and the time '10:19'.